Consolidated Platform Configuration Guide, Cisco IOS XE Release 3.2SE (Catalyst 3850 Switches)

First Published: January 29, 2013

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Text Part Number: OL-29322-01

CONTENTS

CHAPTER 74

Preface

Document Conventions

This document uses the following conventions:

<table>
<tr><th>Convention</th><th>Description</th></tr>
<tr><td>^ or Ctrl</td><td>Both the ^ symbol and Ctrl represent the Control (Ctrl) key on a keyboard. For example, the key combination ^D or Ctrl-D means that you hold down the Control key while you press the D key. (Keys are indicated in capital letters but are not case sensitive.)</td></tr>
<tr><td>bold font</td><td>Commands and keywords and user-entered text appear in bold font.</td></tr>
<tr><td>Italic font</td><td>Document titles, new or emphasized terms, and arguments for which you supply values are in italic font.</td></tr>
<tr><td><code>Courier</code> font</td><td>Terminal sessions and information the system displays appear in <code>courier</code> font.</td></tr>
<tr><td><code>Bold Courier</code> font</td><td><code>Bold Courier</code> font indicates text that the user must enter.</td></tr>
<tr><td>[x]</td><td>Elements in square brackets are optional.</td></tr>
<tr><td>...</td><td>An ellipsis (three consecutive nonbolded periods without spaces) after a syntax element indicates that the element can be repeated.</td></tr>
<tr><td>|</td><td>A vertical line, called a pipe, indicates a choice within a set of keywords or arguments.</td></tr>
<tr><td>[x | y]</td><td>Optional alternative keywords are grouped in brackets and separated by vertical bars.</td></tr>
</table>

Convention	Description
{x | y}	Required alternative keywords are grouped in braces and separated by vertical bars.
[x {y | z}]	Nested set of square brackets or braces indicate optional or required choices within optional or required elements. Braces and a vertical bar within square brackets indicate a required choice within an optional element.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
< >	Nonprinting characters such as passwords are in angle brackets.
[ ]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

Reader Alert Conventions

This document may use the following conventions for reader alerts:

Note Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the manual.

Tip Means *the following information will help you solve a problem.*

Caution Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

Timesaver Means *the described action saves time.* You can save time by performing the action described in the paragraph.

Warning **Means *reader be warned.* In this situation, you might perform an action that could result in bodily injury.**

Related Documentation

Note Before installing or upgrading the switch, refer to the switch release notes.

- Cisco Catalyst 3850 Switch documentation, located at:
  http://www.cisco.com/go/cat3850_docs
- Cisco SFP and SFP+ modules documentation, including compatibility matrixes, located at:
  http://www.cisco.com/en/US/products/hw/modules/ps5455/tsd_products_support_series_home.html
- Cisco Validated Designs documents, located at:
  http://www.cisco.com/go/designzone
- Error Message Decoder, located at:
  https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service and Cisco currently supports RSS version 2.0.

CHAPTER 1

Using the Command-Line Interface

Information About Using the Command-Line Interface

Command Modes

The Cisco IOS user interface is divided into many different modes. The commands available to you depend on which mode you are currently in. Enter a question mark (?) at the system prompt to obtain a list of commands available for each command mode.

You can start a CLI session through a console connection, through Telnet, a SSH, or by using the browser.

When you start a session, you begin in user mode, often called user EXEC mode. Only a limited subset of the commands are available in user EXEC mode. For example, most of the user EXEC commands are one-time commands, such as **show** commands, which show the current configuration status, and **clear** commands, which clear counters or interfaces. The user EXEC commands are not saved when the switch reboots.

To have access to all commands, you must enter privileged EXEC mode. Normally, you must enter a password to enter privileged EXEC mode. From this mode, you can enter any privileged EXEC command or enter global configuration mode.

Using the configuration modes (global, interface, and line), you can make changes to the running configuration. If you save the configuration, these commands are stored and used when the switch reboots. To access the various configuration modes, you must start at global configuration mode. From global configuration mode, you can enter interface configuration mode and line configuration mode.

This table describes the main command modes, how to access each one, the prompt you see in that mode, and how to exit the mode.

Table 1: Command Mode Summary

Mode	Access Method	Prompt	Exit Method	About This Mode
User EXEC	Begin a session using Telnet, SSH, or console.	`Switch>`	Enter **logout** or **quit**.	Use this mode to • Change terminal settings. • Perform basic tests. • Display system information.
Privileged EXEC	While in user EXEC mode, enter the **enable** command.	`Switch#`	Enter **disable** to exit.	Use this mode to verify commands that you have entered. Use a password to protect access to this mode.
Global configuration	While in privileged EXEC mode, enter the **configure** command.	`Switch(config)#`	To exit to privileged EXEC mode, enter **exit** or **end**, or press **Ctrl-Z**.	Use this mode to configure parameters that apply to the entire switch.
VLAN configuration	While in global configuration mode, enter the **vlan** *vlan-id* command.	`Switch(config-vlan)#`	To exit to global configuration mode, enter the **exit** command. To return to privileged EXEC mode, press **Ctrl-Z** or enter **end**.	Use this mode to configure VLAN parameters. When VTP mode is transparent, you can create extended-range VLANs (VLAN IDs greater than 1005) and save configurations in the switch startup configuration file.
Interface configuration	While in global configuration mode, enter the **interface** command (with a specific interface).	`Switch(config-if)#`	To exit to global configuration mode, enter **exit**. To return to privileged EXEC mode, press **Ctrl-Z** or enter **end**.	Use this mode to configure parameters for the Ethernet ports.

Mode	Access Method	Prompt	Exit Method	About This Mode
Line configuration	While in global configuration mode, specify a line with the **line vty** or **line console** command.	`Switch(config-line)#`	To exit to global configuration mode, enter **exit**. To return to privileged EXEC mode, press **Ctrl-Z** or enter **end**.	Use this mode to configure parameters for the terminal line.

Using the Help System

You can enter a question mark (?) at the system prompt to display a list of commands available for each command mode. You can also obtain a list of associated keywords and arguments for any command.

SUMMARY STEPS

1. **help**
2. *abbreviated-command-entry* **?**
3. *abbreviated-command-entry* <Tab>
4. **?**
5. *command* **?**
6. *command keyword* **?**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**help** **Example:** `Switch# help`	Obtains a brief description of the help system in any command mode.

Step 2	*abbreviated-command-entry* **?**
	
	Example:
	`Switch# di?`
	`dir disable disconnect`
		Obtains a list of commands that begin with a particular character string.

Step 3	*abbreviated-command-entry* <Tab>
	
	Example:
	`Switch# sh conf<tab>`
	`Switch# show configuration`
		Completes a partial command name.

	Command or Action	Purpose
Step 4	**?** **Example:** `Switch> ?`	Lists all commands available for a particular command mode.
Step 5	*command* **?** **Example:** `Switch> show ?`	Lists the associated keywords for a command.

Step 6	*command keyword* **?**
	
	Example:
	`Switch(config)# cdp holdtime ?`
	`<10-255> Length of time (in sec) that receiver must keep this packet`
		Lists the associated arguments for a keyword.

Understanding Abbreviated Commands

You need to enter only enough characters for the switch to recognize the command as unique.

This example shows how to enter the **show configuration** privileged EXEC command in an abbreviated form:

```
Switch# show conf
```

No and Default Forms of Commands

Almost every configuration command also has a **no** form. In general, use the **no** form to disable a feature or function or reverse the action of a command. For example, the **no shutdown** interface configuration command reverses the shutdown of an interface. Use the command without the keyword **no** to reenable a disabled feature or to enable a feature that is disabled by default.

Configuration commands can also have a **default** form. The **default** form of a command returns the command setting to its default. Most commands are disabled by default, so the **default** form is the same as the **no** form. However, some commands are enabled by default and have variables set to certain default values. In these cases, the **default** command enables the command and sets variables to their default values.

CLI Error Messages

This table lists some error messages that you might encounter while using the CLI to configure your switch.

Table 2: Common CLI Error Messages

Error Message	Meaning	How to Get Help
`% Ambiguous command: "show con"`	You did not enter enough characters for your switch to recognize the command.	Reenter the command followed by a question mark (?) without any space between the command and the question mark. The possible keywords that you can enter with the command appear.
`% Incomplete command.`	You did not enter all of the keywords or values required by this command.	Reenter the command followed by a question mark (?) with a space between the command and the question mark. The possible keywords that you can enter with the command appear.
`% Invalid input detected at '^' marker.`	You entered the command incorrectly. The caret (^) marks the point of the error.	Enter a question mark (?) to display all of the commands that are available in this command mode. The possible keywords that you can enter with the command appear.

Configuration Logging

You can log and view changes to the switch configuration. You can use the Configuration Change Logging and Notification feature to track changes on a per-session and per-user basis. The logger tracks each configuration command that is applied, the user who entered the command, the time that the command was entered, and the parser return code for the command. This feature includes a mechanism for asynchronous notification to registered applications whenever the configuration changes. You can choose to have the notifications sent to the syslog.

Note Only CLI or HTTP changes are logged.

How to Use the CLI to Configure Features

Configuring the Command History

The software provides a history or record of commands that you have entered. The command history feature is particularly useful for recalling long or complex commands or entries, including access lists. You can customize this feature to suit your needs.

Changing the Command History Buffer Size

By default, the switch records ten command lines in its history buffer. You can alter this number for a current terminal session or for all sessions on a particular line. This procedure is optional.

SUMMARY STEPS

1. **terminal history** [**size** *number-of-lines*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**terminal history** [**size** *number-of-lines*] **Example:** `Switch#` **`terminal history size 200`**	Changes the number of command lines that the switch records during the current terminal session in privileged EXEC mode. You can configure the size from 0 to 256.

Recalling Commands

To recall commands from the history buffer, perform one of the actions listed in this table. These actions are optional.

Note The arrow keys function only on ANSI-compatible terminals such as VT100s.

SUMMARY STEPS

1. **Ctrl-P** or use the **up arrow** key
2. **Ctrl-N** or use the **down arrow** key
3. **show history**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**Ctrl-P** or use the **up arrow** key	Recalls commands in the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Step 2	**Ctrl-N** or use the **down arrow** key	Returns to more recent commands in the history buffer after recalling commands with **Ctrl-P** or the up arrow key. Repeat the key sequence to recall successively more recent commands.

	Command or Action	Purpose
Step 3	**show history** **Example:** `Switch# show history`	Lists the last several commands that you just entered in privileged EXEC mode. The number of commands that appear is controlled by the setting of the **terminal history** global configuration command and the **history** line configuration command.

Disabling the Command History Feature

The command history feature is automatically enabled. You can disable it for the current terminal session or for the command line. This procedure is optional.

SUMMARY STEPS

1. **terminal no history**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**terminal no history** **Example:** `Switch# terminal no history`	Disables the feature during the current terminal session in privileged EXEC mode.

Enabling and Disabling Editing Features

Although enhanced editing mode is automatically enabled, you can disable it and reenable it.

SUMMARY STEPS

1. **terminal editing**
2. **terminal no editing**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**terminal editing** **Example:** `Switch# terminal editing`	Reenables the enhanced editing mode for the current terminal session in privileged EXEC mode.

	Command or Action	Purpose
Step 2	**terminal no editing** **Example:** `Switch# terminal no editing`	Disables the enhanced editing mode for the current terminal session in privileged EXEC mode.

Editing Commands Through Keystrokes

The keystrokes help you to edit the command lines. These keystrokes are optional.

Note The arrow keys function only on ANSI-compatible terminals such as VT100s.

Table 3: Editing Commands

Editing Commands	Description
Ctrl-B or use the **left arrow** key	Moves the cursor back one character.
Ctrl-F or use the **right arrow** key	Moves the cursor forward one character.
Ctrl-A	Moves the cursor to the beginning of the command line.
Ctrl-E	Moves the cursor to the end of the command line.
Esc B	Moves the cursor back one word.
Esc F	Moves the cursor forward one word.
Ctrl-T	Transposes the character to the left of the cursor with the character located at the cursor.
Delete or **Backspace** key	Erases the character to the left of the cursor.
Ctrl-D	Deletes the character at the cursor.
Ctrl-K	Deletes all characters from the cursor to the end of the command line.
Ctrl-U or **Ctrl-X**	Deletes all characters from the cursor to the beginning of the command line.
Ctrl-W	Deletes the word to the left of the cursor.

Esc D	Deletes from the cursor to the end of the word.
Esc C	Capitalizes at the cursor.
Esc L	Changes the word at the cursor to lowercase.
Esc U	Capitalizes letters from the cursor to the end of the word.
Ctrl-V or **Esc Q**	Designates a particular keystroke as an executable command, perhaps as a shortcut.
Return key	Scrolls down a line or screen on displays that are longer than the terminal screen can display. **Note** The More prompt is used for any output that has more lines than can be displayed on the terminal screen, including **show** command output. You can use the **Return** and **Space** bar keystrokes whenever you see the More prompt.
Space bar	Scrolls down one screen.
Ctrl-L or **Ctrl-R**	Redisplays the current command line if the switch suddenly sends a message to your screen.

Editing Command Lines That Wrap

You can use a wraparound feature for commands that extend beyond a single line on the screen. When the cursor reaches the right margin, the command line shifts ten spaces to the left. You cannot see the first ten characters of the line, but you can scroll back and check the syntax at the beginning of the command. The keystroke actions are optional.

To scroll back to the beginning of the command entry, press **Ctrl-B** or the left arrow key repeatedly. You can also press **Ctrl-A** to immediately move to the beginning of the line.

Note The arrow keys function only on ANSI-compatible terminals such as VT100s.

The following example shows how to wrap a command line that extends beyond a single line on the screen.

SUMMARY STEPS

1. **access-list**
2. **Ctrl-A**
3. **Return** key

DETAILED STEPS

	Command or Action	Purpose
Step 1	**access-list** **Example:** `Switch(config)#` **`access-list 101 permit tcp 10.15.22.25 255.255.255.0 10.15.22.35`** `Switch(config)# $` **`101 permit tcp 10.15.22.25 255.255.255.0 10.15.22.35 255.25`** `Switch(config)# $`**`t tcp 10.15.22.25 255.255.255.0 131.108.1.20 255.255.255.0 eq`** `Switch(config)# $`**`15.22.25 255.255.255.0 10.15.22.35 255.255.255.0 eq 45`**	Displays the global configuration command entry that extends beyond one line. When the cursor first reaches the end of the line, the line is shifted ten spaces to the left and redisplayed. The dollar sign ($) shows that the line has been scrolled to the left. Each time the cursor reaches the end of the line, the line is again shifted ten spaces to the left.
Step 2	**Ctrl-A** **Example:** `Switch(config)#` **`access-list 101 permit tcp 10.15.22.25 255.255.255.0 10.15.2$`**	Checks the complete syntax. The dollar sign ($) appears at the end of the line to show that the line has been scrolled to the right.
Step 3	**Return** key	Execute the commands. The software assumes that you have a terminal screen that is 80 columns wide. If you have a different width, use the **terminal width** privileged EXEC command to set the width of your terminal. Use line wrapping with the command history feature to recall and modify previous complex command entries.

Searching and Filtering Output of show and more Commands

You can search and filter the output for **show** and **more** commands. This is useful when you need to sort through large amounts of output or if you want to exclude output that you do not need to see. Using these commands is optional.

SUMMARY STEPS

1. {**show** \| **more**} *command* \| {**begin** \| **include** \| **exclude**} *regular-expression*

DETAILED STEPS

	Command or Action	Purpose
Step 1	{**show** | **more**} *command* | {**begin** | **include** | **exclude**} *regular-expression*	Searches and filters the output.

	Command or Action	Purpose

	Example:
	`Switch# show interfaces | include protocol`
	`Vlan1 is up, line protocol is up`
	`Vlan10 is up, line protocol is down`
	`GigabitEthernet1/0/1 is up, line protocol is down`
	`GigabitEthernet1/0/2 is up, line protocol is up`
		Expressions are case sensitive. For example, if you enter **| exclude output**, the lines that contain **output** are not displayed, but the lines that contain **output** appear.

Accessing the CLI on a Switch Stack

You can access the CLI through a console connection, through Telnet, a SSH, or by using the browser.

You manage the switch stack and the stack member interfaces through the . You cannot manage stack members on an individual switch basis. You can connect to the through the console port or the Ethernet management port of one or more stack members. Be careful with using multiple CLI sessions on the . Commands that you enter in one session are not displayed in the other sessions. Therefore, it is possible to lose track of the session from which you entered commands.

Note We recommend using one CLI session when managing the switch stack.

If you want to configure a specific stack member port, you must include the stack member number in the CLI command interface notation.

Accessing the CLI Through a Console Connection or Through Telnet

Before you can access the CLI, you must connect a terminal or a PC to the switch console or connect a PC to the Ethernet management port and then power on the switch, as described in the hardware installation guide that shipped with your switch.

If your switch is already configured, you can access the CLI through a local console connection or through a remote Telnet session, but your switch must first be configured for this type of access.

You can use one of these methods to establish a connection with the switch:

- Connect the switch console port to a management station or dial-up modem, or connect the Ethernet management port to a PC. For information about connecting to the console or Ethernet management port, see the switch hardware installation guide.
- Use any Telnet TCP/IP or encrypted Secure Shell (SSH) package from a remote management station. The switch must have network connectivity with the Telnet or SSH client, and the switch must have an enable secret password configured.
  - The switch supports up to 16 simultaneous Telnet sessions. Changes made by one Telnet user are reflected in all other Telnet sessions.
  - The switch supports up to five simultaneous secure SSH sessions.

After you connect through the console port, through the Ethernet management port, through a Telnet session or through an SSH session, the user EXEC prompt appears on the management station.

PART I

System Management

CHAPTER 2

Administering the System

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information About Administering the Switch

System Time and Date Management

You can manage the system time and date on your switch using automatic configuration methods (RTC and NTP), or manual configuration methods.

System Clock

The basis of the time service is the system clock. This clock runs from the moment the system starts up and keeps track of the date and time.

The system clock can then be set from these sources:

- NTP
- Manual configuration

The system clock can provide time to these services:

- User **show** commands
- Logging and debugging messages

The system clock keeps track of time internally based on Coordinated Universal Time (UTC), also known as Greenwich Mean Time (GMT). You can configure information about the local time zone and summer time (daylight saving time) so that the time appears correctly for the local time zone.

The system clock keeps track of whether the time is *authoritative* or not (that is, whether it has been set by a time source considered to be authoritative). If it is not authoritative, the time is available only for display purposes and is not redistributed.

Network Time Protocol

The NTP is designed to time-synchronize a network of devices. NTP runs over User Datagram Protocol (UDP), which runs over IP. NTP is documented in RFC 1305.

An NTP network usually gets its time from an authoritative time source, such as a radio clock or an atomic clock attached to a time server. NTP then distributes this time across the network. NTP is extremely efficient; no more than one packet per minute is necessary to synchronize two devices to within a millisecond of one another.

NTP Stratum

NTP uses the concept of a *stratum* to describe how many NTP hops away a device is from an authoritative time source. A stratum 1 time server has a radio or atomic clock directly attached, a stratum 2 time server receives its time through NTP from a stratum 1 time server, and so on. A device running NTP automatically chooses as its time source the device with the lowest stratum number with which it communicates through NTP. This strategy effectively builds a self-organizing tree of NTP speakers.

NTP avoids synchronizing to a device whose time might not be accurate by never synchronizing to a device that is not synchronized. NTP also compares the time reported by several devices and does not synchronize to a device whose time is significantly different than the others, even if its stratum is lower.

NTP Associations

The communications between devices running NTP (known as *associations*) are usually statically configured; each device is given the IP address of all devices with which it should form associations. Accurate timekeeping is possible by exchanging NTP messages between each pair of devices with an association. However, in a LAN environment, NTP can be configured to use IP broadcast messages instead. This alternative reduces configuration complexity because each device can simply be configured to send or receive broadcast messages. However, in that case, information flow is one-way only.

NTP Security

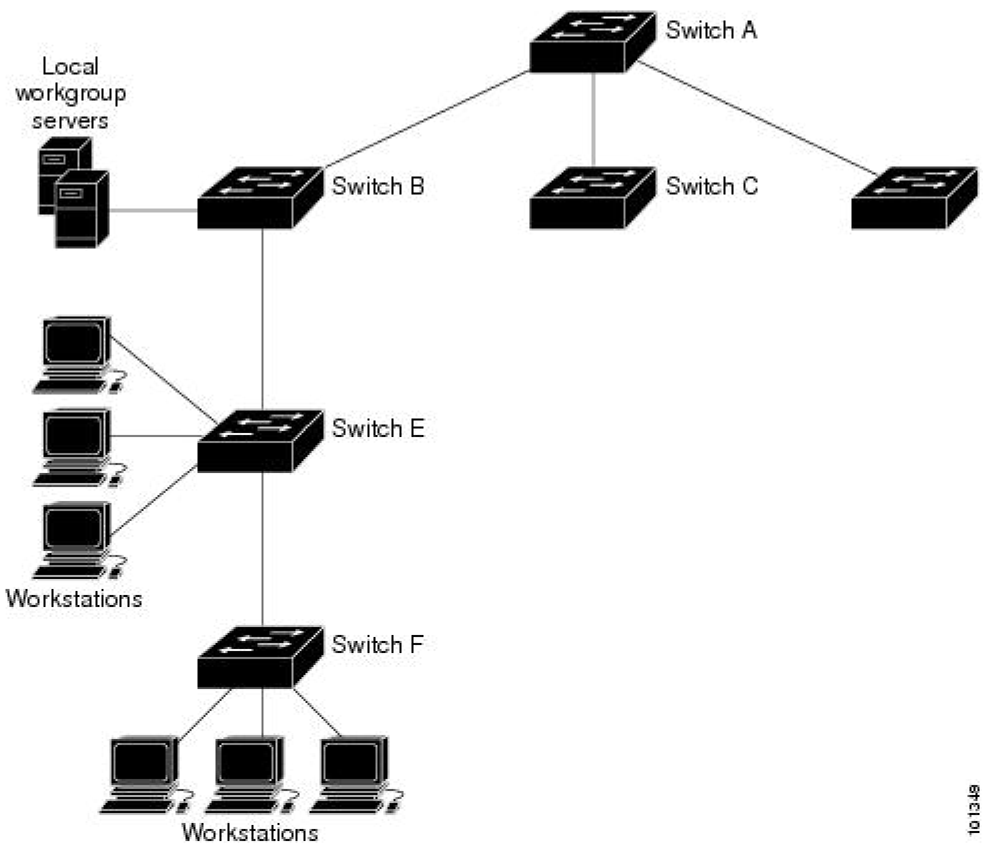
The time kept on a device is a critical resource; you should use the security features of NTP to avoid the accidental or malicious setting of an incorrect time. Two mechanisms are available: an access list-based restriction scheme and an encrypted authentication mechanism.

NTP Implementation

Implementation of NTP does not support stratum 1 service; it is not possible to connect to a radio or atomic clock. We recommend that the time service for your network be derived from the public NTP servers available on the IP Internet.

The following figure shows a typical network example using NTP. Switch A is the NTP master, with the Switch B, C, and D configured in NTP server mode, in server association with Switch A. Switch E is configured as an NTP peer to the upstream and downstream switches, Switch B and Switch F, respectively.

Figure 1: Typical NTP Network Configuration



If the network is isolated from the Internet, NTP allows a device to act as if it is synchronized through NTP, when in fact it has learned the time by using other means. Other devices then synchronize to that device through NTP.

When multiple sources of time are available, NTP is always considered to be more authoritative. NTP time overrides the time set by any other method.

Several manufacturers include NTP software for their host systems, and a publicly available version for systems running UNIX and its various derivatives is also available. This software allows host systems to be time-synchronized as well.

NTP Version 4

NTP version 4 is implemented on the switch. NTPv4 is an extension of NTP version 3. NTPv4 supports both IPv4 and IPv6 and is backward-compatible with NTPv3.

NTPv4 provides these capabilities:

- Support for IPv6.
- Improved security compared to NTPv3. The NTPv4 protocol provides a security framework based on public key cryptography and standard X509 certificates.
- Automatic calculation of the time-distribution hierarchy for a network. Using specific multicast groups, NTPv4 automatically configures the hierarchy of the servers to achieve the best time accuracy for the lowest bandwidth cost. This feature leverages site-local IPv6 multicast addresses.

System Name and Prompt

You configure the system name on the switch to identify it. By default, the system name and prompt are Switch.

If you have not configured a system prompt, the first 20 characters of the system name are used as the system prompt. A greater-than symbol [>] is appended. The prompt is updated whenever the system name changes.

Stack System Name and Prompt

If you are accessing a stack member through the , you must use the **session** *stack-member-number* privileged EXEC command. The stack member number range is from 1 through 4. When you use this command, the stack member number is appended to the system prompt. For example, Switch-2# is the prompt in privileged EXEC mode for stack member 2, and the system prompt for the switch stack is Switch.

Default System Name and Prompt Configuration

The default switch system name and prompt is *Switch*.

DNS

The DNS protocol controls the Domain Name System (DNS), a distributed database with which you can map hostnames to IP addresses. When you configure DNS on your switch, you can substitute the hostname for the IP address with all IP commands, such as **ping**, **telnet**, **connect**, and related Telnet support operations.

IP defines a hierarchical naming scheme that allows a device to be identified by its location or domain. Domain names are pieced together with periods (.) as the delimiting characters. For example, Cisco Systems is a commercial organization that IP identifies by a *com* domain name, so its domain name is *cisco.com*. A specific device in this domain, for example, the File Transfer Protocol (FTP) system is identified as *ftp.cisco.com*.

To keep track of domain names, IP has defined the concept of a domain name server, which holds a cache (or database) of names mapped to IP addresses. To map domain names to IP addresses, you must first identify the hostnames, specify the name server that is present on your network, and enable the DNS.

Default DNS Settings

Table 4: Default DNS Settings

Feature	Default Setting
DNS enable state	Enabled.
DNS default domain name	None configured.
DNS servers	No name server addresses are configured.

Login Banners

You can configure a message-of-the-day (MOTD) and a login banner. The MOTD banner is displayed on all connected terminals at login and is useful for sending messages that affect all network users (such as impending system shutdowns).

The login banner is also displayed on all connected terminals. It appears after the MOTD banner and before the login prompts.

The MOTD and login banners are not configured.

Default Banner Configuration

The MOTD and login banners are not configured.

MAC Address Table

The MAC address table contains address information that the switch uses to forward traffic between ports. All MAC addresses in the address table are associated with one or more ports. The address table includes these types of addresses:

- Dynamic address—A source MAC address that the switch learns and then ages when it is not in use.
- Static address—A manually entered unicast address that does not age and that is not lost when the switch resets.

The address table lists the destination MAC address, the associated VLAN ID, and port number associated with the address and the type (static or dynamic).

MAC Address Table Creation

With multiple MAC addresses supported on all ports, you can connect any port on the switch to other network devices. The switch provides dynamic addressing by learning the source address of packets it receives on each port and adding the address and its associated port number to the address table. As devices are added or removed from the network, the switch updates the address table, adding new dynamic addresses and aging out those that are not in use.

The aging interval is globally configured. However, the switch maintains an address table for each VLAN, and STP can accelerate the aging interval on a per-VLAN basis.

The switch sends packets between any combination of ports, based on the destination address of the received packet. Using the MAC address table, the switch forwards the packet only to the port associated with the destination address. If the destination address is on the port that sent the packet, the packet is filtered and not forwarded. The switch always uses the store-and-forward method: complete packets are stored and checked for errors before transmission.

MAC Addresses and VLANs

All addresses are associated with a VLAN. An address can exist in more than one VLAN and have different destinations in each. Unicast addresses, for example, could be forwarded to port 1 in VLAN 1 and ports 9, 10, and 1 in VLAN 5.

Each VLAN maintains its own logical address table. A known address in one VLAN is unknown in another until it is learned or statically associated with a port in the other VLAN.

MAC Addresses and Switch Stacks

The MAC address tables on all stack members are synchronized. At any given time, each stack member has the same copy of the address tables for each VLAN. When an address ages out, the address is removed from the address tables on all stack members. When a switch joins a switch stack, that switch receives the addresses for each VLAN learned on the other stack members. When a stack member leaves the switch stack, the remaining stack members age out or remove all addresses learned by the former stack member.

Default MAC Address Table Settings

The following table shows the default settings for the MAC address table.

Table 5: Default Settings for the MAC Address

Feature	Default Setting
Aging time	300 seconds
Dynamic addresses	Automatically learned
Static addresses	None configured

ARP Table Management

To communicate with a device (over Ethernet, for example), the software first must learn the 48-bit MAC address or the local data link address of that device. The process of learning the local data link address from an IP address is called address resolution.

The Address Resolution Protocol (ARP) associates a host IP address with the corresponding media or MAC addresses and the VLAN ID. Using an IP address, ARP finds the associated MAC address. When a MAC address is found, the IP-MAC address association is stored in an ARP cache for rapid retrieval. Then the IP datagram is encapsulated in a link-layer frame and sent over the network. Encapsulation of IP datagrams and ARP requests and replies on IEEE 802 networks other than Ethernet is specified by the Subnetwork Access Protocol (SNAP). By default, standard Ethernet-style ARP encapsulation (represented by the **arpa** keyword) is enabled on the IP interface.

ARP entries added manually to the table do not age and must be manually removed.

How to Administer the Switch

Configuring the Time and Date Manually

System time remains accurate through restarts and reboot, however, you can manually configure the time and date after the system is restarted.

We recommend that you use manual configuration only when necessary. If you have an outside source to which the switch can synchronize, you do not need to manually set the system clock.

Note You must reconfigure this setting if you have manually configured the system clock before the fails and a different stack member assumes the role of .

Setting the System Clock

If you have an outside source on the network that provides time services, such as an NTP server, you do not need to manually set the system clock.

SUMMARY STEPS

1. Use one of the following:
   - **clock set** *hh:mm:ss day month year*
   - **clock set** *hh:mm:ss month day year*

DETAILED STEPS

	Command or Action	Purpose
Step 1	Use one of the following: • **clock set** *hh:mm:ss day month year* • **clock set** *hh:mm:ss month day year*	Sets the system clock using one of these formats: • *hh:mm:ss*—Specifies the time in hours (24-hour format), minutes, and seconds. The time specified is relative to the configured time zone.

	Command or Action	Purpose
	Example: `Switch# clock set 13:32:00 23 March 2013`	• *day*—Specifies the day by date in the month. • *month*—Specifies the month by name. • *year*—Specifies the year (no abbreviation).

Configuring the Time Zone

SUMMARY STEPS

1. **configure terminal**
2. **clock timezone** *zone hours-offset* [*minutes-offset*]
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**clock timezone** *zone hours-offset* [*minutes-offset*] **Example:** `Switch(config)# clock timezone AST -3 30`	Sets the time zone. Internal time is kept in Coordinated Universal Time (UTC), so this command is used only for display purposes and when the time is manually set. • *zone*—Enters the name of the time zone to be displayed when standard time is in effect. The default is UTC. • *hours-offset*—Enters the hours offset from UTC. • (Optional) *minutes-offset*—Enters the minutes offset from UTC. This available where the local time zone is a percentage of an hour different from UTC.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring Summer Time (Daylight Saving Time)

To configure summer time (daylight saving time) in areas where it starts and ends on a particular day of the week each year, perform this task:

SUMMARY STEPS

1. **configure terminal**
2. **clock summer-time** *zone* **date** *date month year hh:mm date month year hh:mm* [*offset*]]
3. **clock summer-time** *zone* **recurring** [*week day month hh:mm week day month hh:mm* [*offset*]]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

Step 2	**clock summer-time** *zone* **date** *date month year hh:mm date month year hh:mm* [*offset*]]
	
	Example:
	`Switch(config)# clock summer-time PDT date`
	`10 March 2013 2:00 3 November 2013 2:00`
		Configures summer time to start and end on specified days every year.

Step 3	**clock summer-time** *zone* **recurring** [*week day month hh:mm week day month hh:mm* [*offset*]] **Example:** `Switch(config)# clock summer-time PDT recurring 10 March 2013 2:00 3 November 2013 2:00`	Configures summer time to start and end on the specified days every year. All times are relative to the local time zone. The start time is relative to standard time. The end time is relative to summer time. Summer time is disabled by default. If you specify **clock summer-time** *zone* **recurring** without parameters, the summer time rules default to the United States rules. If the starting month is after the ending month, the system assumes that you are in the southern hemisphere. • *zone*—Specifies the name of the time zone (for example, PDT) to be displayed when summer time is in effect. • (Optional) *week*— Specifies the week of the month (1 to 4, **first**, or **last**). • (Optional) *day*—Specifies the day of the week (Sunday, Monday...). • (Optional) *month*—Specifies the month (January, February...).

	Command or Action	Purpose
		• (Optional) *hh:mm*—Specifies the time (24-hour format) in hours and minutes. • (Optional) *offset*—Specifies the number of minutes to add during summer time. The default is 60.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring a System Name

SUMMARY STEPS

1. **configure terminal**
2. **hostname** *name*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**hostname** *name* **Example:** `Switch(config)# hostname remote-users`	Configures a system name. When you set the system name, it is also used as the system prompt. The default setting is Switch. The name must follow the rules for ARPANET hostnames. They must start with a letter, end with a letter or digit, and have as interior characters only letters, digits, and hyphens. Names can be up to 63 characters.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Setting Up DNS

If you use the switch IP address as its hostname, the IP address is used and no DNS query occurs. If you configure a hostname that contains no periods (.), a period followed by the default domain name is appended to the hostname before the DNS query is made to map the name to an IP address. The default domain name is the value set by the **ip domain-name** global configuration command. If there is a period (.) in the hostname, the Cisco IOS software looks up the IP address without appending any default domain name to the hostname.

SUMMARY STEPS

1. **configure terminal**
2. **ip domain-name** *name*
3. **ip name-server** *server-address1* [*server-address2 ... server-address6*]
4. **ip domain-lookup** [**nsap** | **source-interface** *interface*]
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip domain-name** *name* **Example:** `Switch(config)# ip domain-name Cisco.com`	Defines a default domain name that the software uses to complete unqualified hostnames (names without a dotted-decimal domain name). Do not include the initial period that separates an unqualified name from the domain name. At boot time, no domain name is configured; however, if the switch configuration comes from a BOOTP or Dynamic Host Configuration Protocol (DHCP) server, then the default domain name might be set by the BOOTP or DHCP server (if the servers were configured with this information).
Step 3	**ip name-server** *server-address1* [*server-address2 ... server-address6*] **Example:** `Switch(config)# ip name-server 192.168.1.100 192.168.1.200 192.168.1.300`	Specifies the address of one or more name servers to use for name and address resolution. You can specify up to six name servers. Separate each server address with a space. The first server specified is the primary server. The switch sends DNS queries to the primary server first. If that query fails, the backup servers are queried.
Step 4	**ip domain-lookup** [**nsap** | **source-interface** *interface*]	(Optional) Enables DNS-based hostname-to-address translation on your switch. This feature is enabled by default.

	Command or Action	Purpose
	Example: `Switch(config)# ip domain-lookup`	If your network devices require connectivity with devices in networks for which you do not control name assignment, you can dynamically assign device names that uniquely identify your devices by using the global Internet naming scheme (DNS).
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring a Message-of-the-Day Login Banner

You can create a single or multiline message banner that appears on the screen when someone logs in to the switch

SUMMARY STEPS

1. **configure terminal**
2. **banner motd** *c message c*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

Step 2	**banner motd** *c message c*
	Example:
	`Switch(config)# banner motd #`
	`This is a secure site. Only`
	`authorized users are allowed.`
	`For access, contact technical`
	`support.`
	`#`
		Specifies the message of the day.
		c—Enters the delimiting character of your choice, for example, a pound sign (#), and press the **Return** key. The delimiting character signifies the beginning and end of the banner text. Characters after the ending delimiter are discarded.
		message—Enters a banner message up to 255 characters. You cannot use the delimiting character in the message.

	Command or Action	Purpose
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring a Login Banner

You can configure a login banner to be displayed on all connected terminals. This banner appears after the MOTD banner and before the login prompt.

SUMMARY STEPS

1. **configure terminal**
2. **banner login** *c message c*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

Step 2	**banner login** *c message c*
	
	Example:
	
	`Switch(config)# banner login $`
	`Access for authorized users only.`
	`Please enter your username and`
	`password.`
	`$`
		Specifies the login message.
		
		c— Enters the delimiting character of your choice, for example, a pound sign (#), and press the **Return** key. The delimiting character signifies the beginning and end of the banner text. Characters after the ending delimiter are discarded.
		
		message—Enters a login message up to 255 characters. You cannot use the delimiting character in the message.

Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Managing the MAC Address Table

Changing the Address Aging Time

SUMMARY STEPS

1. **configure terminal**
2. **mac address-table aging-time** [*0* | *10-1000000*] [**routed-mac** | **vlan** *vlan-id*]
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**mac address-table aging-time** [*0* | *10-1000000*] [**routed-mac** | **vlan** *vlan-id*] **Example:** `Switch(config)# mac address-table aging-time 500 vlan 2`	Sets the length of time that a dynamic entry remains in the MAC address table after the entry is used or updated. The range is 10 to 1000000 seconds. The default is 300. You can also enter 0, which disables aging. Static address entries are never aged or removed from the table. *vlan-id*—Valid IDs are 1 to 4094.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring MAC Address Change Notification Traps

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server host** *host-addr community-string notification-type* { **informs** | **traps** } {**version** {**1** | **2c** | **3**}} {**vrf** *vrf instance name*}
3. **snmp-server enable traps mac-notification change**
4. **mac address-table notification change**
5. **mac address-table notification change** [**interval** *value*] [**history-size** *value*]
6. **interface** *interface-id*
7. **snmp trap mac-notification change** {**added** | **removed**}
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch#` **`configure terminal`**	Enters global configuration mode.
Step 2	**snmp-server host** *host-addr community-string notification-type* { **informs** | **traps** } {**version** {**1** | **2c** | **3**}} {**vrf** *vrf instance name*} **Example:** `Switch(config)#` **`snmp-server host 172.20.10.10 traps private mac-notification`**	Specifies the recipient of the trap message. • *host-addr*—Specifies the name or address of the NMS. • **traps** (the default)—Sends SNMP traps to the host. • **informs**—Sends SNMP informs to the host. • **version**—Specifies the SNMP version to support. Version 1, the default, is not available with informs. • *community-string*—Specifies the string to send with the notification operation. Though you can set this string by using the **snmp-server host** command, we recommend that you define this string by using the **snmp-server community** command before using the **snmp-server host** command. • *notification-type*—Uses the **mac-notification** keyword. • **vrf** *vrf instance name*—Specifies the VPN routing/forwarding instance for this host.

	Command or Action	Purpose
Step 3	**snmp-server enable traps mac-notification change** **Example:** `Switch(config)# snmp-server enable traps mac-notification change`	Enables the switch to send MAC address change notification traps to the NMS.
Step 4	**mac address-table notification change** **Example:** `Switch(config)# mac address-table notification change`	Enables the MAC address change notification feature.

Step 5	**mac address-table notification change** [**interval** *value*] [**history-size** *value*]
	
	Example:
	
	`Switch(config)# mac address-table notification change interval 123`
	`Switch(config)#mac address-table notification change history-size 100`
		Enters the trap interval time and the history table size.
		
		• (Optional) **interval** *value*—Specifies the notification trap interval in seconds between each set of traps that are generated to the NMS. The range is 0 to 2147483647 seconds; the default is 1 second.
		
		• (Optional) **history-size** *value*—Specifies the maximum number of entries in the MAC notification history table. The range is 0 to 500; the default is 1.

Step 6	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/2`	Enters interface configuration mode, and specifies the Layer 2 interface on which to enable the SNMP MAC address notification trap.
Step 7	**snmp trap mac-notification change** {**added** | **removed**} **Example:** `Switch(config-if)# snmp trap mac-notification change added`	Enables the MAC address change notification trap on the interface. • Enables the trap when a MAC address is **added** on this interface. • Enables the trap when a MAC address is **removed** from this interface.
Step 8	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Configuring MAC Address Move Notification Traps

When you configure MAC-move notification, an SNMP notification is generated and sent to the network management system whenever a MAC address moves from one port to another within the same VLAN.

Beginning in privileged EXEC mode, follow these steps to configure the switch to send MAC address-move notification traps to an NMS host:

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server host** *host-addr* {**traps** | **informs**} {**version** {**1** | **2c** | **3**}} *community-string notification-type*
3. **snmp-server enable traps mac-notification move**
4. **mac address-table notification mac-move**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**snmp-server host** *host-addr* {**traps** | **informs**} {**version** {**1** | **2c** | **3**}} *community-string notification-type* **Example:** `Switch(config)# snmp-server host 172.20.10.10 traps private mac-notification`	Specifies the recipient of the trap message. • *host-addr*—Specifies the name or address of the NMS. • **traps** (the default)—Sends SNMP traps to the host. • **informs**—Sends SNMP informs to the host. • **version**—Specifies the SNMP version to support. Version 1, the default, is not available with informs. • *community-string*—Specifies the string to send with the notification operation. Though you can set this string by using the **snmp-server host** command, we recommend that you define this string by using the **snmp-server community** command before using the **snmp-server host** command. • *notification-type*—Uses the **mac-notification** keyword.
Step 3	**snmp-server enable traps mac-notification move** **Example:** `Switch(config)# snmp-server enable traps mac-notification move`	Enables the switch to send MAC address move notification traps to the NMS.

	Command or Action	Purpose
Step 4	**mac address-table notification mac-move** **Example:** `Switch(config)# mac address-table notification mac-move`	Enables the MAC address move notification feature.
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring MAC Threshold Notification Traps

When you configure MAC threshold notification, an SNMP notification is generated and sent to the network management system when a MAC address table threshold limit is reached or exceeded.

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server host** *host-addr* {**traps** | **informs**} {**version** {**1** | **2c** | **3**}} *community-string notification-type*
3. **snmp-server enable traps mac-notification threshold**
4. **mac address-table notification threshold**
5. **mac address-table notification threshold** [**limit** *percentage*] | [**interval** *time*]
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**snmp-server host** *host-addr* {**traps** | **informs**} {**version** {**1** | **2c** | **3**}} *community-string notification-type* **Example:** `Switch(config)# snmp-server host`	Specifies the recipient of the trap message. • *host-addr*—Specifies the name or address of the NMS. • **traps** (the default)—Sends SNMP traps to the host. • **informs**—Sends SNMP informs to the host.

	Command or Action	Purpose
	`172.20.10.10 traps private mac-notification`	• **version**—Specifies the SNMP version to support. Version 1, the default, is not available with informs. • *community-string*—Specifies the string to send with the notification operation. You can set this string by using the **snmp-server host** command, but we recommend that you define this string by using the **snmp-server community** command before using the **snmp-server host** command. • *notification-type*—Uses the **mac-notification** keyword.
Step 3	**snmp-server enable traps mac-notification threshold** **Example:** `Switch(config)# snmp-server enable traps mac-notification threshold`	Enables MAC threshold notification traps to the NMS.
Step 4	**mac address-table notification threshold** **Example:** `Switch(config)# mac address-table notification threshold`	Enables the MAC address threshold notification feature.

Step 5	**mac address-table notification threshold** [**limit** *percentage*] | [**interval** *time*]
	
	Example:
	
	`Switch(config)# mac address-table notification threshold interval 123`
	`Switch(config)# mac address-table notification threshold limit 78`
		Enters the threshold value for the MAC address threshold usage monitoring.
		• (Optional) **limit** *percentage*—Specifies the percentage of the MAC address table use; valid values are from 1 to 100 percent. The default is 50 percent.
		• (Optional) **interval** *time*—Specifies the time between notifications; valid values are greater than or equal to 120 seconds. The default is 120 seconds.

Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Adding and Removing Static Address Entries

SUMMARY STEPS

1. **configure terminal**
2. **mac address-table static** *mac-addr* **vlan** *vlan-id* **interface** *interface-id*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**mac address-table static** *mac-addr* **vlan** *vlan-id* **interface** *interface-id* **Example:** `Switch(config)# mac address-table static c2f3.220a.12f4 vlan 4 interface gigabitethernet 1/0/1`	Adds a static address to the MAC address table. • *mac-addr*—Specifies the destination MAC unicast address to add to the address table. Packets with this destination address received in the specified VLAN are forwarded to the specified interface. • *vlan-id*—Specifies the VLAN for which the packet with the specified MAC address is received. Valid VLAN IDs are 1 to 4094. • *interface-id*—Specifies the interface to which the received packet is forwarded. Valid interfaces include physical ports or port channels. For static multicast addresses, you can enter multiple interface IDs. For static unicast addresses, you can enter only one interface at a time, but you can enter the command multiple times with the same MAC address and VLAN ID.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Unicast MAC Address Filtering

SUMMARY STEPS

1. **configure terminal**
2. **mac address-table static** *mac-addr* **vlan** *vlan-id* **drop**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**mac address-table static** *mac-addr* **vlan** *vlan-id* **drop** **Example:** `Switch(config)# mac address-table static c2f3.220a.12f4 vlan 4 drop`	Enables unicast MAC address filtering and configure the switch to drop a packet with the specified source or destination unicast static address. • *mac-addr*—Specifies a source or destination unicast MAC address (48-bit). Packets with this MAC address are dropped. • *vlan-id*—Specifies the VLAN for which the packet with the specified MAC address is received. Valid VLAN IDs are 1 to 4094.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Monitoring and Maintaining Administration of the Switch

Command	Purpose
clear mac address-table dynamic	Removes all dynamic entries.
clear mac address-table dynamic address *mac-address*	Removes a specific MAC address.
clear mac address-table dynamic interface *interface-id*	Removes all addresses on the specified physical port or port channel.
clear mac address-table dynamic vlan *vlan-id*	Removes all addresses on a specified VLAN.
show clock [*detail*]	Displays the time and date configuration.
show ip igmp snooping groups	Displays the Layer 2 multicast entries for all VLANs or the specified VLAN.

Command	Purpose
show mac address-table address *mac-address*	Displays MAC address table information for the specified MAC address.
show mac address-table aging-time	Displays the aging time in all VLANs or the specified VLAN.
show mac address-table count	Displays the number of addresses present in all VLANs or the specified VLAN.
show mac address-table dynamic	Displays only dynamic MAC address table entries.
show mac address-table interface *interface-name*	Displays the MAC address table information for the specified interface.
show mac address-table move update	Displays the MAC address table move update information.
show mac address-table multicast	Displays a list of multicast MAC addresses.
show mac address-table notification {change | mac-move | threshold}	Displays the MAC notification parameters and history table.
show mac address-table secure	Displays the secure MAC addresses.
show mac address-table static	Displays only static MAC address table entries.
show mac address-table vlan *vlan-id*	Displays the MAC address table information for the specified VLAN.

Configuration Examples for Switch Administration

Example: Setting the System Clock

This example shows how to manually set the system clock:

```
Switch# clock set 13:32:00 23 July 2013
```

Examples: Configuring Summer Time

This example (for daylight savings time) shows how to specify that summer time starts on March 10 at 02:00 and ends on November 3 at 02:00:

```
Switch(config)# clock summer-time PDT recurring PST date
```

```
10 March 2013 2:00 3 November 2013 2:00
```

This example shows how to set summer time start and end dates:

```
Switch(config)#clock summer-time PST date
20 March 2013 2:00 20 November 2013 2:00
```

Example: Configuring a MOTD Banner

This example shows how to configure a MOTD banner by using the pound sign (#) symbol as the beginning and ending delimiter:

```
Switch(config)# banner motd #

This is a secure site. Only authorized users are allowed.
For access, contact technical support.

#

Switch(config)#
```

This example shows the banner that appears from the previous configuration:

```
Unix> telnet 192.0.2.15

Trying 192.0.2.15...

Connected to 192.0.2.15.

Escape character is '^]'.

This is a secure site. Only authorized users are allowed.

For access, contact technical support.

User Access Verification

Password:
```

Example: Configuring a Login Banner

This example shows how to configure a login banner by using the dollar sign ($) symbol as the beginning and ending delimiter:

```
Switch(config)# banner login $

Access for authorized users only. Please enter your username and password.

$

Switch(config)#
```

Example: Configuring MAC Address Change Notification Traps

This example shows how to specify 172.20.10.10 as the NMS, enable MAC address notification traps to the NMS, enable the MAC address-change notification feature, set the interval time to 123 seconds, set the history-size to 100 entries, and enable traps whenever a MAC address is added on the specified port:

```
Switch(config)# snmp-server host 172.20.10.10 traps private mac-notification
Switch(config)# snmp-server enable traps mac-notification change
Switch(config)# mac address-table notification change
Switch(config)# mac address-table notification change interval 123
Switch(config)# mac address-table notification change history-size 100
Switch(config)# interface gigabitethernet1/2/1
Switch(config-if)# snmp trap mac-notification change added
```

Example: Configuring MAC Threshold Notification Traps

This example shows how to specify 172.20.10.10 as the NMS, enable the MAC address threshold notification feature, set the interval time to 123 seconds, and set the limit to 78 per cent:

```
Switch(config)# snmp-server host 172.20.10.10 traps private mac-notification
Switch(config)# snmp-server enable traps mac-notification threshold
Switch(config)# mac address-table notification threshold
Switch(config)# mac address-table notification threshold interval 123
Switch(config)# mac address-table notification threshold limit 78
```

Example: Adding the Static Address to the MAC Address Table

This example shows how to add the static address c2f3.220a.12f4 to the MAC address table. When a packet is received in VLAN 4 with this MAC address as its destination address, the packet is forwarded to the specified port:

```
Switch(config)# mac address-table static c2f3.220a.12f4 vlan 4 interface gigabitethernet1/1/1
```

Example: Configuring Unicast MAC Address Filtering

This example shows how to enable unicast MAC address filtering and how to configure drop packets that have a source or destination address of c2f3.220a.12f4. When a packet is received in VLAN 4 with this MAC address as its source or destination, the packet is dropped:

```
Switch(config)# mac address-table static c2f3.220a.12f4 vlan 4 drop
```

Additional References for Switch Administration

Related Documents

Related Topic	Document Title
System management commands	*System Management Command Reference (Catalyst 3850 Switches)*
Network management configuration	*Network Management Configuration Guide (Catalyst 3850 Switches)*
Layer 2 configuration	*Layer 2/3 Configuration Guide (Catalyst 3850 Switches)*
VLAN configuration	*VLAN Configuration Guide (Catalyst 3850 Switches)*
Platform-independent command references	*Configuration Fundamentals Command Reference, Cisco IOS XE Release 3S (Catalyst 3850 Switches)*
Platform-independent configuration information	*Configuration Fundamentals Configuration Guide, Cisco IOS XE Release 3S (Catalyst 3850 Switches)* *IP Addressing Configuration Guide Library, Cisco IOS XE Release 3S (Catalyst 3850 Switches)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Switch Administration

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 3

Performing Switch Setup Configuration

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information About Performing Switch Setup Configuration

Review the sections in this module before performing your initial switch configuration tasks that include IP address assignments and DHCP autoconfiguration.

Switch Boot Process

To start your switch, you need to follow the procedures in the hardware installation guide for installing and powering on the switch and setting up the initial switch configuration (IP address, subnet mask, default gateway, secret and Telnet passwords, and so forth).

The normal boot process involves the operation of the boot loader software and includes these activities:

- Locates the bootable (base) package in the bundle or installed package set.
- Performs low-level CPU initialization. It initializes the CPU registers, which control where physical memory is mapped, its quantity, its speed, and so forth.
- Performs power-on self-test (POST) for the CPU subsystem and tests the system DRAM.
- Initializes the file systems on the system board.
- Loads a default operating system software image into memory and boots up the switch.

The boot loader provides access to the file systems before the operating system is loaded. Normally, the boot loader is used only to load, decompress, and start the operating system. After the boot loader gives the operating system control of the CPU, the boot loader is not active until the next system reset or power-on.

The boot loader also provides trap-door access into the system if the operating system has problems serious enough that it cannot be used. The trap-door mechanism provides enough access to the system so that if it is necessary, you can reinstall the operating system software image by using the **emergency-install** command and restart the operating system.

Before you can assign switch information, make sure you have connected a PC or terminal to the console port or a PC to the Ethernet management port, and make sure you have configured the PC or terminal-emulation software baud rate and character format to match these of the switch console port:

- Baud rate default is 9600.
- Data bits default is 8.

Note If the data bits option is set to 8, set the parity option to none.

- Stop bits default is 2 (minor).
- Parity settings default is none.

Software Installer Features

The following software installer features are supported on your switch:

- Software bundle installation on a standalone switch, a switch stack, or a subset of switches in a stack. The default is installation on all the switches if a switch stack is configured.
- In a stack of switches, Cisco recommends all switches in install mode.
- Software rollback to a previously installed package set.
- Emergency installation in the event that no valid installed packages reside on the boot flash.
- Auto-upgrade of a switch that joins the switch stack with incompatible software.
- Installation using packages on one switch as the source for installing packages on another switch in the switch stack.

Note Software installation and rollback must be performed while running only in installed mode. You can use the **software expand** EXEC command to convert bundle boot mode to install mode.

Software Boot Modes

Your switch supports two modes to boot the software packages:

- Installed mode
- Bundle mode

Related Topics

Examples: Displaying Software Bootup in Install Mode, on page 72

Example: Emergency Installation, on page 73

Installed Boot Mode

You can boot your switch in installed mode by booting the software package provisioning file that resides in flash:

```
switch: boot flash:packages.conf
```

The provisioning file contains a list of software packages to boot, mount, and run. The ISO file system in each installed package is mounted to the root file system directly from flash.

Note The packages and provisioning file used to boot in installed mode must reside in flash. Booting in installed mode from usbflash0: or tftp: is not supported.

Related Topics

Examples: Displaying Software Bootup in Install Mode, on page 72

Example: Emergency Installation, on page 73

Bundle Boot Mode

You can boot your switch in bundle boot mode by booting the bundle (.bin) file:

```
switch:  boot flash:cat3850-universalk9.SSA.03.08.83.EMD.150-8.83.EMD.bin
```

The provisioning file contained in a bundle is used to decide which packages to boot, mount, and run. Packages are extracted from the bundle and copied to RAM. The ISO file system in each package is mounted to the root file system.

Unlike install boot mode, additional memory that is equivalent to the size of the bundle is used when booting in bundle mode.

Unlike install boot mode, bundle boot mode is available from several locations:

- flash:

- usbflash0:
- tftp:

Note Auto install and smart install functionality is not supported in bundle boot mode.

Note The AP image pre-download feature is not supported in bundle boot mode. For more information about the pre-download feature see the Cisco WLC 5700 Series *Preloading an Image to Access Points* chapter.

Related Topics

Examples: Displaying Software Bootup in Install Mode, on page 72

Example: Emergency Installation, on page 73

Boot Mode for a Switch Stack

All the switches in a stack must be running in installed mode or bundle boot mode. A mixed mode stack is not supported. If a new switch tries to join the stack in a different boot mode then the active switch, the new switch is given a V-mismatch state.

If a mixed mode switch stack is booted at the same time, then all the switches except for the active switch is given a V-mismatch state. If the boot mode does not support auto-upgrade, then the switch stack members must be re-booted in the same boot mode as the active switch.

If the stack is running in installed mode, the auto-upgrade feature can be used to automatically upgrade the new switch that is attempting to join the switch stack.

The auto-upgrade feature changes the boot mode of the new switch to installed mode. If the stack is running in bundle boot mode, the auto-upgrade feature is not available. You will be required to use the bundle mode to boot the new switch so that it can join the switch stack.

This is an example of the state of a switch that attempts to join the switch stack when the boot mode is not compatible with the active switch:

```
Switch# show switch

Switch/Stack Mac Address : 6400.f125.1100 - Local Mac Address
Mac persistency wait time: Indefinite
H/W Current
Switch#   Role   Mac Address    Priority Version   State
------------------------------------------------------------
 1       Member 6400 f125.1a00   1        0        V-Mismatch
*2       Active 6400.f125.1100   1        V01      Ready
Switch
```

Switches Information Assignment

You can assign IP information through the switch setup program, through a DHCP server, or manually.

Use the switch setup program if you want to be prompted for specific IP information. With this program, you can also configure a hostname and an enable secret password.

It gives you the option of assigning a Telnet password (to provide security during remote management) and configuring your switch as a command or member switch of a cluster or as a standalone switch.

The switch stack is managed through a single IP address. The IP address is a system-level setting and is not specific to the stack master or to any other stack member. You can still manage the stack through the same IP address even if you remove the stack master or any other stack member from the stack, provided there is IP connectivity.

Note Stack members retain their IP address when you remove them from a switch stack. To avoid a conflict by having two devices with the same IP address in your network, change the IP address of the switch that you removed from the switch stack.

Use a DHCP server for centralized control and automatic assignment of IP information after the server is configured.

Note If you are using DHCP, do not respond to any of the questions in the setup program until the switch receives the dynamically assigned IP address and reads the configuration file.

If you are an experienced user familiar with the switch configuration steps, manually configure the switch. Otherwise, use the setup program described in the *Boot Process* section.

Default Switch Information

Table 6: Default Switch Information

Feature	Default Setting
IP address and subnet mask	No IP address or subnet mask are defined.
Default gateway	No default gateway is defined.
Enable secret password	No password is defined.
Hostname	The factory-assigned default hostname is Switch.
Telnet password	No password is defined.

DHCP-Based Autoconfiguration Overview

DHCP provides configuration information to Internet hosts and internetworking devices. This protocol consists of two components: one for delivering configuration parameters from a DHCP server to a device and an operation for allocating network addresses to devices. DHCP is built on a client-server model, in which designated DHCP servers allocate network addresses and deliver configuration parameters to dynamically configured devices. The switch can act as both a DHCP client and a DHCP server.

During DHCP-based autoconfiguration, your switch (DHCP client) is automatically configured at startup with IP address information and a configuration file.

With DHCP-based autoconfiguration, no DHCP client-side configuration is needed on your switch. However, you need to configure the DHCP server for various lease options associated with IP addresses.

If you want to use DHCP client autoconfiguration, you need to configure a Trivial File Transfer Protocol (TFTP) server to fetch the configuration file. The DHCP client then applies the new configuration file to its running configuration.

Note If the new configuration is downloaded to a switch that already has a configuration, the downloaded configuration is appended to the configuration file stored on the switch. (Any existing configuration is not overwritten by the downloaded one.)

Note We recommend a redundant connection between a switch stack and the DHCP, DNS, and TFTP servers. This is to help ensure that these servers remain accessible in case one of the connected stack members is removed from the switch stack.

The DHCP server for your switch can be on the same LAN or on a different LAN than the switch. If the DHCP server is running on a different LAN, you should configure a DHCP relay device between your switch and the DHCP server. A relay device forwards broadcast traffic between two directly connected LANs. A router does not forward broadcast packets, but it forwards packets based on the destination IP address in the received packet.

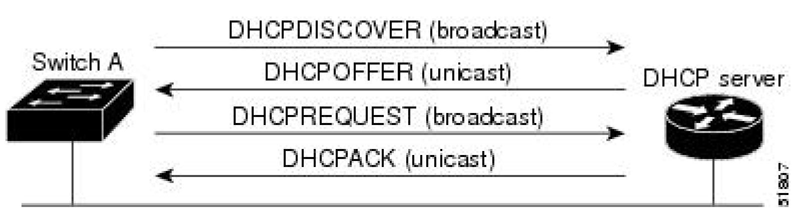
DHCP-based autoconfiguration replaces the BOOTP client functionality on your switch.

DHCP Client Request Process

When you boot up your switch, the DHCP client is invoked and requests configuration information from a DHCP server when the configuration file is not present on the switch. If the configuration file is present and the configuration includes the **ip address dhcp** interface configuration command on specific routed interfaces, the DHCP client is invoked and requests the IP address information for those interfaces.

This is the sequence of messages that are exchanged between the DHCP client and the DHCP server.

Figure 2: DHCP Client and Server Message Exchange



The client, Switch A, broadcasts a DHCPDISCOVER message to locate a DHCP server. The DHCP server offers configuration parameters (such as an IP address, subnet mask, gateway IP address, DNS IP address, a lease for the IP address, and so forth) to the client in a DHCPOFFER unicast message.

In a DHCPREQUEST broadcast message, the client returns a formal request for the offered configuration information to the DHCP server. The formal request is broadcast so that all other DHCP servers that received the DHCPDISCOVER broadcast message from the client can reclaim the IP addresses that they offered to the client.

The DHCP server confirms that the IP address has been allocated to the client by returning a DHCPACK unicast message to the client. With this message, the client and server are bound, and the client uses configuration information received from the server. The amount of information the switch receives depends on how you configure the DHCP server.

If the configuration parameters sent to the client in the DHCPOFFER unicast message are invalid (a configuration error exists), the client returns a DHCPDECLINE broadcast message to the DHCP server.

The DHCP server sends the client a DHCPNAK denial broadcast message, which means that the offered configuration parameters have not been assigned, that an error has occurred during the negotiation of the parameters, or that the client has been slow in responding to the DHCPOFFER message (the DHCP server assigned the parameters to another client).

A DHCP client might receive offers from multiple DHCP or BOOTP servers and can accept any of the offers; however, the client usually accepts the first offer it receives. The offer from the DHCP server is not a guarantee that the IP address is allocated to the client; however, the server usually reserves the address until the client has had a chance to formally request the address. If the switch accepts replies from a BOOTP server and configures itself, the switch broadcasts, instead of unicasts, TFTP requests to obtain the switch configuration file.

The DHCP hostname option allows a group of switches to obtain hostnames and a standard configuration from the central management DHCP server. A client (switch) includes in its DCHPDISCOVER message an option 12 field used to request a hostname and other configuration parameters from the DHCP server. The configuration files on all clients are identical except for their DHCP-obtained hostnames.

If a client has a default hostname (the **hostname** *name* global configuration command is not configured or the **no hostname** global configuration command is entered to remove the hostname), the DHCP hostname option is not included in the packet when you enter the **ip address dhcp** interface configuration command. In this case, if the client receives the DCHP hostname option from the DHCP interaction while acquiring an IP address for an interface, the client accepts the DHCP hostname option and sets the flag to show that the system now has a hostname configured.

DHCP-based Autoconfiguration and Image Update

You can use the DHCP image upgrade features to configure a DHCP server to download both a new image and a new configuration file to one or more switches in a network. Simultaneous image and configuration upgrade for all switches in the network helps ensure that each new switch added to a network are synchronous with the network.

There are two types of DHCP image upgrades: DHCP autoconfiguration and DHCP auto-image update.

Restrictions for DHCP-based Autoconfiguration

- The DHCP-based autoconfiguration with a saved configuration process stops if there is not at least one Layer 3 interface in an up state without an assigned IP address in the network.
- Unless you configure a timeout, the DHCP-based autoconfiguration with a saved configuration feature tries indefinitely to download an IP address.
- The auto-install process stops if a configuration file cannot be downloaded or if the configuration file is corrupted.

- The configuration file that is downloaded from TFTP is merged with the existing configuration in the running configuration but is not saved in the NVRAM unless you enter the **write memory** or **copy running-configuration startup-configuration** privileged EXEC command. If the downloaded configuration is saved to the startup configuration, the feature is not triggered during subsequent system restarts.

DHCP Autoconfiguration

DHCP autoconfiguration downloads a configuration file to one or more switches in your network from a DHCP server. The downloaded configuration file becomes the running configuration of the switch. It does not over write the bootup configuration saved in the flash, until you reload the switch.

DHCP Auto-Image Update

You can use DHCP auto-image upgrade with DHCP autoconfiguration to download both a configuration and a new image to one or more switches in your network. The switch (or switches) downloading the new configuration and the new image can be blank (or only have a default factory configuration loaded).

To enable a DHCP auto-image update on the switch, the TFTP server where the image and configuration files are located must be configured with the correct option 67 (the configuration filename), option 66 (the DHCP server hostname) option 150 (the TFTP server address), and option 125 (description of the Cisco IOS image file) settings.

After you install the switch in your network, the auto-image update feature starts. The downloaded configuration file is saved in the running configuration of the switch, and the new image is downloaded and installed on the switch. When you reboot the switch, the configuration is stored in the saved configuration on the switch.

DHCP Server Configuration Guidelines

Follow these guidelines if you are configuring a device as a DHCP server:

- You should configure the DHCP server with reserved leases that are bound to each switch by the switch hardware address.
- If you want the switch to receive IP address information, you must configure the DHCP server with these lease options:
    - IP address of the client (required)
    - Subnet mask of the client (required)
    - DNS server IP address (optional)
    - Router IP address (default gateway address to be used by the switch) (required)
- If you want the switch to receive the configuration file from a TFTP server, you must configure the DHCP server with these lease options:
    - TFTP server name (required)
    - Boot filename (the name of the configuration file that the client needs) (recommended)
    - Hostname (optional)

- Depending on the settings of the DHCP server, the switch can receive IP address information, the configuration file, or both.
- If you do not configure the DHCP server with the lease options described previously, it replies to client requests with only those parameters that are configured. If the IP address and the subnet mask are not in the reply, the switch is not configured. If the router IP address or the TFTP server name are not found, the switch might send broadcast, instead of unicast, TFTP requests. Unavailability of other lease options does not affect autoconfiguration.
- The switch can act as a DHCP server. By default, the Cisco IOS DHCP server and relay agent features are enabled on your switch but are not configured.

Purpose of the TFTP Server

Based on the DHCP server configuration, the switch attempts to download one or more configuration files from the TFTP server. If you configured the DHCP server to respond to the switch with all the options required for IP connectivity to the TFTP server, and if you configured the DHCP server with a TFTP server name, address, and configuration filename, the switch attempts to download the specified configuration file from the specified TFTP server.

If you did not specify the configuration filename, the TFTP server, or if the configuration file could not be downloaded, the switch attempts to download a configuration file by using various combinations of filenames and TFTP server addresses. The files include the specified configuration filename (if any) and these files: network-config, cisconet.cfg, *hostname*.config, or *hostname*.cfg, where *hostname* is the switch's current hostname. The TFTP server addresses used include the specified TFTP server address (if any) and the broadcast address (255.255.255.255).

For the switch to successfully download a configuration file, the TFTP server must contain one or more configuration files in its base directory. The files can include these files:

- The configuration file named in the DHCP reply (the actual switch configuration file).
- The network-confg or the cisconet.cfg file (known as the default configuration files).
- The router-confg or the ciscortr.cfg file (These files contain commands common to all switches. Normally, if the DHCP and TFTP servers are properly configured, these files are not accessed.)

If you specify the TFTP server name in the DHCP server-lease database, you must also configure the TFTP server name-to-IP-address mapping in the DNS-server database.

If the TFTP server to be used is on a different LAN from the switch, or if it is to be accessed by the switch through the broadcast address (which occurs if the DHCP server response does not contain all the required information described previously), a relay must be configured to forward the TFTP packets to the TFTP server. The preferred solution is to configure the DHCP server with all the required information.

Purpose of the DNS Server

The DHCP server uses the DNS server to resolve the TFTP server name to an IP address. You must configure the TFTP server name-to-IP address map on the DNS server. The TFTP server contains the configuration files for the switch.

You can configure the IP addresses of the DNS servers in the lease database of the DHCP server from where the DHCP replies will retrieve them. You can enter up to two DNS server IP addresses in the lease database.

The DNS server can be on the same LAN or on a different LAN from the switch. If it is on a different LAN, the switch must be able to access it through a router.

How to Obtain Configuration Files

Depending on the availability of the IP address and the configuration filename in the DHCP reserved lease, the switch obtains its configuration information in these ways:

- The IP address and the configuration filename is reserved for the switch and provided in the DHCP reply (one-file read method).

  The switch receives its IP address, subnet mask, TFTP server address, and the configuration filename from the DHCP server. The switch sends a unicast message to the TFTP server to retrieve the named configuration file from the base directory of the server and upon receipt, it completes its boot up process.

- The IP address and the configuration filename is reserved for the switch, but the TFTP server address is not provided in the DHCP reply (one-file read method).

  The switch receives its IP address, subnet mask, and the configuration filename from the DHCP server. The switch sends a broadcast message to a TFTP server to retrieve the named configuration file from the base directory of the server, and upon receipt, it completes its boot-up process.

- Only the IP address is reserved for the switch and provided in the DHCP reply. The configuration filename is not provided (two-file read method).

  The switch receives its IP address, subnet mask, and the TFTP server address from the DHCP server. The switch sends a unicast message to the TFTP server to retrieve the network-confg or cisconet.cfg default configuration file. (If the network-confg file cannot be read, the switch reads the cisconet.cfg file.)

  The default configuration file contains the hostnames-to-IP-address mapping for the switch. The switch fills its host table with the information in the file and obtains its hostname. If the hostname is not found in the file, the switch uses the hostname in the DHCP reply. If the hostname is not specified in the DHCP reply, the switch uses the default *Switch* as its hostname.

  After obtaining its hostname from the default configuration file or the DHCP reply, the switch reads the configuration file that has the same name as its hostname (*hostname*-confg or *hostname*.cfg, depending on whether network-confg or cisconet.cfg was read earlier) from the TFTP server. If the cisconet.cfg file is read, the filename of the host is truncated to eight characters.

  If the switch cannot read the network-confg, cisconet.cfg, or the hostname file, it reads the router-confg file. If the switch cannot read the router-confg file, it reads the ciscortr.cfg file.

Note The switch broadcasts TFTP server requests if the TFTP server is not obtained from the DHCP replies, if all attempts to read the configuration file through unicast transmissions fail, or if the TFTP server name cannot be resolved to an IP address.

How to Control Environment Variables

With a normally operating switch, you enter the boot loader mode only through the console connection configured for 9600 bps. Unplug the switch power cord, and press the **Mode** button while reconnecting the power cord. You can release the **Mode** button after all the amber system LEDs turn on and remain solid. The boot loader switch prompt then appears.

The switch boot loader software provides support for nonvolatile environment variables, which can be used to control how the boot loader, or any other software running on the system, operates. Boot loader environment variables are similar to environment variables that can be set on UNIX or DOS systems.

Environment variables that have values are stored in flash memory outside of the flash file system.

Each line in these files contains an environment variable name and an equal sign followed by the value of the variable. A variable has no value if it is not present; it has a value if it is listed even if the value is a null string. A variable that is set to a null string (for example, “ ”) is a variable with a value. Many environment variables are predefined and have default values.

You can change the settings of the environment variables by accessing the boot loader or by using Cisco IOS commands. Under normal circumstances, it is not necessary to alter the setting of the environment variables.

Common Environment Variables

This table describes the function of the most common environment variables.

Table 7: Common Environment Variables

Variable	Boot Loader Command	Cisco IOS Global Configuration Command
BOOT	**set BOOT** *filesystem* **:/** *file-url* ... A semicolon-separated list of executable files to try to load and execute when automatically booting.	**boot system** {*filesystem* **:** */file-url* ... | **switch** {*number* | **all**}} Specifies the Cisco IOS image to load during the next boot cycle and the stack members on which the image is loaded. This command changes the setting of the BOOT environment variable. The package provisioning file, also referred to as the *packages.conf* file, is used by the system to determine which software packages to activate during boot up. • When booting in installed mode, the package provisioning file specified in the **boot** command is used to determine which packages to activate. For example **boot flash:packages.conf**. • When booting in bundle mode, the package provisioning file contained in the booted bundle is used to activate the packages included in the bundle. For example, **boot flash:image.bin**.

Variable	Boot Loader Command	Cisco IOS Global Configuration Command
MANUAL_BOOT	**set MANUAL_BOOT yes** Decides whether the switch automatically or manually boots. Valid values are 1, yes, 0, and no. If it is set to no or 0, the boot loader attempts to automatically boot up the system. If it is set to anything else, you must manually boot up the switch from the boot loader mode.	**boot manual** Enables manually booting the switch during the next boot cycle and changes the setting of the MANUAL_BOOT environment variable. The next time you reboot the system, the switch is in boot loader mode. To boot up the system, use the **boot flash:** *filesystem* **:/** *file-url* boot loader command, and specify the name of the bootable image.
CONFIG_FILE	**set CONFIG_FILE flash:/** *file-url* Changes the filename that Cisco IOS uses to read and write a nonvolatile copy of the system configuration.	**boot config-file flash:/** *file-url* Specifies the filename that Cisco IOS uses to read and write a nonvolatile copy of the system configuration. This command changes the CONFIG_FILE environment variable.
SWITCH_NUMBER	**set SWITCH_NUMBER** *stack-member-number* Changes the member number of a stack member.	**switch** *current-stack-member-number* **renumber** *new-stack-member-number* Changes the member number of a stack member.
SWITCH_PRIORITY	**set SWITCH_PRIORITY** *stack-member-number* Changes the priority value of a stack member.	**switch** *stack-member-number* **priority** *priority-number* Changes the priority value of a stack member.
BAUD	**set BAUD** *baud-rate*	**line console 0** **speed** *speed-value* Configures the baud rate.
ENABLE_BREAK	**set ENABLE_BREAK yes/no**	**boot enable-break switch yes/no** Enables a break to the auto-boot cycle. You have 5 seconds to enter the **break** command.

Environment Variables for TFTP

When the switch is connected to a PC through the Ethernet management port, you can download or upload a configuration file to the boot loader by using TFTP. Make sure the environment variables in this table are configured.

Table 8: Environment Variables for TFTP

Variable	Description
MAC_ADDR	Specifies the MAC address of the switch. **Note** We recommend that you do not modify this variable. However, if you modify this variable after the boot loader is up or the value is different from the saved value, enter this command before using TFTP.
IP_ADDR	Specifies the IP address and the subnet mask for the associated IP subnet of the switch.
DEFAULT_ROUTER	Specifies the IP address and subnet mask of the default gateway.

Scheduled Reload of the Software Image

You can schedule a reload of the software image to occur on the switch at a later time (for example, late at night or during the weekend when the switch is used less), or you can synchronize a reload network-wide (for example, to perform a software upgrade on all switches in the network).

Note A scheduled reload must take place within approximately 24 days.

You have these reload options:

- Reload of the software to take affect in the specified minutes or hours and minutes. The reload must take place within approximately 24 hours. You can specify the reason for the reload in a string up to 255 characters in length.
- Reload of the software to take place at the specified time (using a 24-hour clock). If you specify the month and day, the reload is scheduled to take place at the specified time and date. If you do not specify the month and day, the reload takes place at the specified time on the current day (if the specified time is later than the current time) or on the next day (if the specified time is earlier than the current time). Specifying 00:00 schedules the reload for midnight.

The **reload** command halts the system. If the system is not set to manually boot up, it reboots itself.

If your switch is configured for manual booting, do not reload it from a virtual terminal. This restriction prevents the switch from entering the boot loader mode and then taking it from the remote user's control.

If you modify your configuration file, the switch prompts you to save the configuration before reloading. During the save operation, the system requests whether you want to proceed with the save if the CONFIG_FILE environment variable points to a startup configuration file that no longer exists. If you proceed in this situation, the system enters setup mode upon reload.

To cancel a previously scheduled reload, use the **reload cancel** privileged EXEC command.

How to Perform Switch Setup Configuration

Using DHCP to download a new image and a new configuration to a switch requires that you configure at least two switches. One switch acts as a DHCP and TFTP server and the second switch (client) is configured to download either a new configuration file or a new configuration file and a new image file.

Configuring DHCP Autoconfiguration (Only Configuration File)

This task describes how to configure DHCP autoconfiguration of the TFTP and DHCP settings on an existing switch in the network so that it can support the autoconfiguration of a new switch.

SUMMARY STEPS

1. **configure terminal**
2. **ip dhcp pool** *poolname*
3. **boot** *filename*
4. **network** *network-number mask prefix-length*
5. **default-router** *address*
6. **option 150** *address*
7. **exit**
8. **tftp-server flash**:*filename.text*
9. **interface** *interface-id*
10. **no switchport**
11. **ip address** *address mask*
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**ip dhcp pool** *poolname* **Example:** `Switch(config)# ip dhcp pool pool`	Creates a name for the DHCP server address pool, and enters DHCP pool configuration mode.
Step 3	**boot** *filename* **Example:** `Switch(dhcp-config)# boot config-boot.text`	Specifies the name of the configuration file that is used as a boot image.
Step 4	**network** *network-number mask prefix-length* **Example:** `Switch(dhcp-config)# network 10.10.10.0 255.255.255.0`	Specifies the subnet network number and mask of the DHCP address pool. **Note** The prefix length specifies the number of bits that comprise the address prefix. The prefix is an alternative way of specifying the network mask of the client. The prefix length must be preceded by a forward slash (/).
Step 5	**default-router** *address* **Example:** `Switch(dhcp-config)# default-router 10.10.10.1`	Specifies the IP address of the default router for a DHCP client.
Step 6	**option 150** *address* **Example:** `Switch(dhcp-config)# option 150 10.10.10.1`	Specifies the IP address of the TFTP server.
Step 7	**exit** **Example:** `Switch(dhcp-config)# exit`	Returns to global configuration mode.
Step 8	**tftp-server flash**:*filename.text* **Example:** `Switch(config)# tftp-server flash:config-boot.text`	Specifies the configuration file on the TFTP server.

	Command or Action	Purpose
Step 9	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/4`	Specifies the address of the client that will receive the configuration file.
Step 10	**no switchport** **Example:** `Switch(config-if)# no switchport`	Puts the interface into Layer 3 mode.
Step 11	**ip address** *address mask* **Example:** `Switch(config-if)# ip address 10.10.10.1 255.255.255.0`	Specifies the IP address and mask for the interface.
Step 12	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Related Topics

Example: Configuring a Switch as a DHCP Server, on page 74

Configuring DHCP Auto-Image Update (Configuration File and Image)

This task describes DHCP autoconfiguration to configure TFTP and DHCP settings on an existing switch to support the installation of a new switch.

Before You Begin

You must first create a text file (for example, autoinstall_dhcp) that will be uploaded to the switch. In the text file, put the name of the image that you want to download.

SUMMARY STEPS

1. **configure terminal**
2. **ip dhcp pool** *poolname*
3. **boot** *filename*
4. **network** *network-number mask prefix-length*
5. **default-router** *address*
6. **option 150** *address*
7. **option 125** *hex*
8. **copy tftp flash** *filename.txt*
9. **copy tftp flash** *imagename.bin*
10. **exit**
11. **tftp-server flash:** *config.text*
12. **tftp-server flash:** *imagename.bin*
13. **tftp-server flash:** *filename.txt*
14. **interface** *interface-id*
15. **no switchport**
16. **ip address** *address mask*
17. **end**
18. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip dhcp pool** *poolname* **Example:** `Switch(config)# ip dhcp pool pool1`	Creates a name for the DHCP server address pool and enter DHCP pool configuration mode.
Step 3	**boot** *filename* **Example:** `Switch(dhcp-config)# boot config-boot.text`	Specifies the name of the file that is used as a boot image.

	Command or Action	Purpose
Step 4	**network** *network-number mask prefix-length* **Example:** `Switch(dhcp-config)# network 10.10.10.0 255.255.255.0`	Specifies the subnet network number and mask of the DHCP address pool. **Note** The prefix length specifies the number of bits that comprise the address prefix. The prefix is an alternative way of specifying the network mask of the client. The prefix length must be preceded by a forward slash (/).
Step 5	**default-router** *address* **Example:** `Switch(dhcp-config)# default-router 10.10.10.1`	Specifies the IP address of the default router for a DHCP client.
Step 6	**option 150** *address* **Example:** `Switch(dhcp-config)# option 150 10.10.10.1`	Specifies the IP address of the TFTP server.
Step 7	**option 125** *hex* **Example:** `Switch(dhcp-config)# option 125 hex 0000.0009.0a05.08661.7574.6f69.6e73.7461.6c6c.5f64.686370`	Specifies the path to the text file that describes the path to the image file.
Step 8	**copy tftp flash** *filename.txt* **Example:** `Switch(config)# copy tftp flash image.bin`	Uploads the text file to the switch.
Step 9	**copy tftp flash** *imagename.bin* **Example:** `Switch(config)# copy tftp flash image.bin`	Uploads the tar file for the new image to the switch.
Step 10	**exit** **Example:** `Switch(dhcp-config)# exit`	Returns to global configuration mode.

	Command or Action	Purpose
Step 11	**tftp-server flash:** *config.text* **Example:** `Switch(config)# tftp-server flash:config-boot.text`	Specifies the Cisco IOS configuration file on the TFTP server.
Step 12	**tftp-server flash:** *imagename.bin* **Example:** `Switch(config)# tftp-server flash:image.bin`	Specifies the image name on the TFTP server.
Step 13	**tftp-server flash:** *filename.txt* **Example:** `Switch(config)# tftp-server flash:boot-config.text`	Specifies the text file that contains the name of the image file to download
Step 14	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitEthernet1/0/4`	Specifies the address of the client that will receive the configuration file.
Step 15	**no switchport** **Example:** `Switch(config-if)# no switchport`	Puts the interface into Layer 3 mode.
Step 16	**ip address** *address mask* **Example:** `Switch(config-if)# ip address 10.10.10.1 255.255.255.0`	Specifies the IP address and mask for the interface.
Step 17	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 18	**copy running-config startup-config** **Example:** `Switch(config-if)# end`	(Optional) Saves your entries in the configuration file.

Related Topics

Example: Configuring DHCP Auto-Image Update, on page 75

Configuring the Client to Download Files from DHCP Server

Note You should only configure and enable the Layer 3 interface. Do not assign an IP address or DHCP-based autoconfiguration with a saved configuration.

SUMMARY STEPS

1. **configure terminal**
2. **boot host dhcp**
3. **boot host retry timeout** *timeout-value*
4. **banner config-save ^C** *warning-message* **^C**
5. **end**
6. **show boot**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**boot host dhcp** **Example:** `Switch(conf)# boot host dhcp`	Enables autoconfiguration with a saved configuration.
Step 3	**boot host retry timeout** *timeout-value* **Example:** `Switch(conf)# boot host retry timeout 300`	(Optional) Sets the amount of time the system tries to download a configuration file. **Note** If you do not set a timeout, the system will try indefinitely to obtain an IP address from the DHCP server.
Step 4	**banner config-save ^C** *warning-message* **^C** **Example:** `Switch(conf)# banner config-save ^C Caution - Saving Configuration File to NVRAM May Cause You to No longer Automatically`	(Optional) Creates warning messages to be displayed when you try to save the configuration file to NVRAM.

	Command or Action	Purpose
	`Download Configuration Files at Reboot^C`	
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**show boot** **Example:** `Switch# show boot`	Verifies the configuration.

Related Topics

Example: Configuring a Switch to Download Configurations from a DHCP Server, on page 75

Manually Assigning IP Information to Multiple SVIs

This task describes how to manually assign IP information to multiple switched virtual interfaces (SVIs):

Note If the switch is running the IP services feature set, you can also manually assign IP information to a port if you first put the port into Layer 3 mode by using the **no switchport** interface configuration command.

SUMMARY STEPS

1. **configure terminal**
2. **interface vlan** *vlan-id*
3. **ip address** *ip-address subnet-mask*
4. **exit**
5. **ip default-gateway** *ip-address*
6. **end**
7. **show interfaces vlan** *vlan-id*
8. **show ip redirects**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface vlan** *vlan-id* **Example:** `Switch(config)# interface vlan 99`	Enters interface configuration mode, and enters the VLAN to which the IP information is assigned. The range is 1 to 4094.
Step 3	**ip address** *ip-address subnet-mask* **Example:** `Switch(config-vlan)# ip address 10.10.10.2 255.255.255.0`	Enters the IP address and subnet mask.
Step 4	**exit** **Example:** `Switch(config-vlan)# exit`	Returns to global configuration mode.
Step 5	**ip default-gateway** *ip-address* **Example:** `Switch(config)# ip default-gateway 10.10.10.1`	Enters the IP address of the next-hop router interface that is directly connected to the switch where a default gateway is being configured. The default gateway receives IP packets with unresolved destination IP addresses from the switch. Once the default gateway is configured, the switch has connectivity to the remote networks with which a host needs to communicate. **Note** When your switch is configured to route with IP, it does not need to have a default gateway set. **Note** The switch capwap relays on default-gateway configuration to support routed access point join the switch.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 7	**show interfaces vlan** *vlan-id* **Example:** `Switch# show interfaces vlan 99`	Verifies the configured IP address.

	Command or Action	Purpose
Step 8	**show ip redirects** **Example:** `Switch# show ip redirects`	Verifies the configured default gateway.

Modifying the Switch Startup Configuration

Specifying the Filename to Read and Write the System Configuration

By default, the Cisco IOS software uses the config.text file to read and write a nonvolatile copy of the system configuration. However, you can specify a different filename, which will be loaded during the next boot cycle.

Before You Begin

Use a standalone switch for this task.

SUMMARY STEPS

1. **configure terminal**
2. **boot flash**:/*file-url*
3. **end**
4. **show boot**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**boot flash**:/*file-url* **Example:** `Switch(config)# boot flash:config.text`	Specifies the configuration file to load during the next boot cycle. *file-url*—The path (directory) and the configuration filename. Filenames and directory names are case-sensitive.

	Command or Action	Purpose
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show boot** **Example:** `Switch# show boot`	Verifies your entries. The **boot** global configuration command changes the setting of the CONFIG_FILE environment variable.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Manually Booting the Switch

By default, the switch automatically boots up; however, you can configure it to manually boot up.

Before You Begin

Use a standalone switch for this task.

SUMMARY STEPS

1. **configure terminal**
2. **boot manual**
3. **end**
4. **show boot**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**boot manual** **Example:** `Switch(config)# boot manual`	Enables the switch to manually boot up during the next boot cycle.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show boot** **Example:** `Switch# show boot`	Verifies your entries. The **boot manual** global command changes the setting of the MANUAL_BOOT environment variable. The next time you reboot the system, the switch is in boot loader mode, shown by the *switch:* prompt. To boot up the system, use the **boot** boot loader command in installed boot mode or bundle boot mode. • `switch: boot flash:packages.conf` • `switch: boot flash:cat3850-universalk9.SSA.03.08.83.EMD.150-8.83.EMD.bin` Filenames and directory names are case-sensitive.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Booting the Switch in Installed Mode

SUMMARY STEPS

1. **cp** *source_file_path destination_file_path*
2. **software expand file** *source_file_path*
3. **reload**
4. **boot flash:packages.conf**
5. **show version**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**cp** *source_file_path destination_file_path* **Example:** `Switch#` **`copy`** **`tftp://10.0.0.6/cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin`** **`flash:`**	(Optional) Copies the bin file (`image.bin`) from the FTP or TFTP server to flash or USB flash.

Step 2	**software expand file** *source_file_path*
	
	Example:
	Expanding the bin file from the TFTP server:
	`Switch#` **`software expand file`**
	`tftp://10.0.0.2/cat3k_caa-universalk9.SSA.03.09.37.EXP.150-9.37.EXP.bin to flash:`
	`Preparing expand operation ...`
	`[1]: Downloading file`
	`tftp://10.0.0.2/cat3k_caa-universalk9.SSA.03.09.37.EXP.150-9.37.EXP.bin to active switch 1`
	`[1]: Finished downloading file`
	`tftp://10.0.0.2/cat3k_caa-universalk9.SSA.03.09.37.EXP.150-9.37.`
	`EXP.bin to active switch 1`
	`[1]: Copying software from active switch 1 to switch 2`
	`[1]: Finished copying software to switch 2`
	`[1 2]: Expanding bundle cat3k_caa-universalk9.SSA.03.09.37.EXP.150-9.37.EXP.bin`
	`[1 2]: Copying package files`
	`[1 2]: Package files copied`
	`[1 2]: Finished expanding bundle cat3k_caa-universalk9.SSA.03.09.37.EXP.150-9.37.EXP.bin`
	
	`18 -rw- 74387812 Dec 7 2012 05:55:43 +00:00 cat3k_caa-base.SSA.03.09.37.EXP.pkg`
	
	`19 -rw- 2738868 Dec 7 2012 05:55:44 +00:00`
	`cat3k_caa-drivers.SSA.03.09.37.EXP.pkg`
	`20 -rw- 32465772 Dec 7 2012 05:55:44 +00:00`
	`cat3k_caa-infra.SSA.03.09.37.EXP.pkg`
	`21 -rw- 30389036 Dec 7 2012 05:55:44 +00:00`
	`cat3k_caa-iosd-universalk9.SSA.150-9.37.EXP.pkg`
	`22 -rw- 18342624 Dec 7 2012 05:55:44 +00:00`
	`cat3k_caa-platform.SSA.03.09.37.EXP.pkg`
	`23 -rw- 63374028 Dec 7 2012 05:55:44 +00:00 cat3k_caa-wcm.SSA.10.0.10.14.pkg`
	
	`17 -rw- 1239 Dec 7 2012 05:56:29 +00:00 packages.conf`
		Expands the bin file stored in flash, FTP, TFTP, HTTP, or HTTPS server on the booted switch.
		Note Ensure that the `packages.conf` file is available in the expanded list.

Step 3	**reload** **Example:** `Switch:` **`reload`**	Reloads the switch.

	Command or Action	Purpose
		Note You can boot the switch manually or automatically using the `packages.conf` file. If you are booting manually, you can proceed to Step 4. Otherwise, the switch boots up automatically.
Step 4	**boot flash:packages.conf** **Example:** `switch:` **`boot flash:packages.conf`**	Boots the switch with the `packages.conf` file.

Step 5	**show version**
	
	Example:
	`switch#` **`show version`**
	
	`Switch Ports Model SW Version SW Image Mode`
	`------ ----- ----- ---------- ---------- ----`
	`1 6 WS-C3850-6DS-S 03.09.26.EXP ct3850-ipservicesk9 INSTALL`
		Verifies that the switch is in the **INSTALL** mode.

Booting the Switch in Bundle Mode

There are several methods by which you can boot the switch—either by copying the bin file from the TFTP server and then boot the switch, or by booting the switch straight from flash or USB flash using the commands **boot flash:<image.bin>** or **boot usbflash0:<image.bin>** .

The following procedure explains how to boot the switch from th TFTP server in the bundle mode.

SUMMARY STEPS

1. **cp** *source_file_path destination_file_path*
2. **switch:BOOT=<source path of .bin file>**
3. **boot**
4. **show version**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**cp** *source_file_path destination_file_path* **Example:** `Switch# copy tftp://10.0.0.6/cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin flash:`	(Optional) Copies the bin file (`image.bin`) from the FTP or TFTP server to flash or USB flash.
Step 2	**switch:BOOT=<source path of .bin file>** **Example:** `Switch: switch:BOOT=tftp://10.0.0.2/cat3k_caa-universalk9.SSA.03.09.37.EXP.150-9.37.EXP.bin`	Sets the boot parameters.
Step 3	**boot** **Example:** `switch: boot`	Boots the switch.

Step 4	**show version**
	
	Example:
	`switch# show version`
	`Switch Ports Model             SW Version       SW Image              Mode`
	`------ ----- -----             ----------       ----------            ----`
	`     1 6     WS-C3850-6DS-S    03.09.40.EXP     ct3850-ipservicesk9   BUNDLE`
		Verifies that the switch is in the **BUNDLE** mode.

Booting a Specific Software Image On a Switch Stack

SUMMARY STEPS

1. **configure terminal**
2. **boot system switch** {*number* | **all**} **flash:***image_file*| **tftp:** *image_file* | **usbflash0:** *image_file*
3. **end**
4. **show boot system**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**boot system switch** {*number* | **all**} **flash:***image_file*| **tftp:** *image_file* | **usbflash0:** *image_file* **Example:** `Switch(config)# boot system switch 2 flash:cat3850-universalk9.SSA.03.08.83.EMD.150-8.83.EMD.bin`	(Optional) For switches in a stack, specifies the switch members on which the system image is loaded during the next boot cycle: • Use *number* to specify a stack member. (Specify only one stack member.) • Use **all** to specify all stack members.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show boot system** **Example:** `Switch# show boot system`	Verifies your entries. The **boot system** global command changes the setting of the BOOT environment variable. During the next boot cycle, the switch attempts to automatically boot up the system using information in the BOOT environment variable.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring a Scheduled Software Image Reload

This task describes how to configure your switch to reload the software image at a later time.

SUMMARY STEPS

1. **configure terminal**
2. **copy running-config startup-config**
3. **reload in** [*hh*:]*mm* [*text*]
4. **reload slot** [*stack-member-number*]
5. **reload at** *hh*: *mm* [*month day* | *day month*] [*text*]
6. **reload cancel**
7. **show reload**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**copy running-config startup-config** **Example:** `copy running-config startup-config`	Saves your switch configuration information to the startup configuration before you use the **reload** command.

Step 3	**reload in** [*hh*:]*mm* [*text*]
	
	Example:
	`Switch(config)# reload in 12`
	
	`System configuration has been modified. Save? [yes/no]: y`
		Schedules a reload of the software to take affect in the specified minutes or hours and minutes. The reload must take place within approximately 24 days. You can specify the reason for the reload in a string up to 255 characters in length.

Step 4	**reload slot** [*stack-member-number*]
	
	Example:
	`Switch(config)# reload slot 6`
	
	`Proceed with reload? [confirm] y`
		Schedules a reload of the software in a switch stack.

Step 5	**reload at** *hh*: *mm* [*month day* | *day month*] [*text*] **Example:** `Switch(config)# reload at 14:00`	Specifies the time in hours and minutes for the reload to occur. **Note** Use the **at** keyword only if the switch system clock has been set (through Network Time Protocol (NTP), the hardware calendar, or manually). The time is relative to the configured time zone on the switch. To schedule reloads across several switches to occur simultaneously, the time on each switch must be synchronized with NTP.

	Command or Action	Purpose
Step 6	**reload cancel** **Example:** `Switch(config)# reload cancel`	Cancels a previously scheduled reload.
Step 7	**show reload** **Example:** `show reload`	Displays information about a previously scheduled reload or identifies if a reload has been scheduled on the switch.

Monitoring Switch Setup Configuration

Example: Verifying the Switch Running Configuration

```
Switch# show running-config
Building configuration...

Current configuration: 1363 bytes
!
version 12.4
no service pad
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Stack1
!
enable secret 5 $1$ej9.$DMUvAUnZOAmvmgqBEzIxE0
!
.
<output truncated>
.
interface gigabitethernet6/0/2
mvr type source

<output truncated>

...!
interface VLAN1
 ip address 172.20.137.50 255.255.255.0
 no ip directed-broadcast
!
ip default-gateway 172.20.137.1 !
!
snmp-server community private RW
snmp-server community public RO
snmp-server community private@es0 RW
snmp-server community public@es0 RO
snmp-server chassis-id 0x12
!
end
```

Examples: Displaying Software Bootup in Install Mode

This example displays software bootup in install mode:

```
switch: boot flash:packages.conf

Getting rest of image
Reading full image into memory....done
Reading full base package into memory...: done = 74596432
Nova Bundle Image
--------------------------------------
Kernel Address : 0x6042f354
Kernel Size : 0x318412/3245074
Initramfs Address : 0x60747768
Initramfs Size : 0xdc08e8/14420200
Compression Format: .mzip

Bootable image at @ ram:0x6042f354
Bootable image segment 0 address range [0x81100000, 0x81b80000] is in range [0x80180000,
0x90000000].
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@boot_system:
 377
Loading Linux kernel with entry point 0x811060f0 ...
Bootloader: Done loading app on core_mask: 0xf

### Launching Linux Kernel (flags = 0x5)

All packages are Digitally Signed
Starting System Services
Nov 7 09:57:05 %IOSXE-1-PLATFORM: process stack-mgr: %STACKMGR-1-DISC_START: Switch 2 is
starting stack discovery
#########################################################################################
Nov 7 09:59:07 %IOSXE-1-PLATFORM: process stack-mgr: %STACKMGR-1-DISC_DONE: Switch 2 has
finished stack discovery
Nov 7 09:59:07 %IOSXE-1-PLATFORM: process stack-mgr: %STACKMGR-1-SWITCH_ADDED: Switch 2 has
 been added to the stack
Nov 7 09:59:14 %IOSXE-1-PLATFORM: process stack-mgr: %STACKMGR-1-ACTIVE_ELECTED: Switch 2
has been elected ACTIVE

Restricted Rights Legend

Use, duplication, or disclosure by the Government is
subject to restrictions as set forth in subparagraph
(c) of the Commercial Computer Software - Restricted
Rights clause at FAR sec. 52.227-19 and subparagraph
(c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.

cisco Systems, Inc.
170 West Tasman Drive
San Jose, California 95134-1706



Cisco IOS Software, IOS-XE Software, Catalyst L3 Switch Software (CAT3K_CAA-UNIVERSALK9-M),

Version 03.09.12.EMD EARLY DEPLOYMENT ENGINEERING NOVA_WEEKLY BUILD, synced to
DSGS_PI2_POSTPC_FLO_DSBU7_NG3K_1105
Copyright (c) 1986-2012 by Cisco Systems, Inc.
Compiled Sun 04-Nov-12 22:53 by gereddy
License level to iosd is ipservices
```

This example display software bootup in bundle mode:

```
switch: boot flash:cat3k_caa-universalk9.SSA.03.09.12.EMD.150-9.12.EMD.bin


Reading full image into
memory..................................................................done
Nova Bundle Image
```

```
--------------------------------------
Kernel Address : 0x6042ff38
Kernel Size : 0x318412/3245074
Initramfs Address : 0x6074834c
Initramfs Size : 0xdc08e8/14420200
Compression Format: .mzip

Bootable image at @ ram:0x6042ff38
Bootable image segment 0 address range [0x81100000, 0x81b80000] is in range [0x80180000,
0x90000000].
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
File "flash:cat3k_caa-universalk9.SSA.03.09.12.EMD.150-9.12.EMD.bin" uncompressed and
installed, entry point: 0x811060f0
Loading Linux kernel with entry point 0x811060f0 ...
Bootloader: Done loading app on core_mask: 0xf

### Launching Linux Kernel (flags = 0x5)

All packages are Digitally Signed
Starting System Services
Nov 7 09:45:49 %IOSXE-1-PLATFORM: process stack-mgr: %STACKMGR-1-DISC_START: Switch 2 is
starting stack discovery
###########################################################################################
Nov 7 09:47:50 %IOSXE-1-PLATFORM: process stack-mgr: %STACKMGR-1-DISC_DONE: Switch 2 has
finished stack discovery
Nov 7 09:47:50 %IOSXE-1-PLATFORM: process stack-mgr: %STACKMGR-1-SWITCH_ADDED: Switch 2 has
 been added to the stack
Nov 7 09:47:58 %IOSXE-1-PLATFORM: process stack-mgr: %STACKMGR-1-ACTIVE_ELECTED: Switch 2
has been elected ACTIVE

Restricted Rights Legend

Use, duplication, or disclosure by the Government is
subject to restrictions as set forth in subparagraph
(c) of the Commercial Computer Software - Restricted
Rights clause at FAR sec. 52.227-19 and subparagraph
(c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.

cisco Systems, Inc.
170 West Tasman Drive
San Jose, California 95134-1706



Cisco IOS Software, IOS-XE Software, Catalyst L3 Switch Software (CAT3K_CAA-UNIVERSALK9-M),
 Version 03.09.12.EMD
EARLY DEPLOYMENT ENGINEERING NOVA_WEEKLY BUILD, synced to DSGS_PI2_POSTPC_FLO_DSBU7_NG3K_1105
Copyright (c) 1986-2012 by Cisco Systems, Inc.
Compiled Sun 04-Nov-12 22:53 by gereddy
License level to iosd is ipservices
```

Related Topics

Software Boot Modes, on page 43

Installed Boot Mode, on page 43

Bundle Boot Mode, on page 43

Example: Emergency Installation

This sample output is an example when the **emergency-install** boot command is initiated:

```
switch: emergency-install
tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.09.12.EMD.150-9.12.EMD.bin
```

```
The bootflash will be erased during install operation, continue (y/n)?y
Starting emergency recovery
(tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.09.12.EMD.150-9.12.EMD.bin)...
Reading full image into memory......................done
Nova Bundle Image
--------------------------------------
Kernel Address : 0x6042e5cc
Kernel Size : 0x318261/3244641
Initramfs Address : 0x60746830
Initramfs Size : 0xdb0fb9/14356409
Compression Format: .mzip

Bootable image at @ ram:0x6042e5cc
Bootable image segment 0 address range [0x81100000, 0x81b80000] is in range [0x80180000,
0x90000000].
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
File "sda9:c3850-recovery.bin" uncompressed and installed, entry point: 0x811060f0
Loading Linux kernel with entry point 0x811060f0 ...
Bootloader: Done loading app on core_mask: 0xf

### Launching Linux Kernel (flags = 0x5)



Initiating Emergency Installation of bundle
tftp://172.19.211.47/cstohs/cat3k_caa-universalk9.SSA.03.09.12.EMD.150-9.12.EMD.bin


Downloading bundle
tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.09.12.EMD.150-9.12.EMD.bin...
Validating bundle
tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.09.12.EMD.150-9.12.EMD.bin...
Installing bundle
tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.09.12.EMD.150-9.12.EMD.bin...
Verifying bundle
tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.09.12.EMD.150-9.12.EMD.bin...
Package cat3k_caa-base.SSA.03.09.12.EMD.pkg is Digitally Signed
Package cat3k_caa-drivers.SSA.03.09.12.EMD.pkg is Digitally Signed
Package cat3k_caa-infra.SSA.03.09.12.EMD.pkg is Digitally Signed
Package cat3k_caa-iosd-universalk9.SSA.150-9.12.EMD.pkg is Digitally Signed
Package cat3k_caa-platform.SSA.03.09.12.EMD.pkg is Digitally Signed
Package cat3k_caa-wcm.SSA.03.09.12.EMD.pkg is Digitally Signed
Preparing flash...
Syncing device...
Emergency Install successful... Rebooting
Restarting system.


Booting...(use DDR clock 667 MHz)Initializing and Testing RAM +++@@@@####...++@@++@@++@@++@
```

Related Topics

Software Boot Modes, on page 43

Installed Boot Mode, on page 43

Bundle Boot Mode, on page 43

Configuration Examples for Performing Switch Setup

Example: Configuring a Switch as a DHCP Server

```
Switch# configure terminal
Switch(config)# ip dhcp pool pool1
```

```
Switch(dhcp-config)# network 10.10.10.0 255.255.255.0
Switch(dhcp-config)# boot config-boot.text
Switch(dhcp-config)# default-router 10.10.10.1
Switch(dhcp-config)# option 150 10.10.10.1
Switch(dhcp-config)# exit
Switch(config)# tftp-server flash:config-boot.text
Switch(config)# interface gigabitethernet1/0/4
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.10.10.1 255.255.255.0
Switch(config-if)# end
```

Related Topics

Configuring DHCP Autoconfiguration (Only Configuration File), on page 54

Example: Configuring DHCP Auto-Image Update

```
Switch# configure terminal
Switch(config)# ip dhcp pool pool1
Switch(dhcp-config)# network 10.10.10.0 255.255.255.0
Switch(dhcp-config)# boot config-boot.text
Switch(dhcp-config)# default-router 10.10.10.1
Switch(dhcp-config)# option 150 10.10.10.1
Switch(dhcp-config)# option 125 hex 0000.0009.0a05.08661.7574.6f69.6e73.7461.6c6c.5f64.686370

Switch(dhcp-config)# exit
Switch(config)# tftp-server flash:config-boot.text
Switch(config)# tftp-server flash:image_name
Switch(config)# tftp-server flash:boot-config.text
Switch(config)# tftp-server flash: autoinstall_dhcp
Switch(config)# interface gigabitethernet1/0/4
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.10.10.1 255.255.255.0
Switch(config-if)# end
```

Related Topics

Configuring DHCP Auto-Image Update (Configuration File and Image), on page 56

Example: Configuring a Switch to Download Configurations from a DHCP Server

This example uses a Layer 3 SVI interface on VLAN 99 to enable DHCP-based autoconfiguration with a saved configuration:

```
Switch# configure terminal
Switch(conf)# boot host dhcp
Switch(conf)# boot host retry timeout 300
Switch(conf)# banner config-save ^C Caution - Saving Configuration File to NVRAM May Cause
 You to No longer Automatically Download Configuration Files at Reboot^C
Switch(config)# vlan 99
Switch(config-vlan)# interface vlan 99
Switch(config-if)# no shutdown
Switch(config-if)# end
Switch# show boot
BOOT path-list:
Config file:          flash:/config.text
Private Config file:  flash:/private-config.text
Enable Break:         no
Manual Boot:          no
HELPER path-list:
```

```
NVRAM/Config file
      buffer size:    32768
Timeout for Config
          Download:    300 seconds
Config Download
       via DHCP:       enabled (next boot: enabled)
Switch#
```

Related Topics

Configuring the Client to Download Files from DHCP Server, on page 60

Examples: Scheduling Software Image Reload

This example shows how to reload the software on the switch on the current day at 7:30 p.m:

```
Switch# reload at 19:30
Reload scheduled for 19:30:00 UTC Wed Jun 5 2013 (in 2 hours and 25 minutes)
Proceed with reload? [confirm]
```

This example shows how to reload the software on the switch at a future time:

```
Switch# reload at 02:00 jun 20
Reload scheduled for 02:00:00 UTC Thu Jun 20 2013 (in 344 hours and 53 minutes)
Proceed with reload? [confirm]
```

Additional References For Performing Switch Setup

Related Documents

Related Topic	Document Title
Switch setup commands Boot loader commands	*System Management Command Reference (Catalyst 3850 Switches)*
Pre-download feature	*System Management Configuration Guide (Cisco WLC 5700 Series)*
IOS XE DHCP configuration	*IP Addressing Configuration Guide Library, Cisco IOS XE Release 3S (Catalyst 3850 Switches)*
Hardware installation	*Catalyst 3850 Switch Hardware Installation Guide*
Platform-independent command references	*Configuration Fundamentals Command Reference, Cisco IOS XE Release 3S (Catalyst 3850 Switches)*

Related Topic	Document Title
Platform-independent configuration information	*Configuration Fundamentals Configuration Guide, Cisco IOS XE Release 3S (Catalyst 3850 Switches)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information For Performing Switch Setup Configuration

Command History

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 4

Configuring Right-To-Use Licenses

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Restrictions for Configuring RTU Licenses

The following are the restrictions for configuring and using RTU licenses.

- AP count licenses can be ordered and pre-activated on your switch.
- Imaged based licenses can be upgraded. AP count licenses can be deactivated and moved between switches and controllers.

- To activate a permanent license, you must reboot your switch after configuring the new image level. The AP-count license does not require a reboot to activate.
- An expired image based evaluation license can not be reactivated after reboot.
- Stack members of a switch stack must run the same license level.
- Your switch is pre-installed with the image that you ordered. If an image was not pre-ordered, then the switch is booted with a LAN base image by default.
- Adder AP-count licenses are installed in the factory.

Related Topics

Activating an Imaged Based License, on page 83

Examples: Activating RTU Image Based Licenses, on page 89

Information About Configuring RTU Licenses

Right-To-Use Licensing

Right-to-use (RTU) licensing allows you to order and activate a specific license type and level, and then to manage license usage on your switch. The types of licenses available to order are:

- Permanent licenses—Purchased with a specific feature set with no expiration date.
- Evaluation licenses—Pre-installed on the switch and is valid for only a 90 day in-use period.

To activate a permanent or evaluation license, you are required to accept the End-User License Agreement (EULA). For the evaluation license, you are notified to purchase a permanent license or deactivate the license before the 90 day period expires.

A permanent license can be moved from one device to another. To activate a license, you must reboot your switch.

An evaluation license is a manufacturing image on your switch and is not transferable to another switch. This type of license cannot be reactivated after reboot.

Related Topics

Activating an Imaged Based License, on page 83

Examples: Activating RTU Image Based Licenses, on page 89

Right-To-Use Image Based Licenses

Right-to-use imaged licenses support a set of features based on a specific image-based license:

- LAN Base—Layer 2 features.
- IP Base—Layer 2 and Layer 3 features.
- IP Services—Layer 2, Layer 3, and IPv6 features. (Applicable only to switches and not controllers.)

The default image based license is LAN Base.

Right-To-Use License States

After you configure a specific license type and level, you can manage your licenses by monitoring the license state.

Table 9: RTU License States

License State	Description
Active, In Use	EULA was accepted and the license is in use after device reboot.
Active, Not In Use	EULA was accepted and the switch is ready to use when the license is enabled.
Not Activated	EULA was not accepted.

Guidelines to follow when monitoring your image based license state:

- A purchased permanent license is set to *Active, In Use* state only after a switch reboot.
- If more than one license was purchased, a reboot will activate the license with the highest feature set. For instance, the IP Services license is activated and not the LAN Base license.
- Remaining licenses purchased after switch reboot, stay in **Active, Not In Use** state.

Note For the AP count license, to change the state to Active, In Use, you must first make sure that the evaluation AP count license is deactivated.

License Activation for Switch Stacks

Right-to-use licensing is supported on switch stacks. You can connect only one switch type in a stack. One switch in the stack is identified as the active switch and the remaining switches are standby switches. The active switch is the switch that is activated with an RTU license and from its active console, the license level for the standby switches in the stack can be activated at the same time.

A new switch is allowed to join the switch stack if its license level matches. If there is a mismatch, then the active switch can reconfigure the license level and reboot it to allow it to join the stack.

Mobility Controller Mode

AP-count licenses are used only when the switch is in Mobility Controller mode. The MC is the gatekeeper for tracking the AP-count licenses and allows an access point to join or not.

Management of AP-count licenses is performed by the switch in mobility controller mode configurable through the CLI.

Related Topics

Changing Mobility Mode, on page 87

Right-To-Use AP-Count Licensing

Right-to-use licensing (RTU) allows you to order and activate a specific license type, and then to manage license usage on your switch.

You can order your switch with support for any number of access points as the adder access point count licenses but the total number of the licenses ordered should not exceed 50. You can also order the adder access point count licenses after receiving the switch.

For example, if you have ordered 50 new adder licenses, you can add only those ordered adder licenses to the switch. The licenses can be added in increments of 1, but the total number of licenses added for the switch should not exceed 50 .

You can configure your switch to manage the access point count licenses and view the number of access points currently in use from the CLI.

The following are the two different types of access point licenses:

1. Permanent licenses for the access points
   - Adder access point count license—You can purchase the adder license to increase the switch capacity at the later point of time. You can transfer the adder access point count license from one switch to another.
2. Evaluation licenses for the access points
   - You can activate the evaluation licenses to evaluate more access point count licenses before purchasing.
   - Maximum number of access points that can be evaluated is 50 .
   - The evaluation period for trying for the access point licenses is 90 days.
   - You can activate and deactivate the evaluation license for the access points from the CLI.

Related Topics

Activating an AP-Count License, on page 84
Obtaining an Upgrade or Capacity Adder License, on page 85
Rehosting a License, on page 86

Right-to-Use AP-Count Evaluation Licenses

If you are considering upgrading to a license with a higher access point count, you can try an evaluation license before upgrading to a permanent version of the license.

When an evaluation license is activated, the permanent AP-count licenses are ignored. .

To prevent disruptions in operation, the switch does not change licenses when an evaluation license expires. A warning expiry message is displayed daily starting five days prior to the expiry date. After 90 days, the evaluation license expires with a warning message. You must disable the evaluation license and then purchase the permanent license.

When the switch reboots after the evaluation license expiry, the license defaults to a permanent license.

Related Topics

Activating an AP-Count License, on page 84
Obtaining an Upgrade or Capacity Adder License, on page 85
Rehosting a License, on page 86

Right-To-Use Adder AP-Count Rehosting Licenses

Revoking a license from one device and installing it on another is called rehosting. You might want to rehost a license to change the purpose of a device.

To rehost a license, you must deactivate the adder ap-count license from one device and activate the same license on another device.

Evaluation licenses cannot be rehosted.

How to Configure RTU Licenses

Activating an Imaged Based License

SUMMARY STEPS

1. **license right-to-use activate**{**ipbase** |**ipservices** | **lanbase**} {**all** | **evaluation all** } [**slot** *slot-number*] [**acceptEULA**]
2. **reload** [*LINE* | **at** | **cancel** | **in** | **slot** *stack-member-number* | **standby-cpu**]
3. **show license right-to-use usage** [**slot** *slot-number*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**license right-to-use activate**{**ipbase** |**ipservices** | **lanbase**} {**all** | **evaluation all** } [**slot** *slot-number*] [**acceptEULA**] **Example:** `Switch# license right-to-use activate ipservices all acceptEULA`	Activates a type of image based license. Activation can happen on all switches and also include the EULA acceptance. **Note** If you do not accept EULA, the modified configuration will not take effect after reload. The default license (or a license that was not deactivated) becomes active after reload.

	Command or Action	Purpose

Step 2	**reload** [*LINE* | **at** | **cancel** | **in** | **slot** *stack-member-number* | **standby-cpu**]
	
	Example:
	`Switch# reload slot 1`
	`Proceed with reload? [confirm] y`
		Reloads a specific stack member to complete the activation process for the RTU adder AP-count license.
		
		Note The reminder to accept a EULA is displayed after reload if it was not accepted earlier.

Step 3	**show license right-to-use usage** [**slot** *slot-number*]
	
	Example:
	`Switch# show license right-to-use usage`
	
	`Slot#  License Name   Type   usage-duration(y:m:d)  In-Use  EULA`
	`-----------------------------------------------------------------------`
	`1  ipservices  permanent  0 :10 :0  yes  yes`
	`1  ipbase  permanent  0 :0 :0  no  no`
	`1  ipbase  evaluation  0 :0 :0  no  no`
	`1  lanbase  permanent  0 :0 :7  no  yes`
	`1  apcount  evaluation  0 :0 :0  no  no`
	`1  apcount  base  0 :0 :0  no  no`
	`1  apcount  adder  0 :0 :0  no  no`
	
	`Switch#`
		Displays detailed usage information.

Related Topics

Restrictions for Configuring RTU Licenses, on page 79

Right-To-Use Licensing, on page 80

Monitoring and Maintaining RTU Licenses, on page 88

Examples: Activating RTU Image Based Licenses, on page 89

Activating an AP-Count License

SUMMARY STEPS

1. **license right-to-use activate**{**apcount** *ap-number* **slot** *slot-num*} | **evaluation**} [**acceptEULA**]
2. **show license right-to-use usage** [**slot** *slot-number*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**license right-to-use activate**{**apcount** *ap-number* **slot** *slot-num*} | **evaluation**} [**acceptEULA**] **Example:** `Switch# license right to use activate apcount 5 slot 1 acceptEULA`	Activates one or more adder AP-count licenses and immediately accepts the EULA.
Step 2	**show license right-to-use usage** [**slot** *slot-number*] **Example:** (see output below)	Displays detailed usage information.

```
Switch# show license right-to-use usage

 Slot#  License Name     Type     usage-duration(y:m:d)  In-Use  EULA
-----------------------------------------------------------------------

1       ipservices      permanent  0 :3 :29               yes    yes
1       ipservices      evaluation 0 :0 :0                no     no
1       ipbase          permanent  0 :0 :0                no     no
1       ipbase          evaluation 0 :0 :0                no     no
1       lanbase         permanent  0 :0 :0                no     no
1       apcount         evaluation 0 :3 :11               no     no
1       apcount         base       0 :0 :0                no     yes
1       apcount         adder      0 :0 :17               yes    yes


Switch#
```

Related Topics

Monitoring and Maintaining RTU Licenses, on page 88

Right-To-Use AP-Count Licensing, on page 82

Right-to-Use AP-Count Evaluation Licenses, on page 82

Obtaining an Upgrade or Capacity Adder License

You can use the capacity adder licenses to increase the number of access points supported by the switch.

SUMMARY STEPS

1. **license right-to-use** {**activate** | **deactivate**} **apcount** {*ap-number* | **evaluation** } **slot** *slot-num* [**acceptEULA**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**license right-to-use** {**activate** | **deactivate**} **apcount** {*ap-number* | **evaluation** } **slot** *slot-num* [**acceptEULA**] **Example:** `Switch# license right to use activate apcount 5 slot 2 acceptEULA`	Activates one or more adder AP-count licenses and immediately accepts the EULA.

Related Topics

Right-to-Use AP-Count Evaluation Licenses, on page 82

Right-To-Use AP-Count Licensing, on page 82

Rehosting a License

To rehost a license, you have to deactivate the license from one switch and then activate the same license on another switch.

SUMMARY STEPS

1. **license right-to-use deactivate apcount** *ap-number* **slot** *slot-num* [**acceptEULA**]
2. **license right-to-use activate apcount** *ap-number* **slot** *slot-num* [**acceptEULA**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**license right-to-use deactivate apcount** *ap-number* **slot** *slot-num* [**acceptEULA**] **Example:** `Switch# license right to use deactivate apcount 1 slot 1 acceptEULA`	Deactivates the license on one switch.
Step 2	**license right-to-use activate apcount** *ap-number* **slot** *slot-num* [**acceptEULA**] **Example:** `Switch# license right to use activate apcount 2 slot 2 acceptEULA`	Activates the license on another switch.

Related Topics

Right-To-Use AP-Count Licensing, on page 82

Right-to-Use AP-Count Evaluation Licenses, on page 82

Changing Mobility Mode

SUMMARY STEPS

1. **wireless mobility controller**
2. **write memory**
3. **reload** [*LINE* | **at** | **cancel** | **in** | **slot** *stack-member-number* | **standby-cpu**]
4. **no wireless mobility controller**
5. **write memory**
6. **reload** [*LINE* | **at** | **cancel** | **in** | **slot** *stack-member-number* | **standby-cpu**]

DETAILED STEPS

	Command or Action	Purpose

Step 1	**wireless mobility controller**
	
	Example:
	`Switch(config)# wireless mobility controller`
	`%`
	`Mobility role changed to Mobility Controller.`
	`Please save config and reboot the whole stack.`
		Changes a switch in Mobility Agent mode to Mobility Controller mode.

Step 2	**write memory**
	
	Example:
	`Switch# write memory`
	
	`Building configuration...`
	`Compressed configuration from 13870 bytes to 5390 bytes[OK]`
	`Switch#`	

Step 3	**reload** [*LINE*	**at**	**cancel**	**in**	**slot** *stack-member-number*	**standby-cpu**]
						
						Example:
						`Switch# reload slot 3`
						`Proceed with reload? [confirm] y`	

Step 4	**no wireless mobility controller**
	
	Example:
	`Switch(config)# no wireless mobility controller`
	`%`
	`Mobility role changed to Mobility Agent.`
	`Please save config and reboot the whole stack.`
	`Switch(config)#`
		Changes a switch in Mobility Controller mode to Mobility Agent mode.

Step 5	**write memory**	

	Command or Action	Purpose

	Example:
	`Switch# write memory`
	
	`Building configuration...`
	`Compressed configuration from 13870 bytes to 5390 bytes[OK]`
	`Switch#`	

Step 6	**reload** [*LINE* | **at** | **cancel** | **in** | **slot** *stack-member-number* | **standby-cpu**]
	
	Example:
	`Switch# reload slot 3`
	`Proceed with reload? [confirm] y`	

Related Topics

Mobility Controller Mode, on page 81

Monitoring and Maintaining RTU Licenses

Command	Purpose
show license right-to-use default	Displays the default license information.
show license right-to-use detail	Displays detailed information of all the licenses in the switch stack.
show license right-to-use eula {adder | evaluation | permanent}	Displays the end user license agreement.
show license right-to-use mismatch	Displays the license information that does not match.
show license right-to-use slot *slot-number*	Displays the license information for a specific slot in a switch stack.
show license right-to-use summary	Displays a summary of the license information on the entire switch stack.
show license right-to-use usage [**slot** *slot-number*]	Displays detailed information about usage for all licenses in the switch stack.
show switch	Displays detailed information of every member in a switch stack including the state of the license.

Related Topics

Activating an Imaged Based License, on page 83
Examples: Activating RTU Image Based Licenses, on page 89
Activating an AP-Count License, on page 84

Configuration Examples for RTU Licensing

Examples: Activating RTU Image Based Licenses

This example shows how to activate an IP Services image license and accept the EULA for a specific slot:

```
Switch# license right-to-use activate ipservices slot 1 acceptEULA
% switch-1:stack-mgr:Reboot the switch to invoke the highest activated License level
```

This example shows how to activate a license for evaluation:

```
Switch# license right-to-use activate ipservices evaluation acceptEULA
% switch-1:stack-mgr:Reboot the switch to invoke the highest activated License level
```

Related Topics

Activating an Imaged Based License, on page 83
Restrictions for Configuring RTU Licenses, on page 79
Right-To-Use Licensing, on page 80
Monitoring and Maintaining RTU Licenses, on page 88

Examples: Displaying RTU Licensing Information

This example shows the consolidated RTU licensing information from the active switch on a switch stack. All of the members in the stack have the same license level. When the evaluation AP-count license is activated, the adder AP-count licenses are ignored. The maximum number of AP-count licenses are available when evaluation is enabled.

```
Switch# show license right-to-use summary

 License Name     Type           Count   Period left
-------------------------------------------------------
  ipservices      permanent        10      Lifetime
  apcount         evaluation       40        90

-------------------------------------------------------

License Level In Use: ipservices
License Level on Reboot: ipbase
Evaluation AP-Count: Enabled
Total AP Count Licenses: 50
AP Count Licenses In-use: 10
AP Count Licenses Remaining: 40
```

This example shows a summary of permanent and adder licenses. The evaluation AP-count license is disabled displaying the total number of activated adder AP-count licenses in the switch stack. AP-count licenses in-use mean that they are connected.

```
Switch# show license right-to-use summary

  License Name    Type          Count      Period left
-------------------------------------------------------------
  ipservices     permanent      N/A         Lifetime
  apcount        base           0
  apcount        adder          40          Lifetime

-------------------------------------------------------------

License Level In Use: ipservices
License Level on Reboot: ipservices eval
Evaluation AP-Count: Disabled
Total AP Count Licenses: 40
AP Count Licenses In-use: 10
AP Count Licenses Remaining: 30
```

This example shows the RTU default licenses. Default licenses are pre-installed and cannot be removed or transferred. If no license is activated the switch uses the default license, after a reboot.

```
Switch# show license right-to-use default

 Slot#    License Name    Type           Count
----------------------------------------------------
 1        ipservices      permanent     N/A
 1        apcount         base          0
 1        apcount         adder         10

 Slot#    License Name    Type           Count
----------------------------------------------------
 2        ipservices      permanent     N/A
 2        apcount         base          0
 2        apcount         adder         10

 Slot#    License Name    Type           Count
----------------------------------------------------
 3        ipservices      permanent     N/A
 3        apcount         base          0
 3        apcount         adder         10
```

Example: Displaying RTU License Details

This example shows all the detailed information for the RTU licenses on slot 1:

```
Switch# show license right-to-use detail slot 1

Index 1:  License Name: ipservices
           Period left: Lifetime
           License Type: permanent
           License State: Active, In use
           License Count: Non-Counted
           License Location: Slot 1
Index 2:  License Name: ipservices
           Period left: 90
           License Type: evaluation
           License State: Not Activated
           License Count: Non-Counted
           License Location: Slot 1
Index 3:  License Name: ipbase
           Period left: Lifetime
```

```
           License Type: permanent
           License State: Active, Not In use
           License Count: Non-Counted
           License Location: Slot 1
Index 4:  License Name: ipbase
           Period left: 90
           License Type: evaluation
           License State: Not Activated
           License Count: Non-Counted
           License Location: Slot 1
           License Location: Standby Switch 1
Index 5:  License Name: lanbase
           Period left: Lifetime
           License Type: permanent
           License State: Not Activated
           License Count: Non-Counted
           License Location: Slot 1
Index 6:  License Name: apcount
           Period left: 90
           License Type: evaluation
           License State: Active, In use
           License Count: 50
           License Location: Slot 1
Index 7:  License Name: apcount
           Period left: Lifetime
           License Type: base
           License State: Active, Not In use
           License Count: 0
           License Location: Slot 1
Index 8:  License Name: apcount
           Period left: Lifetime
           License Type: adder
           License State: Active, Not In use
           License Count: 10
           License Location: Slot 1
```

Example: Displaying RTU License Mismatch

This example shows the license information of the switches in a stack and a mismatch state of a member switch. The member must match the active.

```
Switch# show switch

Switch/Stack Mac Address : 6400.f125.0c80
                                               H/W   Current
Switch#   Role        Mac Address     Priority Version  State
------------------------------------------------------------------------------
 1       Standby     6400.f125.1b00     1      0        Ready
*2       Active      6400.f125.0c80     1      V01      Ready
 3       Member      6400.f125.1780     1      0        Lic-Mismatch
```

Note To resolve the license mismatch, first check the RTU license summary:

```
Switch# show switch right-to-use summary
```

Then change the license level of the mismatched switched so that it is the same license level of the active switch. This example shows that the IP Base license was activated for the member switch to match the active switch.

```
Switch# license right-to-use activate ipbase slot 1 acceptEULA
```

Example: Displaying RTU Licensing Usage

This example shows the detailed licensing usage on your switch stack. The IP Services license in Slot 1 is permanent and usage is one day. An AP-count license in Slot 2 is ready for evaluation. EULA was accepted and state shows in use, but after reboot the evaluation license will be deactivated.

```
Switch# show license right-to-use usage

 Slot#  License Name     Type     usage-duration(y:m:d)  In-Use  EULA
-------------------------------------------------------------------------------------------

 1      ipservices     permanent      0 :0 :1            yes     yes
 1      ipservices     evaluation     0 :0 :0            no      no
 1      ipbase         permanent      0 :0 :0            no      yes
 1      ipbase         evaluation     0 :0 :0            no      no
 1      lanbase        permanent      0 :0 :0            no      no
 1      apcount        evaluation     0 :0 :0            yes     yes
 1      apcount        base           0 :0 :0            no      yes
 1      apcount        adder          0 :0 :0            no      yes

 Slot#  License Name     Type     usage-duration(y:m:d)  In-Use  EULA
-------------------------------------------------------------------------------------------

 2      ipservices     permanent       0 :0 :1           yes     no
 2      ipservices     evaluation      0 :0 :0           no      yes
 2      ipbase         permanent       0 :0 :0           no      yes
 2      ipbase         evaluation      0 :0 :0           no      no
 2      lanbase        permanent       0 :0 :0           no      no
 2      apcount        evaluation      0 :0 :0           yes     yes
 2      apcount        base            0 :0 :0           no      yes
 2      apcount        adder           0 :0 :0           no      no

 Slot#  License Name     Type     usage-duration(y:m:d)  In-Use  EULA
-------------------------------------------------------------------------------------------

 3      ipservices     permanent       0 :0 :1           yes     yes
 3      ipservices     evaluation      0 :0 :0           no      no
 3      ipbase         permanent       0 :0 :0           no      no
 3      ipbase         evaluation      0 :0 :0           no      no
 3      lanbase        permanent       0 :0 :0           no      no
 3      apcount        evaluation      0 :0 :0           yes     yes
 3      apcount        base            0 :0 :0           no      yes
 3      apcount        adder           0 :0 :0           no      no
```

Additional References for RTU Licensing

Related Documents

Related Topic	Document Title
RTU commands	*System Management Command Reference (Catalyst 3850 Switches)*
RTU AP image preload feature	*System Management Configuration Guide (Cisco WLC 5700 Series)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for RTU Licensing

Release	Feature Information
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 5

Configuring Administrator Usernames and Passwords

Finding Feature Information

Your software release may not support all of the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About Configuring Administrator Usernames and Passwords

You can configure administrator usernames and passwords to prevent unauthorized users from reconfiguring the switch and viewing configuration information. This section provides instructions for initial configuration and for password recovery.

You can also set administrator usernames and passwords to manage and configure one or more access points that are associated with the switch.

Strong Passwords

You can set strong administrator passwords such as encrypted passwords with ASCII keys for the administrator user for managing access points.

Use the following guidelines while creating strong passwords:

- There should be at least three of the following categories—lowercase letters, uppercase letters, digits, and special characters.
- The new password should not be the same as that of the associated username and the username should not be reversed.
- The characters in the password should not be repeated more than three times consecutively.
- The password should not be **cisco**, **ocsic**, **admin**, **nimda**, or any variant obtained by changing the capitalization of letters therein, or by substituting "1" "|" or "!" for i, and/or substituting "0" for "o", and/or substituting "$" for "s".
- The maximum number of characters accepted for the username and password is 32.

Encrypted Passwords

You can set three types of keys for the password:

- Randomly generated key—This key is generated randomly and it is the most secure option. To export the configuration file from one system to another, the key should also be exported.
- Static key—The simplest option is to use a fixed (static) encryption key. By using a fixed key, no key management is required, but if the key is somehow discovered, the data can be decrypted by anyone with the knowledge of that key. This is not a secure option and it is called obfuscation in the CLI.
- User defined key—You can define the key by yourself. To export the configuration file from one system to another, both systems should have the same key configured.

Configuring Administrator Usernames and Passwords

SUMMARY STEPS

1. **configure terminal**
2. **wireless security strong-password**
3. **username** *admin-username* **password** **{0** *unencrypted_password* **| 7** *hidden_password* **|** *unencrypted_text***}**
4. **username** *admin-username* **secret** **{0** *unencrypted_secret_text* **| 4** *SHA256 encrypted_secret_text* **| 5** *MD5 encrypted_secret_text* **| LINE}**
5. **ap mgmtuser username** *username* **password {0** *unencrypted password* **| 8** *AES encrypted password* **}secret {0** *unencrypted password* **| 8** *AES encrypted password* **}**
6. **ap dot1x username** *username* **password {0** *unencrypted password* **| 8** *AES encrypted password* **}**
7. **end**
8. **ap name apname mgmtuser username** *username***password** *password* **secret** *secret _text*
9. **ap name apname dot1x-user** *username* **password** *password*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch#` **`configure terminal`**	Enters global configuration mode.
Step 2	**wireless security strong-password** **Example:** `Switch(config)#` **`wireless security strong-password`**	Enables strong password policy for the administrator user.
Step 3	**username** *admin-username* **password** **{0** *unencrypted_password* **| 7** *hidden_password* **|** *unencrypted_text***}** **Example:** `Switch(config)#` **`username adminuser1 password 0 QZsek239@`**	Specifies a username and password for an administrator. The administrator can configure the switch and view the configured information.
Step 4	**username** *admin-username* **secret** **{0** *unencrypted_secret_text* **| 4** *SHA256 encrypted_secret_text* **| 5** *MD5 encrypted_secret_text* **| LINE}** **Example:** `Switch(config)#` **`username adminuser1 secret 0 QZsek239@`**	Specifies the secret for the administrator.
Step 5	**ap mgmtuser username** *username* **password {0** *unencrypted password* **| 8** *AES encrypted password* **}secret {0** *unencrypted password* **| 8** *AES encrypted password* **}** **Example:** `Switch(config)#` **`ap mgmtuser username cisco password 0 Qwci12@ secret 0 Qwci14@!`**	Specifies administrator username and password for managing all of the access points configured to the switch. You can also include the secret text to perform privileged access point management. **Note** If your password is not strong enough to fulfill the strong password policy, then the password is rejected with a valid error message. For example, the following password is rejected because it is not a strong password. `Switch#` **`ap mgmtuser username cisco password 0 abcd secret 0 1234`**
Step 6	**ap dot1x username** *username* **password {0** *unencrypted password* **| 8** *AES encrypted password* **}** **Example:** `Switch(config)#` **`ap dot1x username cisco password 0 Qwci12@`**	Specifies the 802.1X username and password for managing all of the access points configured to the switch.
Step 7	**end** **Example:** `Switch(config)#` **`end`**	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

	Command or Action	Purpose
Step 8	**ap name apname mgmtuser username** *username***password** *password* **secret** *secret _text* **Example:** `Switch# ap name APf0f7.55c7.7b23 mgmtuser username cisco password Qne35! secret Nzep592$`	Configures the administrator username, password, and secret text for managing a specific access point that is configured to the switch.
Step 9	**ap name apname dot1x-user** *username* **password** *password* **Example:** `Switch# ap name APf0f7.55c7.7b23 dot1x-user username cisco password Qne35!`	Configures the 802.1X username and password for a specific access point.

Examples: Administrator Usernames and Passwords Configuration

This example shows how to configure administrator usernames and passwords with the strong password policy in configuration mode:

```
Switch# configure terminal
Switch(config)# wireless security strong-password
Switch(config)# username adminuser1 password 0 QZsek239@
Switch(config)# ap mgmtuser username cisco password 0 Qwci12@ secret 0 Qwci14@!
Switch(config)# ap dot1x username cisco password 0 Qwci12@
Switch# end
```

This example shows how to configure administrator usernames and passwords for an access point in global EXEC mode:

```
Switch# wireless security strong-password
Switch# ap name APf0f7.55c7.7b23 mgmtuser username cisco password Qwci12@ secret Qwci14@
Switch# ap name APf0f7.55c7.7b23 dot1x-user username cisco password Qwci12@
Switch# end
```

Additional References for Administrator Usernames and Passwords

Related Documents

Related Topic	Document Title
System management commands	*System Management Command Reference Guide (Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information For Performing Administrator Usernames and Passwords Configuration

Release	Feature Information
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 6

Configuring 802.11 parameters and Band Selection

Finding Feature Information

Your software release may not support all of the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions on Band Selection, 802.11 Bands, and Parameters

- Band-selection enabled WLANs do not support time-sensitive applications like voice and video because of roaming delays.
- Band selection can be used only with Cisco Aironet 1040, 1140, 1250, 1260, 3500, and the 3600 series access points.
- Band selection operates only on access points that are connected to a controller. A FlexConnect access point without a controller connection does not perform band selection after a reboot.

- The band-selection algorithm directs dual-band clients only from the 2.4-GHz radio to the 5-GHz radio of the same access point, and it only runs on an access point when both the 2.4-GHz and 5-GHz radios are up and running.
- You can enable both band selection and aggressive load balancing on the controller. They run independently and do not impact one another.
- It is not possible to enable or disable band selection and client load balancing globally through the controller GUI or CLI. You can, however, enable or disable band selection and client load balancing for a particular WLAN. Band selection and client load balancing are enabled globally by default.

Information About Configuring Band Selection, 802.11 Bands, and Parameters

Band Selection

Band selection enables client radios that are capable of dual-band (2.4- and 5-GHz) operation to move to a less congested 5-GHz access point. The 2.4-GHz band is often congested. Clients on this band typically experience interference from Bluetooth devices, microwave ovens, and cordless phones as well as co-channel interference from other access points because of the 802.11b/g limit of three nonoverlapping channels. To prevent these sources of interference and improve overall network performance, you can configure band selection on the switch.

Band selection is enabled globally by default.

Band selection works by regulating probe responses to clients. It makes 5-GHz channels more attractive to clients by delaying probe responses to clients on 2.4-GHz channels.

802.11 Bands

You can configure the 802.11b/g/n (2.4-GHz) and 802.11a/n (5-GHz) bands for the controller to comply with the regulatory requirements in your country. By default, both 802.11b/g/n and 802.11a/n are enabled.

When a controller is configured to allow only 802.11g traffic, 802.11b client devices are able to successfully connect to an access point but cannot pass traffic. When you configure the controller for 802.11g traffic only, you must mark 11g rates as mandatory.

802.11n Parameter

This section provides instructions for managing 802.11n devices such as the Cisco Aironet 1140 and 3600 Series Access Points on your network. The 802.11n devices support the 2.4- and 5-GHz bands and offer high-throughput data rates.

The 802.11n high-throughput rates are available on all 802.11n access points for WLANs using WMM with no Layer 2 encryption or with WPA2/AES encryption enabled.

802.11h Parameter

802.11h informs client devices about channel changes and can limit the transmit power of those client devices.

How to Configure 802.11 Bands and Parameters

Configuring Band Selection (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **wireless client band-select cycle-count** *cycle_count*
3. **wireless client band-select cycle-threshold** *milliseconds*
4. **wireless client band-select expire suppression** *seconds*
5. **wireless client band-select expire dual-band** *seconds*
6. **wireless client band-select client-rssi** *client_rssi*
7. **end**
8. **wlan** *wlan_profile_name wlan_ID SSID_network_name* **band-select**
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**wireless client band-select cycle-count** *cycle_count* **Example:** `Switch(config)# wireless client band-select cycle-count 3`	Sets the probe cycle count for band select. You can enter a value between 1 and 10 for the *cycle_count* parameter.
Step 3	**wireless client band-select cycle-threshold** *milliseconds* **Example:** `Switch(config)# wireless client band-select cycle-threshold 5000`	Sets the time threshold for a new scanning cycle period. You can enter a value for threshold between 1 and 1000 for the *milliseconds* parameter.
Step 4	**wireless client band-select expire suppression** *seconds* **Example:** `Switch(config)# wireless client band-select expire suppression 100`	Sets the suppression expire to the band select. You can enter a value for suppression between 10 to 200 for the *seconds* parameter.
Step 5	**wireless client band-select expire dual-band** *seconds*	Sets the dual band expire.

	Command or Action	Purpose
	Example: `Switch(config)# wireless client band-select expire dual-band 100`	You can enter a value for dual band between 10 and 300 for the *seconds* parameter.
Step 6	**wireless client band-select client-rssi** *client_rssi* **Example:** `Switch(config)# wireless client band-select client-rssi 40`	Sets the client RSSI threshold. You can enter a value for minimum dBm of a client RSSI to respond to a probe between 20 and 90 for the *client_rssi* parameter.
Step 7	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Step 8	**wlan** *wlan_profile_name wlan_ID SSID_network_name* **band-select**
	
	Example:
	`Switch(config)# wlan wlan1 25 ssid12`
	`Switch(config-wlan)# band-select`
		Configures band selection on specific WLANs.
		
		You can enter a value between 1 and 512 for the *wlan_ID* parameter.
		
		You can enter the up to 32 alphanumeric characters for *SSID_network_name* parameter.

Step 9	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring the 802.11 Bands (CLI)

You can configure 802.11 bands and parameters.

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 5ghz shutdown**
3. **ap dot11 24ghz shutdown**
4. **ap dot11 {5ghz | 24ghz} beaconperiod** *time_unit*
5. **ap dot11 {5ghz | 24ghz} fragmentation** *threshold*
6. **ap dot11 {5ghz | 24ghz} dtpc**
7. **wireless client association limit** *number* **interval** *milliseconds*
8. **ap dot11 {5ghz | 24ghz} rate** *rate {disable | mandatory | supported}*
9. **no ap dot11 5ghz shutdown**
10. **no ap dot11 24ghz shutdown**
11. **ap dot11 24ghz dot11g**
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 5ghz shutdown** **Example:** `Switch(config)# ap dot11 5ghz shutdown`	Disables the 802.11a band. **Note** You must disable the 802.11a band before configuring the 802.11a network parameters.
Step 3	**ap dot11 24ghz shutdown** **Example:** `Switch(config)# ap dot11 24ghz shutdown`	Disables the 802.11b band. **Note** You must disable the 802.11b band before configuring the 802.11b network parameters.
Step 4	**ap dot11 {5ghz | 24ghz} beaconperiod** *time_unit* **Example:** `Switch(config)# ap dot11 5ghz beaconperiod 500`	Specifies the rate at which the SSID is broadcast by the access point. The beacon interval is measured in time units (TUs). One TU is 1024 microseconds. You can configure the access point to send a beacon every 20 to 1000 milliseconds.
Step 5	**ap dot11 {5ghz | 24ghz} fragmentation** *threshold* **Example:** `Switch(config)# ap dot11 5ghz fragmentation 300`	Specifies the size at which packets are fragmented. The threshold is a value between 256 and 2346 bytes (inclusive). Specify a low number for areas where communication is poor or where there is a great deal of radio interference.

Step 6	**ap dot11 {5ghz | 24ghz} dtpc**
	
	Example:
	`Switch(config)# ap dot11 5ghz dtpc`
	
	`Switch(config)# no ap dot11 24ghz dtpc`
		Enables access points to advertise their channels and transmit the power levels in beacons, and probe responses.
		
		The default value is enabled. Client devices using dynamic transmit power control (DTPC) receive the channel and power level information from the access points and adjust their settings automatically. For example, a client device used primarily in Japan could rely on DTPC to adjust its channel and power settings automatically when it travels to Italy and joins a network there.
		Note On access points that run Cisco IOS software, this feature is called world mode.
		
		The **no** form of the command disables the 802.11a or 802.11b DTPC setting.

Step 7	**wireless client association limit** *number* **interval** *milliseconds* **Example:** `Switch(config)# wireless client association limit 50 interval 1000`	Specifies the maximum allowed clients that can be configured. You can configure a maximum number of association request on a single access point slot at a given interval. The range of association limit that you can configure is from one through 100. The association request limit interval is measured between 100 to 10000 milliseconds.

	Command or Action	Purpose
Step 8	**ap dot11 {5ghz | 24ghz} rate** *rate {disable | mandatory | supported}* **Example:** `Switch(config)# ap dot11 5ghz rate 36 mandatory`	Specifies the rate at which data can be transmitted between the controller and the client. • *disabled*—Defines that the clients specify the data rates used for communication. • *mandatory*—Defines that the clients support this data rate in order to associate to an access point on the controller. • *supported*—Any associated clients that support this data rate may communicate with the access point using that rate. However, the clients are not required to be able to use this rate in order to associate. • *rate*—Specifies the rate at which data is transmitted. For the 802.11a and 802.11b bands, the data is transmitted at the rate of 1, 2, 5.5, 6, 9, 11, 12, 18, 24, 36, 48, or 54 Mbps.
Step 9	**no ap dot11 5ghz shutdown** **Example:** `Switch(config)# no ap dot11 5ghz shutdown`	Enables the 802.11a band. **Note** The default value is enabled.
Step 10	**no ap dot11 24ghz shutdown** **Example:** `Switch(config)# no ap dot11 24ghz shutdown`	Enables the 802.11b band. **Note** The default value is enabled.
Step 11	**ap dot11 24ghz dot11g** **Example:** `Switch(config)# ap dot11 24ghz dot11g`	Enables or disables 802.11g network support. The default value is enabled. You can use this command only if the 802.11b band is enabled. If you disable this feature, the 802.11b band is enabled without 802.11g support.
Step 12	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring 802.11n Parameters (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 {5ghz | 24ghz} dot11n**
3. **ap dot11 {5ghz | 24ghz} dot11n mcs tx** *rtu*
4. **wlan***wlan_profile_name wlan_ID SSID_network_name* **wmm require**
5. **ap dot11 {5ghz | 24ghz} shutdown**
6. **{ap | no ap} dot11 {5ghz | 24 ghz} dot11n a-mpdu tx priority {all | 0-7}**
7. **no ap dot11 {5ghz | 24ghz} shutdown**
8. **ap dot11 {5ghz | 24ghz} dot11n guard-interval {any | long}**
9. **ap dot11 {5ghz | 24ghz} dot11n rifs rx**
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 {5ghz | 24ghz} dot11n** **Example:** `Switch(config)# ap dot11 5ghz dot11n`	Enables 802.11n support on the network. The **no** form of the command disables the 802.11n support on the network.
Step 3	**ap dot11 {5ghz | 24ghz} dot11n mcs tx** *rtu* **Example:** `Switch(config)# ap dot11 5ghz dot11n mcs tx 20`	Specifies the modulation and coding scheme (MCS) rates at which data can be transmitted between the access point and the client. You can set a value from 0 through 23 for the **mcs tx** parameter. The **no** form of the command disables the MCS rates that is configured.

Step 4	**wlan***wlan_profile_name wlan_ID SSID_network_name* **wmm require**
	
	Example:
	`Switch(config)# wlan wlan1 25 ssid12`
	`Switch(config-wlan)# wmm require`
		Enables WMM on the WLAN and uses the 802.11n data rates that you configured.
		The **require** parameter requires client devices to use WMM. Devices that do not support WMM cannot join the WLAN.

Step 5	**ap dot11 {5ghz | 24ghz} shutdown** **Example:** `Switch(config)# ap dot11 5ghz shutdown`	Disables the network.
Step 6	**{ap | no ap} dot11 {5ghz | 24 ghz} dot11n a-mpdu tx priority {all | 0-7}**	Specifies the aggregation method used for 802.11n packets. Aggregation is the process of grouping packet data frames together rather than transmitting them separately. Two aggregation methods are available:

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td></td><td>

Example:

```
Switch(config)# ap dot11 5ghz dot11n
a-mpdu tx priority all
```

</td><td>

Aggregated MAC Protocol Data Unit (A-MPDU) and Aggregated MAC Service Data Unit (A-MSDU). Both A-MPDU and A-MSDU are performed in the software.

You can specify the aggregation method for various types of traffic from the access point to the clients.

The following table defines the priority levels (0-7) assigned per traffic type.

Table 10: Traffic Type Priority Levels

<table>
<tr><th>User Priority</th><th>Traffic Type</th></tr>
<tr><td>0</td><td>Best effort</td></tr>
<tr><td>1</td><td>Background</td></tr>
<tr><td>2</td><td>Spare</td></tr>
<tr><td>3</td><td>Excellent effort</td></tr>
<tr><td>4</td><td>Controlled load</td></tr>
<tr><td>5</td><td>Video, less than 100-ms latency and jitter</td></tr>
<tr><td>6</td><td>Voice, less than 100-ms latency and jitter</td></tr>
<tr><td>7</td><td>Network control</td></tr>
</table>

You can configure each priority level independently, or you can use the all parameter to configure all of the priority levels at once. You can configure priority levels so that the traffic uses either A-MPDU transmission or A-MSDU transmission.

- When you use the **ap** command along with the other options, the traffic associated with that priority level uses A-MPDU transmission.
- When you use the **no ap** command along with the other options, the traffic associated with that priority level uses A-MSDU transmission.

  Configure the priority levels to match the aggregation method used by the clients. By default, A-MPDU is enabled for priority level 0, 4 and 5 and the rest are disabled. By default, A-MPDU is enabled for all priorities except 6 and 7.

</td></tr>
<tr><td>Step 7</td><td>no ap dot11 {5ghz | 24ghz} shutdown</td><td>Reenables the network.</td></tr>
</table>

	Command or Action	Purpose
	Example: `Switch(config)# no ap dot11 5ghz shutdown`	
Step 8	**ap dot11 {5ghz | 24ghz} dot11n guard-interval {any | long}** **Example:** `Switch(config)# ap dot11 5ghz dot11n guard-interval long`	Configures the guard interval for the network.
Step 9	**ap dot11 {5ghz | 24ghz} dot11n rifs rx** **Example:** `Switch(config)# ap dot11 5ghz dot11n rifs rx`	Configures the Reduced Interframe Space (RIFS) for the network.
Step 10	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring 802.11h Parameters (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 5ghz shutdown**
3. **{ap | no ap} dot11 5ghz channelswitch mode** *switch_mode*
4. **ap dot11 5ghz power-constraint** *value*
5. **no ap dot11 5ghz shutdown**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 5ghz shutdown**	Disables the 802.11a network.

	Command or Action	Purpose
	Example: `Switch(config)# ap dot11 5ghz shutdown`	
Step 3	**{ap | no ap} dot11 5ghz channelswitch mode** *switch_mode* **Example:** `Switch(config)# ap dot11 5ghz channelswitch mode 0`	Enables or disables the access point to announce when it is switching to a new channel. You can enter a 0 or 1 for the **channelswitch** parameter to specify whether transmissions are restricted until the actual channel switch (0) or are not restricted (1). The default value is disabled.
Step 4	**ap dot11 5ghz power-constraint** *value* **Example:** `Switch(config)# ap dot11 5ghz power-constraint 200`	Configures the 802.11h power constraint value in a range from zero through 255. The default value for the value parameter is 3 dB.
Step 5	**no ap dot11 5ghz shutdown** **Example:** `Switch(config)# no ap dot11 5ghz shutdown`	Reenables the 802.11a network.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Monitoring Configuration Settings for Band Selection, 802.11 Bands, and Parameters

Monitoring Configuration Settings Using Band Selection and 802.11 Bands Commands

This section describes the new commands for band selection and 802.11 bands.

The following commands can be used to monitor band selection, and 802.11 bands and parameters the switch.

Table 11: Monitoring Configuration Settings Using Band Selection and 802.11 Bands Commands

Command	Purpose
show ap dot11 5ghz network	Displays 802.11a bands network parameters, 802.11a operational rates, 802.11n MCS settings, and 802.11n status information.

show ap dot11 24ghz network	Displays 802.11b bands network parameters, 802.11b/g operational rates, 802.11n MCS settings, and 802.11n status information.
show wireless dot11h	Displays 802.11h configuration parameters.
show wireless band-select	Displays band select configuration settings.

Example: Viewing the Configuration Settings for 5-GHz Band

```
Switch# show ap dot11 5ghz network
802.11a Network : Enabled
11nSupport : Enabled
  802.11a Low Band : Enabled
  802.11a Mid Band : Enabled
  802.11a High Band : Enabled

802.11a Operational Rates
  802.11a 6M : Mandatory
  802.11a 9M : Supported
  802.11a 12M : Mandatory
  802.11a 18M : Supported
  802.11a 24M : Mandatory
  802.11a 36M : Supported
  802.11a 48M : Supported
  802.11a 54M : Supported
802.11n MCS Settings:
  MCS 0 : Supported
  MCS 1 : Supported
  MCS 2 : Supported
  MCS 3 : Supported
  MCS 4 : Supported
  MCS 5 : Supported
  MCS 6 : Supported
  MCS 7 : Supported
  MCS 8 : Supported
  MCS 9 : Supported
  MCS 10 : Supported
  MCS 11 : Supported
  MCS 12 : Supported
  MCS 13 : Supported
  MCS 14 : Supported
  MCS 15 : Supported
  MCS 16 : Supported
  MCS 17 : Supported
  MCS 18 : Supported
  MCS 19 : Supported
  MCS 20 : Supported
  MCS 21 : Supported
  MCS 22 : Supported
  MCS 23 : Supported
802.11n Status:
  A-MPDU Tx:
    Priority 0 : Enabled
    Priority 1 : Disabled
    Priority 2 : Disabled
    Priority 3 : Disabled
    Priority 4 : Enabled
    Priority 5 : Enabled
    Priority 6 : Disabled
    Priority 7 : Disabled
  A-MSDU Tx:
    Priority 0 : Enabled
```

```
    Priority 1 : Enabled
    Priority 2 : Enabled
    Priority 3 : Enabled
    Priority 4 : Enabled
    Priority 5 : Enabled
    Priority 6 : Disabled
    Priority 7 : Disabled
Guard Interval : Any
  Rifs Rx : Enabled
Beacon Interval : 100
CF Pollable mandatory : Disabled
CF Poll Request Mandatory : Disabled
CFP Period : 4
CFP Maximum Duration : 60
Default Channel : 36
Default Tx Power Level : 1
DTPC Status : Enabled
Fragmentation Threshold : 2346
Pico-Cell Status : Disabled
Pico-Cell-V2 Status : Disabled
TI Threshold : 0
Legacy Tx Beamforming setting : Disabled
Traffic Stream Metrics Status : Disabled
Expedited BW Request Status : Disabled
EDCA profile type check : default-wmm
Call Admision Control (CAC) configuration
Voice AC
  Voice AC - Admission control (ACM) : Disabled
  Voice Stream-Size : 84000
  Voice Max-Streams : 2
  Voice Max RF Bandwidth : 75
  Voice Reserved Roaming Bandwidth : 6
  Voice Load-Based CAC mode : Enabled
  Voice tspec inactivity timeout : Enabled
CAC SIP-Voice configuration
  SIP based CAC : Disabled
  SIP Codec Type : CODEC_TYPE_G711
  SIP call bandwidth : 64
  SIP call bandwith sample-size : 20
Video AC
  Video AC - Admission control (ACM) : Disabled
  Video max RF bandwidth : Infinite
  Video reserved roaming bandwidth : 0
```

Example: Viewing the Configuration Settings for 24-GHz Band

```
Switch# show ap dot11 24ghz network
802.11b Network : Enabled
11gSupport : Enabled
11nSupport : Enabled

  802.11b/g Operational Rates
  802.11b 1M : Mandatory
  802.11b 2M : Mandatory
  802.11b 5.5M : Mandatory
  802.11g 6M : Supported
  802.11g 9M : Supported
  802.11b 11M : Mandatory
  802.11g 12M : Supported
  802.11g 18M : Supported
  802.11g 24M : Supported
  802.11g 36M : Supported
  802.11g 48M : Supported
  802.11g 54M : Supported
802.11n MCS Settings:
  MCS 0 : Supported
  MCS 1 : Supported
  MCS 2 : Supported
  MCS 3 : Supported
  MCS 4 : Supported
```

```
  MCS 5 : Supported
  MCS 6 : Supported
  MCS 7 : Supported
  MCS 8 : Supported
  MCS 9 : Supported
  MCS 10 : Supported
  MCS 11 : Supported
  MCS 12 : Supported
  MCS 13 : Supported
  MCS 14 : Supported
  MCS 15 : Supported
  MCS 16 : Supported
  MCS 17 : Supported
  MCS 18 : Supported
  MCS 19 : Supported
  MCS 20 : Supported
  MCS 21 : Supported
  MCS 22 : Supported
  MCS 23 : Supported
802.11n Status:
  A-MPDU Tx:
    Priority 0 : Enabled
    Priority 1 : Disabled
    Priority 2 : Disabled
    Priority 3 : Disabled
    Priority 4 : Enabled
    Priority 5 : Enabled
    Priority 6 : Disabled
    Priority 7 : Disabled
  A-MSDU Tx:
    Priority 0 : Enabled
    Priority 1 : Enabled
    Priority 2 : Enabled
    Priority 3 : Enabled
    Priority 4 : Enabled
    Priority 5 : Enabled
  Priority 6 : Disabled
    Priority 7 : Disabled
  Guard Interval : Any
  Rifs Rx : Enabled
Beacon Interval : 100
CF Pollable Mandatory : Disabled
CF Poll Request Mandatory : Disabled
CFP Period : 4
CFP Maximum Duration : 60
Default Channel : 11
Default Tx Power Level : 1
DTPC Status : true
Call Admission Limit : 105
G711 CU Quantum : 15
ED Threshold : -50
Fragmentation Threshold : 2346
PBCC Mandatory : Disabled
Pico-Cell Status : Disabled
Pico-Cell-V2 Status : Disabled
RTS Threshold : 2347
Short Preamble Mandatory : Enabled
Short Retry Limit : 7
Legacy Tx Beamforming setting : Disabled
Traffic Stream Metrics Status : Disabled
Expedited BW Request Status : Disabled
EDCA profile type : default-wmm
Call Admision Control (CAC) configuration
Voice AC
  Voice AC - Admission control (ACM) : Disabled
  Voice Stream-Size : 84000
  Voice Max-Streams : 2
  Voice Max RF Bandwidth : 75
  Voice Reserved Roaming Bandwidth : 6
  Voice Load-Based CAC mode : Enabled
  Voice tspec inactivity timeout : Enabled
CAC SIP-Voice configuration
  SIP based CAC : Disabled
```

```
  SIP Codec Type : CODEC_TYPE_G711
  SIP call bandwidth : 64
  SIP call bandwith sample-size : 20
Video AC
  Video AC - Admission control (ACM) : Disabled
  Video max RF bandwidth : Infinite
  Video reserved roaming bandwidth : 0
```

Example: Viewing the status of 802.11h Parameters

```
Switch# show wireless dot11h
Power Constraint: 0
Channel Switch: 0
Channel Switch Mode: 0
```

Example: Verifying the Band Selection Settings

```
Switch# show wireless band-select
Band Select Probe Response    : per WLAN enabling
Cycle Count                  : 2
Cycle Threshold (millisec)   : 200
Age Out Suppression (sec)    : 20
Age Out Dual Band (sec)      : 60
Client RSSI (dBm)            : 80
```

Configuration Examples for Band Selection, 802.11 Bands, and Parameters

Examples: Band Selection Configuration

This example shows how to set the probe cycle count and time threshold for a new scanning cycle period for band select:

```
Switch# configure terminal
Switch(config)# wireless client band-select cycle-count 3
Switch(config)# wireless client band-select cycle-threshold 5000
Switch(config)# end
```

This example shows how to set the suppression expire to the band select:

```
Switch# configure terminal
Switch(config)# wireless client band-select expire suppression 100
Switch(config)# end
```

This example shows how to set the dual band expire for the band select:

```
Switch# configure terminal
Switch(config)# wireless client band-select expire dual-band 100
Switch(config)# end
```

This example shows how to set the client RSSI threshold for the band select:

```
Switch# configure terminal
Switch(config)# wireless client band-select client-rssi 40
Switch(config)# end
```

This example shows how to configure band selection on specific WLANs:

```
Switch# configure terminal
Switch(config)# wlan wlan1 25 ssid12
```

```
Switch(config-wlan)# band-select
Switch(config)# end
```

Examples: 802.11 Bands Configuration

This example shows how to configure 802.11 bands using beacon interval, fragmentation, and dynamic transmit power control:

```
Switch# configure terminal
Switch(config)# ap dot11 5ghz shutdown
Switch(config)# ap dot11 24ghz shutdown
Switch(config)# ap dot11 5ghz beaconperiod 500
Switch(config)# ap dot11 5ghz fragmentation 300
Switch(config)# ap dot11 5ghz dtpc
Switch(config)# wireless client association limit 50 interval 1000
Switch(config)# ap dot11 5ghz rate 36 mandatory
Switch(config)# no ap dot11 5ghz shutdown
Switch(config)# no ap dot11 24ghz shutdown
Switch(config)# ap dot11 24ghz dot11g
Switch(config)#end
```

Examples: 802.11n Configuration

This example shows how to configure 802.11n parameters for 5-GHz band using aggregation method:

```
Switch# configure terminal
Switch(config)# ap dot11 5ghz dot11n
Switch(config)# ap dot11 5ghz dot11n mcs tx 20
Switch(config)# wlan wlan1 25 ssid12
Switch(config-wlan)# wmm require\
Switch(config-wlan)# exit
Switch(config)# ap dot11 5ghz shutdown
Switch(config)# ap dot11 5ghz dot11n a-mpdu tx priority all
Switch(config)# no ap dot11 5ghz shutdown
Switch(config)#exit
```

This example shows how to configure the guard interval for 5-GHz band:

```
Switch# configure terminal
Switch(config)# ap dot11 5ghz dot11n
Switch(config)# ap dot11 5ghz dot11n mcs tx 20
Switch(config)# wlan wlan1 25 ssid12
Switch(config-wlan)# wmm require\
Switch(config-wlan)# exit
Switch(config)# no ap dot11 5ghz shutdown
Switch(config)# ap dot11 5ghz dot11n guard-interval long
Switch(config)#end
```

This example shows how to configure the RIFS for 5-GHz band:

```
Switch# configure terminal
Switch(config)# ap dot11 5ghz dot11n
Switch(config)# ap dot11 5ghz dot11n mcs tx 20
Switch(config)# wlan wlan1 25 ssid12
Switch(config-wlan)# wmm require\
Switch(config-wlan)# exit
Switch(config)# ap dot11 5ghz shutdown
Switch(config)# ap dot11 5ghz dot11n rifs rx
Switch(config)#end
```

Examples: 802.11h Configuration

This example shows how to configure the access point to announce when it is switching to a new channel using restriction transmission:

```
Switch# configure terminal
Switch(config)# ap dot11 5ghz shutdown
Switch(config)# ap dot11 5ghz channelswitch mode 0
Switch(config)# no ap dot11 5ghz shutdown
Switch(config)#end
```

This example shows how to configure the 802.11h power constraint for 5-GHz band:

```
Switch# configure terminal
Switch(config)# ap dot11 5ghz shutdown
Switch(config)# ap dot11 5ghz power-constraint 200
Switch(config)# no ap dot11 5ghz shutdown
Switch(config)#end
```

Additional References for 802.11 Parameters and Band Selection

Related Documents

Related Topic	Document Title
System management commands	*System Management Command Reference, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information For Performing 802.11 parameters and Band Selection Configuration

Release	Feature Information
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 7

Configuring Aggressive Load Balancing

Finding Feature Information

Your software release may not support all of the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for Aggressive Load Balancing

- You can configure aggressive load balancing only from the command-line interface.
- Aggressive load balancing is disabled by default, you must enable it manually.
- You can enable load balancing either separately or together with the band select configurations.
- When the band select is enabled on the dual-band clients, the load balancing parameter selects only the lowest load radio from 5-GHz radios. For the 2.4-GHz clients, there is no probe information of the client on 5 GHz and therefore the load balancing algorithm can only be selected between radio on 2.4 GHz.
- You can operate load balancing of clients between access points on the same switch but not for the clients between access points on the different switch.

- The load balancing uses an existing association denial mechanism based on the number of client on the radio and the band select is implemented by the distributed probe response suppression on the access point only.

Information for Configuring Aggressive Load Balancing Parameters

Aggressive Load Balancing

Enabling aggressive load balancing on the controller allows lightweight access points to load balance wireless clients across access points. You can enable aggressive load balancing using the controller.

When a wireless client attempts to associate to a lightweight access point, association response packets are sent to the client with an 802.11 response packet including status code 17. The code 17 indicates that the AP is busy. The AP responds with an association response bearing 'success' if the AP threshold is not met, and with code 17 (AP busy) if the AP utilization threshold is reached or exceeded and another less busy AP heard the client request.

For example, if the number of clients on AP1 is more than the number of clients on AP2 plus the load-balancing window, then AP1 is considered to be busier than AP2. When a client attempts to associate to AP1, it receives an 802.11 response packet with status code 17, indicating that the access point is busy, and the client attempts to associate to a different access point.

You can configure the controller to deny client associations up to 10 times (if a client attempted to associate 11 times, it would be allowed to associate on the 11th try). You can also enable or disable load balancing on a particular WLAN, which is useful if you want to disable load balancing for a select group of clients (such as time-sensitive voice clients).

The maximum number of client associations that the access points can support is dependent upon the following factors:

- The maximum number of client associations differs for lightweight and autonomous Cisco IOS access points.
- There may be a limit per radio and an overall limit per AP.
- AP hardware (the 16-MB APs have a lower limit than the 32-MB and higher APs)

The Client Association Limits for Lightweight Access Points are as follows:

- For 16-MB APs, the limit is 128 clients per AP. This limit is applicable to 1100 and 1200 series APs.
- For 32-MB and higher APs, there is no per-AP limit.

The maximum Client Association Limits per-radio for all of the Cisco IOS APs is 200 associations.

Note With 32-MB and higher lightweight Cisco IOS APs, with two radios, up to 200 + 200 = 400 associations are supported.

The maximum Client Association Limits per Autonomous Cisco IOS access point is around 80 to 127 clients per AP. This number varies depending on the following factors:

- AP model (whether it is 16 MB or 32 MB or higher)

- Cisco IOS software release
- Hardware configuration (two radios use more memory than one)
- Enabled features (WDS functionality in particular)

The per-radio limit is about 200 associations. One association will likely hit the per-AP limit first. Unlike Cisco Unified Wireless Network, autonomous Cisco IOS supports per-SSID/per-AP association limits. This limit is configured using the max-associations CLI, under dot11 SSID. The maximum number is 255 associations (which is also the default number).

How to Configure Aggressive Load Balancing

Configuring Aggressive Load Balancing

SUMMARY STEPS

1. **configure terminal**
2. **wireless load-balancing window** *client-count*
3. **wireless load-balancing denial** *denial-count*
4. **end**
5. **wlan** *wlan_profile_name wlan_ID SSID_network_name* **load-balance**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**wireless load-balancing window** *client-count* **Example:** `Switch(config)# wireless load-balancing window 1`	Sets the client window for aggressive load balancing. You can enter a value between 0 and 20 for the *client_count* parameter.
Step 3	**wireless load-balancing denial** *denial-count* **Example:** `Switch(config)# wireless load-balancing denial-count 1`	Sets the denial count for load balancing. You can enter a value between 0 and 10 for the *denial_count* parameter.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

	Command or Action	Purpose

Step 5	**wlan** *wlan_profile_name wlan_ID SSID_network_name* **load-balance**
	
	Example:
	`Switch(config)# wlan wlan1 25 ssid12`
	`Switch(config-wlan)# load-balance`
		Enables or disables aggressive load balancing on specific WLANs.
		
		You can enter a value between 1 and 512 for the *wlan_ID* parameter.
		
		You can enter the up to 32 alphanumeric characters for *SSID_network_name* parameter.

Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Monitoring Aggressive Load Balancing

This section describes the new command for aggressive load balancing.

The following command can be used to monitor aggressive load balancing on the switch.

Table 12: Monitoring Aggressive Load Balancing Command

Command	Purpose
show wireless load-balancing	Displays the status of the load-balancing feature.

Examples: Aggressive Load Balancing Configuration

This example shows how to configure the load balancing denial count:

```
Switch# configure terminal
Switch(config)# wireless load-balancing denial-count 1
Switch(config)# end
Switch# show wireless load-balancing
```

This example shows how to configure the client window for aggressive load balancing:

```
Switch# configure terminal
Switch(config)# wireless load-balancing window 1
Switch(config)# end
Switch# show wireless load-balancing
```

This example shows how to configure load balancing on specific WLAN:

```
Switch# configure terminal
Switch(config)# wlan wlan1 25 ssid12
Switch(config-wlan)# load-balance
Switch(config)# end
Switch# show wireless load-balancing
```

Additional References for Aggressive Load Balancing

Related Documents

Related Topic	Document Title
System management commands	*System Management Command Reference Guide, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information For Performing Aggressive Load Balancing Configuration

Release	Feature Information
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 8

Configuring Client Roaming

Finding Feature Information

Your software release may not support all of the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for Configuring Client Roaming

The following are the restrictions that you should be aware while configuring client roaming:

- Cisco Compatible Extensions (CCX) support is enabled automatically for every WLAN on the switch and cannot be disabled. The switch stores the CCX version of the client in its client database and uses it to generate and respond to CCX frames appropriately. Clients must support CCXv4 or v5 (or CCXv2 for access point assisted roaming) to utilize these roaming enhancements.
- Client roaming between 600 Series Access points is not supported.

Information About Client Roaming

The controllers deliver high-end wireless services to the clients roaming across wireless network. Now, the wireless services are integrated with the switches, thus delivering a value-added Cisco unified new mobility architecture. This unified architecture enables client-roaming services to both wireless and wired clients with seamless, fast- roaming services.

The new mobility architecture supports fast client roaming services using logical categorization of network into Mobility Domains (MDs), Mobility Groups (MGs), Mobility Subdomains (MSDs), and Switch Peer Groups (SPGs) using systems such as Mobility Oracle (MO), Mobility Controller (MC), and Mobility Agent (MA).

- A **Mobility Domain** is the entire domain across which client roaming is supported. It is a collection of mobility groups. For example, a campus network can be considered as a mobility domain.
- A **Mobility Group** is a collection of mobility subdomains across which fast roaming is supported. The mobility group can be one or more buildings within a campus across which frequent roaming is supported.
- A **Mobility Subdomain** is an autonomous portion of the mobility domain network. Each mobility subdomain contains one mobility controller (MC) and a collection of SPGs. A subdomain is equivalent to an 802.11r key domain.
- A **Switch Peer Group** is a collection of mobility agents.
- The **Mobility Oracle** acts as the point of contact for mobility events that occur across mobility subdomains. The mobility oracle also maintains a local database of each client in the entire mobility domain, their home and current subdomain. There is only one MO for an entire mobility domain. The Cisco WLC 5700 Series Controllers or Cisco Unified Wireless Networking Solution controller can act as MO.
- The **Mobility Controller** provides mobility management services for inter-SPG roaming events. The MC sends the configuration like SPG name and SPG peer member list to all of the mobility agents under its subdomain. The Cisco WLC 5700 Series Controllers, Cisco Catalyst 3850 Switch, or Cisco Unified Wireless Networking Solution controller can act as MC. The MC has MC functionality and MA functionality that is running internally into it.
- The **Mobility Agent** is the component that maintains client mobility state machine for a mobile client. All APs are connected to the mobility agent.

The New mobility architecture supports seamless roaming in the following scenarios:

- Intra-switch roaming—The client roaming between APs managed by same mobility agent.
- Intra-SPG roaming—The client roaming between mobility agents in the same SPG.
- Inter-SPG, Intra-subdomain roaming—The client roaming between mobility agents in different SPGs within the same subdomain.
- Inter-subdomain roaming—The client roaming between mobility agents across a subdomain.

Fast Roaming

New mobility architecture supports fast roaming when clients roam within a mobility group by eliminating the need for full authentication. Security polices should be same across the switches for fast roaming.

Local, anchor, foreign MAs and MCs

When a client joins an MA initially and its point of attachment has not changed, that MA is referred as local or associated MA. The MC to which this MA is associated is referred as local or associated MC.

When a client roams between two MAs, the MA to which the client was previously associated is the anchor MA (point of attachment) and the MA to which the client is currently associated is the foreign or associated MA (point of presence). The MCs to which these MAs are associated are referred as anchor, foreign, or associated MCs, respectively.

Inter-Subnet Roaming

Multiple-controller deployments support client roaming across access points managed by controllers in the same mobility group on different subnets. This roaming is transparent to the client because the session is sustained and a tunnel between the controllers allows the client to continue using the same DHCP-assigned or client-assigned IP address as long as the session remains active. The tunnel is torn down, and the client must reauthenticate when the client sends a DHCP Discover with a 0.0.0.0 client IP address or a 169.254.*.* client auto-IP address or when the operator-set user timeout is exceeded.

Voice-over-IP Telephone Roaming

802.11 voice-over-IP (VoIP) telephones actively seek out associations with the strongest RF signal to ensure the best quality of service (QoS) and the maximum throughput. The minimum VoIP telephone requirement of 20-millisecond or shorter latency time for the roaming handover is easily met by the Cisco Unified Wireless Network (Cisco UWN) solution, which has an average handover latency of 5 or fewer milliseconds when open authentication is used. This short latency period is controlled by controllers rather than allowing independent access points to negotiate roaming handovers.

The Cisco UWN solution supports 802.11 VoIP telephone roaming across lightweight access points managed by controllers on different subnets, as long as the controllers are in the same mobility group. This roaming is transparent to the VoIP telephone because the session is sustained and a tunnel between controllers allows the VoIP telephone to continue using the same DHCP-assigned IP address as long as the session remains active. The tunnel is torn down, and the VoIP client must reauthenticate when the VoIP telephone sends a DHCP Discover with a 0.0.0.0 VoIP telephone IP address or a 169.254.*.* VoIP telephone auto-IP address or when the operator-set user timeout is exceeded.

CCX Layer 2 Client Roaming

The controller supports five CCX Layer 2 client roaming enhancements:

- Access point assisted roaming—This feature helps clients save scanning time. When a CCXv2 client associates to an access point, it sends an information packet to the new access point listing the characteristics of its previous access point. Roaming time decreases when the client recognizes and uses an access point list built by compiling all previous access points to which each client was associated and sent (unicast) to the client immediately after association. The access point list contains the channels, BSSIDs of neighbor access points that support the client's current SSID(s), and time elapsed since disassociation.
- Enhanced neighbor list—This feature focuses on improving a CCXv4 client's roam experience and network edge performance, especially when servicing voice applications. The access point provides its associated client information about its neighbors using a neighbor-list update unicast message.
- Enhanced neighbor list request (E2E)—The End-2-End specification is a Cisco and Intel joint program that defines new protocols and interfaces to improve the overall voice and roaming experience. It applies

only to Intel clients in a CCX environment. Specifically, it enables Intel clients to request a neighbor list at will. When this occurs, the access point forwards the request to the controller. The controller receives the request and replies with the current CCX roaming sublist of neighbors for the access point to which the client is associated.

Note To see whether a particular client supports E2E, choose **Wireless** > **Clients** on the controller GUI, click the **Detail** link for the desired client, and look at the E2E Version text box in the Client Properties area.

- Roam reason report**—**This feature enables CCXv4 clients to report the reason why they roamed to a new access point. It also allows network administrators to build and monitor a roam history.
- Directed roam request**—**This feature enables the controller to send directed roam requests to the client in situations when the controller can better service the client on an access point different from the one to which it is associated. In this case, the controller sends the client a list of the best access points that it can join. The client can either honor or ignore the directed roam request. Non-CCX clients and clients running CCXv3 or below must not take any action. No configuration is required for this feature.

How to Configure Layer 2 or Layer 3 Roaming

Configuring Layer 2 or Layer 3 Roaming

Before You Begin

To configure the mobility agent for Layer 2 or Layer 3 roaming, the following requisites should be considered:

- SSID and security polices should be same across MAs for Layer 2 and Layer 3 roaming.
- Client VLAN ID should be same for Layer 2 roaming and different for Layer 3 roaming.
- Bridge domain ID and client VLAN IDs should be same for Layer 2 roaming. Either one or both of the bridge domain ID and client VLAN ID should be different for Layer 3 roaming.

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan_profile_name wlan_ID SSID_network_name*
3. **no mobility anchor sticky**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *wlan_profile_name wlan_ID SSID_network_name* **Example:** `Switch(config)#wlan wlan1`	Enters WLAN configuration mode.
Step 3	**no mobility anchor sticky** **Example:** `Switch(config-wlan)#no mobility anchor sticky`	(Optional) Disables Layer 2 anchoring.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring CCX Client Roaming Parameters (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 {5ghz | 24ghz} l2roam rf-params** {**default | custom** *min-rssi roam-hyst scan-thresh trans-time*}
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 {5ghz | 24ghz} l2roam rf-params** {**default | custom** *min-rssi roam-hyst scan-thresh trans-time*}	Configures CCX Layer 2 client roaming parameters. To choose the default RF parameters, enter the **default** option. To fine-tune the RF parameters that affect client roaming, enter the **custom** option and then enter any one of the following options:

	Command or Action	Purpose
	Example: `Switch#`**`ap dot11 5ghz l2roam rf-params custom -80`**	• Minimum RSSI—Indicates minimum Received Signal Strength Indicator (RSSI) required for the client to associate to an access point. If the client's average received signal power dips below this threshold, reliable communication is usually impossible. Therefore, clients must already have found and roamed to another access point with a stronger signal before the minimum RSSI value is reached. You can configure the minimum RSSI range from –80 through –90 dBm and the default is –85 dBm. • Hysteresis—Indicates how much greater the signal strength of a neighboring access point must be for the client to roam to it. This parameter is intended to reduce the amount of roaming between access points if the client is physically located on or near the border between two access points. You can configure the hysteresis range from 3 through 20 dB and the default is 3 dB. • Scan Threshold—Indicates a minimum RSSI that is allowed before the client should roam to a better access point. When the RSSI drops below the specified value, the client must be able to roam to a better access point within the specified transition time. This parameter also provides a power-save method to minimize the time that the client spends in active or passive scanning. For example, the client can scan slowly when the RSSI is above the threshold and scan more rapidly when the RSSI is below the threshold. You can configure the RSSI range from –70 through –77 dBm and the default value is –72 dBm. • Transition Time—Indicates the maximum time allowed for the client to detect a suitable neighboring access point to roam to and to complete the roam, whenever the RSSI from the client's associated access point is below the scan threshold. The Scan Threshold and Transition Time parameters guarantee a minimum level of client roaming performance. Together with the highest expected client speed and roaming hysteresis, these parameters make it possible to design a wireless LAN network that supports roaming simply by ensuring a certain minimum overlap distance between access points. You can configure the time period in the range from 1 through 10 seconds and the default time is 5 seconds.
Step 3	**end** **Example:** `Switch(config)#` **`end`**	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Mobility Oracle

SUMMARY STEPS

1. **configure terminal**
2. **wireless mobility oracle**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**wireless mobility oracle** **Example:** `Switch(config)# wireless mobility oracle`	Enables mobility oracle on the controller.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Mobility Controller

SUMMARY STEPS

1. **configure terminal**
2. **wireless mobility controller**
3. **wireless mobility controller peer-group** *switch-peer-group-name*
4. **wireless mobility controller peer-group** *switch-peer-group-name* **member ip** *ip-address* {**public-ip** *public-ip-address*}
5. **wireless mobility controller peer-group** *switch-peer-group-name* **multicast**
6. **wireless mobility controller peer-group** *switch-peer-group-name* **multicast ip** *peer-group-multicast-ip-addr*
7. **wireless mobility controller peer-group***switch-peer-group-name* **bridge-domain-id** *id*
8. **wireless mobility group member ip** *ip-address* [**public-ip** *public-ip-address*] [**group** *group-name*]
9. **wireless mobility dscp** *value*
10. **wireless mobility group keepalive** {*count* | *interval*}
11. **wireless mobility group name** *name*
12. **wireless mobility oracle ip***mo-ip-address*
13. **wireless management interface** *interface-name*
14. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**wireless mobility controller** **Example:** `Switch(config)# wireless mobility controller`	Enables wireless mobility controller.
Step 3	**wireless mobility controller peer-group** *switch-peer-group-name* **Example:** `Switch(config)# wireless mobility controller peer-group SPG1`	Configures a switch peer group name. You can enter up to 31 case-sensitive ASCII printable characters for the group name. Spaces are not allowed in mobility group. **Note** The **No** form of the command deletes the switch peer group.
Step 4	**wireless mobility controller peer-group** *switch-peer-group-name* **member ip** *ip-address* {**public-ip** *public-ip-address*}	Adds a mobility group member to a switch peer group. **Note** The **No** form of the command deletes the member from the switch peer group.

	Command or Action	Purpose
	Example: `Switch(config)# wireless mobility controller peer-group SPG1 member ip 10.0.0.1`	
Step 5	**wireless mobility controller peer-group** *switch-peer-group-name* **multicast** **Example:** `Switch(config)# wireless mobility controller peer-group SPG1 multicast`	Configures the multicast mode within a switch peer group.
Step 6	**wireless mobility controller peer-group** *switch-peer-group-name* **multicast ip** *peer-group-multicast-ip-addr* **Example:** `Switch(config)# wireless mobility controller peer-group SPG1 multicast ip 10.0.0.4`	Configures the multicast IP address for a switch peer group. **Note** The **No** form of the command deletes the multicast IP for the switch peer group.
Step 7	**wireless mobility controller peer-group***switch-peer-group-name* **bridge-domain-id** *id* **Example:** `Switch(config)# wireless mobility controller peer-group SPG bridge-domain-id 10.0.0.5`	Configures the bridge domain ID for a switch peer group. The default is zero. **Note** The **No** form of command sets the bridge domain ID to the default value.
Step 8	**wireless mobility group member ip** *ip-address* [**public-ip** *public-ip-address*] [**group** *group-name*] **Example:** `Switch(config)# wireless mobility group member ip 10.0.0.1`	Adds a mobility group member. **Note** The **No** form of the command removes the member from the group. The default group name is the group name of MC.
Step 9	**wireless mobility dscp** *value* **Example:** `Switch(config)# wireless mobility dscp 46`	Sets the DSCP value for mobility control packet. You can configure the DSCP value in a range from 0 through 63. The default value is 46.
Step 10	**wireless mobility group keepalive** {*count* | *interval*} **Example:** `Switch(config)# wireless mobility group keepalive count`	Configures the wireless mobility group keepalive count which is the number of keepalive retries before a member status is termed DOWN and keepalive interval which is interval between two keepalives.
Step 11	**wireless mobility group name** *name* **Example:** `Switch(config)# wireless mobility group name group1`	Specifies the case sensitive wireless mobility group name which can be ASCII printable string up to 31 characters.

	Command or Action	Purpose
Step 12	**wireless mobility oracle ip***mo-ip-address* **Example:** `Switch(config)# wireless mobility oracle ip 10.0.0.5`	Configures the mobility oracle IP address.
Step 13	**wireless management interface** *interface-name* **Example:** `Switch(config)# wireless management interface Vlan21`	Configures the wireless management interface.
Step 14	**end** **Example:** `Switch(config)#` **`end`**	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Mobility Agent

SUMMARY STEPS

1. **configure terminal**
2. **wireless mobility controller ip** *ip-address*
3. **wireless mobility load-balance**
4. **wireless mobility load-balance threshold** *threshold -value*
5. **wireless management interface** *interface-name*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch#` **`configure terminal`**	Enters global configuration mode.
Step 2	**wireless mobility controller ip** *ip-address* **Example:** `Switch(config)# wireless mobility controller ip 10.10.10.20`	Sets the IP address of the mobility controller.
Step 3	**wireless mobility load-balance**	Configures wireless mobility load balancing.

	Command or Action	Purpose
	Example: `Switch(config)# wireless mobility load-balance`	
Step 4	**wireless mobility load-balance threshold** *threshold -value* **Example:** `Switch(config)# wireless mobility load-balance threshold 100`	Configures the number of clients that can be local or anchored on the MA. You can configure the threshold value in a range from 100 to 2000. The default value is 1000.
Step 5	**wireless management interface** *interface-name* **Example:** `Switch(config)# wireless management interface Vlan21`	Configures wireless management interface for the mobility agent.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Monitoring Client Roaming Parameters

This section describes the new commands for the client parameters.

The following commands can be used to monitor the client roaming parameters on the switch.

Table 13: Monitoring Client Roaming Parameters Commands

Command	Purpose
show ap dot11 {5ghz | 24ghz} l2roam rf-param	Displays the current RF parameters configured for client roaming for the 802.11a or 802.11b/g network.
show ap dot11 {5ghz | 24ghz} l2roam statistics	Displays the CCX Layer 2 client roaming statistics for the 802.11a or 802.11b/g network.
show ap dot11 {5ghz | 24ghz} l2roam mac-address *mac-address* **statistics**	Displays the CCX Layer 2 client roaming statistics for a particular access point.

Monitoring Mobility Configurations

This section describes the new commands for monitoring mobility configurations.

The following command can be used to monitor mobility configurations on the Mobility Oracle, Mobility Controller, and Mobility Agent.

Table 14: Monitoring Mobility Configuration Commands on the Mobility Controller and Mobility Agent

Command	Purpose
show wireless mobility summary	Displays the summary information for the Mobility Controller and Mobility Agent.
show wireless mobility statistics	Displays mobility statistics.
show wireless mobility dtls connections	Displays established DTLS connections.

Table 15: Monitoring Mobility Configuration Commands on the Mobility Oracle

Command	Purpose
show wireless mobility oracle summary	Displays the status of the Mobility Controllers known to the Mobility Oracle.
show wireless mobility oracle client summary	Displays the information of a list of clients in the Mobility Oracle database.
show wireless mobility oracle client detail *client -mac-address*	Displays the detailed information of a particular client in the Mobility Oracle database.
show wireless mobility oracle *mc-ip*	Displays the information of a list of clients in the Mobility Oracle database that are anchored or associated to a specified Mobility Controller.

Table 16: Monitoring Mobility Configuration Commands on the Mobility Controller

Command	Purpose
show wireless mobility controller client summary	Displays a list of clients in the subdomain.
show wireless mobility controller client *mac-address* **detail**	Displays detailed information for a client in a subdomain.
show wireless mobility agent *ma-ip* **client summary**	Displays a list of clients anchored or associated to a specified Mobility Agent.
show wireless mobility ap-list	Displays the list of Cisco APs known to the mobility group.

Table 17: Monitoring Mobility Configuration Commands on the Mobility Agent

Command	Purpose
show wireless mobility load-balance summary	Displays the summary of mobility load-balance properties.

Additional References for Configuring Client Roaming

Related Documents

Related Topic	Document Title
Mobility configuration	*Mobility Configuration Guide, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*
Mobility-related commands	*Mobility Command Reference Guide, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information For Performing Client Roaming Configuration

Release	Feature Information
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 9

Configuring Voice and Video Parameters

Finding Feature Information

Your software release may not support all of the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Voice and Video Parameters

You can confirm the following points before configuring voice and video parameters:

- Ensure that the switch has access points connected to it.
- Configure SSID.

Restrictions for Voice and Video Parameters

The following are the restrictions that you should keep in mind while configuring voice and video parameters:

- SIP CAC can be used for the 9971 Cisco phones that support TSPEC-based admission control. You can also use the phones that support Status code 17.
- SIP snooping is supported for providing voice priority to the non-TSPEC SIP phones.
- TSPEC for video CAC is not supported.

Information About Configuring Voice and Video Parameters

Three parameters on the switch affect voice and/or video quality:

- Call Admission Control
- Expedited bandwidth requests
- Unscheduled automatic power save delivery

Call Admission Control (CAC) and UAPSD are supported on Cisco Compatible Extensions (CCX) v4 and v5; however, these parameters are also supported even without CCX but on any device implementing WMM (that supports 802.1e). Expedited bandwidth requests are supported only on CCXv5.

Traffic stream metrics (TSM) can be used to monitor and report issues with voice quality.

Call Admission Control

Call Admission Control (CAC) enables an access point to maintain controlled quality of service (QoS) when the wireless LAN is experiencing congestion. The WMM protocol deployed in CCXv4 maintains QoS under differing network loads.

Two types of Over The Air (OTA) CAC are available: static-based CAC and load-based CAC.

The switch supports the following QoS policies:

- User-defined policies: You can define your own QoS policies. You can have more control over these policies than the existing metal policies.
- System-defined precious metal policies: To support backward compatibility.
  - Platinum: Used for VoIP clients.
  - Gold: Used for video clients.
  - Silver: Used for best effort traffic.
  - Bronze: Used for NRT traffic.

Static-Based CAC

Voice over WLAN applications supporting WMM and TSPEC can specify how much bandwidth or shared medium time is required to initiate a call. Bandwidth-based, or static, CAC enables the access point to determine whether it is capable of accommodating a particular call. The access point rejects the call if necessary in order to maintain the maximum allowed number of calls with acceptable quality.

The QoS setting for a WLAN determines the level of bandwidth-based CAC support. To use bandwidth-based CAC with voice applications, the WLAN must be configured for Platinum QoS. With bandwidth-based CAC, the access point bandwidth availability is determined based on the amount of bandwidth currently used by the access point clients, to which the bandwidth requested by the Voice over WLAN applications is added. If this total exceeds a configured bandwidth threshold, the new call is rejected.

Note You must enable admission control (ACM) for CCXv4 clients that have WMM enabled. Otherwise, bandwidth-based CAC does not operate properly for these CCXv4 clients.

Load-Based CAC

Load-based CAC incorporates a measurement scheme that takes into account the bandwidth consumed by all traffic types (including that from clients), cochannel access point loads, and coallocated channel interference, for voice and video applications. Load-based CAC also covers the additional bandwidth consumption resulting from PHY and channel impairment.

In load-based CAC, the access point continuously measures and updates the utilization of the RF channel (that is, the mean time of bandwidth that has been exhausted), channel interference, and the additional calls that the access point can admit. The access point admits a new call only if the channel has enough unused bandwidth to support that call. By doing so, load-based CAC prevents oversubscription of the channel and maintains QoS under all conditions of WLAN loading and interference.

Note If you disable load-based CAC, the access points start using bandwidth-based CAC.

IOSd Call Admission Control

IOSd Call Admission Control (CAC) controls bandwidth availability from switch to access point.

You can configure class-based, unconditional packet marking features on your switch for CAC.

CAC is a concept that applies to voice and video traffic only—not data traffic. If an influx of data traffic oversubscribes a particular link in the network, queueing, buffering, and packet drop decisions resolve the congestion. The extra traffic is simply delayed until the interface becomes available to send the traffic, or, if traffic is dropped, the protocol or the end user initiates a timeout and requests a retransmission of the information.

Network congestion cannot be resolved in this manner when real-time traffic, sensitive to both latency and packet loss, is present, without jeopardizing the quality of service (QoS) expected by the users of that traffic. For real-time delay-sensitive traffic such as voice, it is better to deny network access under congestion conditions than to allow traffic onto the network to be dropped and delayed, causing intermittent impaired QoS and resulting in customer dissatisfaction.

CAC is therefore a deterministic and informed decision that is made before a voice call is established and is based on whether the required network resources are available to provide suitable QoS for the new call.

Based on the admit CAC CLI configuration in addition to the existing CAC algorithm, switch allows either voice or video with TSPEC or SIP snooping. The **admit cac** CLI is mandatory for the voice call to pass through.

If the BSSID policer is configured for the voice or video traffic, then additional checks are performed on the packets.

Expedited Bandwidth Requests

The expedited bandwidth request feature enables CCXv5 clients to indicate the urgency of a WMM traffic specifications (TSPEC) request (for example, an e911 call) to the WLAN. When the controller receives this request, it attempts to facilitate the urgency of the call in any way possible without potentially altering the quality of other TSPEC calls that are in progress.

You can apply expedited bandwidth requests to both bandwidth-based and load-based CAC. Expedited bandwidth requests are disabled by default. When this feature is disabled, the controller ignores all expedited requests and processes TSPEC requests as normal TSPEC requests.

The following table lists examples of TSPEC request handling for normal TSPEC requests and expedited bandwidth requests.

Table 18: TSPEC Request Handling Examples

CAC Mode	Reserved bandwidth for voice calls[1]	Usage[2]	Normal TSPEC Request	TSPEC with Expedited Bandwidth Request
Bandwidth-based CAC	75% (default setting)	Less than 75%	Admitted	Admitted
		Between 75% and 90% (reserved bandwidth for voice calls exhausted)	Rejected	Admitted
		More than 90%	Rejected	Rejected
Load-based CAC		Less than 75%	Admitted	Admitted
		Between 75% and 85% (reserved bandwidth for voice calls exhausted)	Rejected	Admitted
		More than 85%	Rejected	Rejected

1 For bandwidth-based CAC, the voice call bandwidth usage is per access point radio and does not take into account cochannel access points. For load-based CAC, the voice call bandwidth usage is measured for the entire channel.

2 Bandwidth-based CAC (consumed voice and video bandwidth) or load-based CAC (channel utilization [Pb]).

Note Admission control for TSPEC G711-20ms and G711-40 ms codec types are supported.

U-APSD

Unscheduled automatic power save delivery (U-APSD) is a QoS facility defined in IEEE 802.11e that extends the battery life of mobile clients. In addition to extending battery life, this feature reduces the latency of traffic flow delivered over the wireless media. Because U-APSD does not require the client to poll each individual packet buffered at the access point, it allows delivery of multiple downlink packets by sending a single uplink trigger packet. U-APSD is enabled automatically when WMM is enabled.

Traffic Stream Metrics

In a voice-over-wireless LAN (VoWLAN) deployment, traffic stream metrics (TSM) can be used to monitor voice-related metrics on the client-access point air interface. It reports both packet latency and packet loss. You can isolate poor voice quality issues by studying these reports.

The metrics consist of a collection of uplink (client side) and downlink (access point side) statistics between an access point and a client device that supports CCX v4 or later releases. If the client is not CCX v4 or CCXv5 compliant, only downlink statistics are captured. The client and access point measure these metrics. The access point also collects the measurements every 5 seconds, prepares 90-second reports, and then sends the reports to the controller. The controller organizes the uplink measurements on a client basis and the downlink measurements on an access point basis and maintains an hour's worth of historical data. To store this data, the controller requires 32 MB of additional memory for uplink metrics and 4.8 MB for downlink metrics.

TSM can be configured through either the GUI or the CLI on a per radio-band basis (for example, all 802.11a radios). The controller saves the configuration in flash memory so that it persists across reboots. After an access point receives the configuration from the controller, it enables TSM on the specified radio band.

This table shows the upper limit for TSM entries in different controller series.

TSM Entries	5700
MAX AP TSM entries	100
MAX Client TSM entries	250
MAX TSM entries	100*250=25000

Note Once the upper limit is reached, additional TSM entries cannot be stored and sent to WCS or NCS. If client TSM entries are full and AP TSM entries are available, then only the AP entries are stored, and viceversa. This leads to partial output. TSM cleanup occurs every one hour. Entries are removed only for those APs and clients that are not in the system.

Information About Configuring Voice Prioritization Using Preferred Call Numbers

You can configure a switch to provide support for SIP calls from VoWLAN clients that do not support TSPEC-based calls. This feature is known as SIP CAC support. If bandwidth is available in the configured voice pool, the SIP call uses the normal flow and the switch allocates the bandwidth to those calls.

You can also prioritize up to six preferred call numbers. When a call comes to one of the configured preferred numbers, the switch does not check the configured maximum voice bandwidth. The switch allocates the bandwidth needed for the call, even if it exceeds the maximum bandwidth for voice configured for voice CAC. The preferred call will be rejected if bandwidth allocation exceeds 85% of the radio bandwidth. The bandwidth allocation is 85 percent of the entire bandwidth pool, not just from the maximum configured voice pool. The bandwidth allocation is the same even for roaming calls.

You must configure the following parameters before configuring voice prioritization:

- Set WLAN QoS to allow voice calls to pass through.
- Enable ACM for the radio.
- Enable SIP call snooping on the WLAN.

Information About EDCA Parameters

Enhanced distributed channel access (EDCA) parameters are designed to provide preferential wireless channel access for voice, video, and other quality-of-service (QoS) traffic.

How to Configure Voice and Video Parameters

Configuring Voice Parameters (CLI)

Before You Begin

Ensure that you have configured SIP-based CAC.

You should have created a class map for CAC before beginning this procedure.

SUMMARY STEPS

1. **show wlan summary**
2. **show wlan** *wlan_id*
3. **configure terminal**
4. **policy-map** *policy-map name*
5. **class** {*class-name* | **class-default**}
6. **admit cac wmm-tspec**
7. **service-policy** *policy-map name*
8. **end**
9. **wlan** *wlan_profile_name wlan_ID SSID_network_name* **wlan shutdown**
10. **wlan** *wlan_profile_name wlan_ID SSID_network_name*
11. **wlan** *wlan_name* **call-snoop**
12. **wlan** *wlan_name* **service-policy input** *input_policy_name*
13. **wlan** *wlan_name* **service-policy output** *ouput_policy_name*
14. **wlan** *wlan_name* **service-policy input** *ingress_policy_name*
15. **wlan** *wlan_name* **service-policy output** *egress_policy_name*
16. **ap dot11 {5ghz | 24ghz} shutdown**
17. **ap dot11 {5ghz | 24ghz} cac voice sip**
18. **ap dot11 {5ghz | 24ghz} cac voice acm**
19. **ap dot11 {5ghz | 24ghz} cac voice max-bandwidth** *bandwidth*
20. **ap dot11 {5ghz | 24ghz} cac voice roam-bandwidth** *bandwidth*
21. **no wlan shutdown**
22. **no ap dot11 {5ghz | 24ghz} shutdown**
23. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show wlan summary** **Example:** `Switch# show wlan summary`	Specifies all of the WLANs configured on the switch.
Step 2	**show wlan** *wlan_id* **Example:** `Switch# show wlan 25`	Specifies the WLAN that you plan to modify. For voice over WLAN, ensure that the WLAN is configured for WMM and the QoS level is set to Platinum.
Step 3	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 4	**policy-map** *policy-map name*	Enters policy map configuration mode.

	Command or Action	Purpose

	Example:
	`Switch(config)# policy-map test_2000`
	`Switch(config-pmap)#`
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.
		In WLAN, you need to configure service-policy for these commands to take effect.

Step 5	**class** {*class-name* | **class-default**}
	Example:
	`Switch(config-pmap)# class test_1000`
	`Switch(config-pmap-c)#`
		Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change.
		Specifies the name of the class whose policy you want to create or change.
		You can also create a system default class for unclassified packets.

Step 6	**admit cac wmm-tspec**
	Example:
	`Switch(config-pmap-c)# admit cac wmm-tspec`
	`Switch(config-pmap-c)#`
		(Optional) Admits the request for Call Admission Control (CAC) for policy map.

Step 7	**service-policy** *policy-map name*
	Example:
	`Switch(config-pmap-c)# service-policy test_2000`
	`Switch(config-pmap-c)#`
		Configures the QoS service policy.

Step 8	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Step 9	**wlan** *wlan_profile_name wlan_ID SSID_network_name* **wlan shutdown**
	Example:
	`Switch(config)# wlan wlan1`
	`Switch(config-wlan)# wlan shutdown`
		Disables all WLANs with WMM enabled prior to changing the video parameters.

Step 10	**wlan** *wlan_profile_name wlan_ID SSID_network_name*
	Example:
	`Switch(config)# wlan wlan1`
	`Switch(config-wlan)# wlan shutdown`
		Disables all WLANs with WMM enabled prior to changing the voice parameters.

Step 11	**wlan** *wlan_name* **call-snoop** **Example:** `Switch(config)# wlan wlan1 call-snoop`	Enables the call-snooping on a particular WLAN.
Step 12	**wlan** *wlan_name* **service-policy input** *input_policy_name*	Configures input SSID policy on a particular WLAN to voice.

	Command or Action	Purpose

	Example:
	`Switch(config)# wlan wlan1`
	`Switch(config-wlan)# service-policy input platinum-up`	

Step 13	**wlan** *wlan_name* **service-policy output** *ouput_policy_name*
	
	Example:
	`Switch(config)# wlan wlan1`
	`Switch(config-wlan)# service-policy output platinum`
		Configures output SSID policy on a particular WLAN to voice.

Step 14	**wlan** *wlan_name* **service-policy input** *ingress_policy_name*
	
	Example:
	`Switch(config)# wlan wlan1`
	`Switch(config-wlan)# service-policy input policy1`
		Configures ingress SSID policy on a particular WLAN as user-defined policy.

Step 15	**wlan** *wlan_name* **service-policy output** *egress_policy_name*
	
	Example:
	`Switch(config)# wlan wlan1`
	`Switch(config-wlan)# service-policy output policy2`
		Configures egress SSID policy on a particular WLAN as user-defined policy.

Step 16	**ap dot11 {5ghz | 24ghz} shutdown** **Example:**	Disables the radio network. `Switch(config)# ap dot11 5ghz shutdown`
Step 17	**ap dot11 {5ghz | 24ghz} cac voice sip** **Example:** `Switch(config)# ap dot11 5ghz cac voice sip`	Enables or disables SIP IOSd CAC for the 802.11a or 802.11b/g network.
Step 18	**ap dot11 {5ghz | 24ghz} cac voice acm** **Example:** `Switch(config)# ap dot11 5ghz cac voice acm`	Enables or disables bandwidth-based voice CAC for the 802.11a or 802.11b/g network.
Step 19	**ap dot11 {5ghz | 24ghz} cac voice max-bandwidth** *bandwidth* **Example:** `Switch(config)# ap dot11 5ghz cac voice max-bandwidth 85`	Sets the percentage of maximum bandwidth allocated to clients for voice applications on the 802.11a or 802.11b/g network. The bandwidth range is 5 to 85%, and the default value is 75%. Once the client reaches the value specified, the access point rejects new videos on this network.
Step 20	**ap dot11 {5ghz | 24ghz} cac voice roam-bandwidth** *bandwidth*	Sets the percentage of maximum allocated bandwidth reserved for roaming voice clients.

	Command or Action	Purpose
	Example: `Switch(config)# ap dot11 5ghz cac voice roam-bandwidth 10`	The bandwidth range is 0 to 25%, and the default value is 6%. The switch reserves this much bandwidth from the maximum allocated bandwidth for roaming voice clients.
Step 21	**no wlan shutdown** **Example:** `Switch(config-wlan)# no wlan shutdown`	Reenables all WLANs with WMM enabled.
Step 22	**no ap dot11 {5ghz | 24ghz} shutdown** **Example:** `Switch(config)# no ap dot11 5ghz shutdown`	Reenables the radio network.
Step 23	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Video Parameters (CLI)

SUMMARY STEPS

1. **show wlan summary**
2. **show wlan** *wlan_id*
3. **configure terminal**
4. **policy-map** *policy-map name*
5. **class** {*class-name* | **class-default**}
6. **admit cac wmm-tspec**
7. **service-policy** *policy-map name*
8. **end**
9. **wlan***wlan_profile_name*
10. **ap dot11 {5ghz | 24ghz} shutdown**
11. **ap dot11 {5ghz | 24ghz} cac video acm**
12. **ap dot11 {5ghz | 24ghz} cac video load-based**
13. **ap dot11 {5ghz | 24ghz} cac video max-bandwidth** *bandwidth*
14. **ap dot11 {5ghz | 24ghz} cac video roam-bandwidth** *bandwidth*
15. **no wlan shutdown** *wlan_id*
16. **no ap dot11 {5ghz | 24ghz} shutdown**
17. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show wlan summary** **Example:** `Switch# show wlan summary`	Specifies all of the WLANs configured on the switch.
Step 2	**show wlan** *wlan_id* **Example:** `Switch# show wlan 25`	Specifies the WLAN that you plan to modify.
Step 3	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

Step 4	**policy-map** *policy-map name*
	
	Example:
	`Switch(config)# policy-map test_2000`
	`Switch(config-pmap)#`
		Enters policy map configuration mode.
		
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.
		
		In WLAN, you need to configure service-policy for these commands to take effect.

Step 5	**class** {*class-name* | **class-default**}
	
	Example:
	`Switch(config-pmap)# class test_1000`
	`Switch(config-pmap-c)#`
		Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change.
		
		Specifies the name of the class whose policy you want to create or change.
		
		You can also create a system default class for unclassified packets.

Step 6	**admit cac wmm-tspec**
	
	Example:
	`Switch(config-pmap-c)# admit cac wmm-tspec`
	`Switch(config-pmap-c)#`
		(Optional) Admits the request for Call Admission Control (CAC) for policy map.

Step 7	**service-policy** *policy-map name*
	
	Example:
	`Switch(config-pmap-c)# service-policy test_2000`
	`Switch(config-pmap-c)#`
		Configures the QoS service policy.

Step 8	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

	Command or Action	Purpose

Step 9	**wlan***wlan_profile_name*
	
	Example:
	`Switch(config)# wlan wlan1`
	`Switch(config-wlan)# wlan shutdown`
		Disables all WLANs with WMM enabled prior to changing the video parameters.

Step 10	**ap dot11 {5ghz | 24ghz} shutdown** **Example:** `Switch(config)# ap dot11 5ghz shutdown`	Disables the radio network.
Step 11	**ap dot11 {5ghz | 24ghz} cac video acm** **Example:** `Switch(config)# ap dot11 5ghz cac video acm`	Enables or disables bandwidth-based video CAC for the 802.11a or 802.11b/g network.
Step 12	**ap dot11 {5ghz | 24ghz} cac video load-based** **Example:** `Switch(config)# ap dot11 5ghz cac video load-based`	Configures the load-based CAC method. If you do not enter this command, then the default static CAC is applied.
Step 13	**ap dot11 {5ghz | 24ghz} cac video max-bandwidth** *bandwidth* **Example:** `Switch(config)# ap dot11 5ghz cac video max-bandwidth 20`	Sets the percentage of maximum bandwidth allocated to clients for video applications on the 802.11a or 802.11b/g network. The bandwidth range is 5 to 85%, and the default value is 75%. The default value is 0, which means no bandwidth request control. The sum of the voice bandwidth and video bandwidth should not exceed 85% or configured maximum media bandwidth.
Step 14	**ap dot11 {5ghz | 24ghz} cac video roam-bandwidth** *bandwidth* **Example:** `Switch(config)# ap dot11 5ghz cac video roam-bandwidth 9`	Sets the percentage of maximum allocated bandwidth reserved for roaming clients for video. The bandwidth range is 0 to 25%, and the default value is 0%.
Step 15	**no wlan shutdown** *wlan_id* **Example:** `Switch(config-wlan)# no wlan shutdown 25`	Reenables all WLANs with WMM enabled.
Step 16	**no ap dot11 {5ghz | 24ghz} shutdown** **Example:** `Switch(config)# no ap dot11 5ghz shutdown`	Reenables the radio network.
Step 17	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring SIP-Based CAC (CLI)

SIP CAC controls the total number of SIP calls that can be made.

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name*
3. **call-snoop**
4. **service-policy** [**client**] **input** *policy-map name*
5. **service-policy** [**client**] **output** *policy-map name*
6. **end**
7. **show wlan** {*wlan-id* | *wlan-name*}
8. **configure terminal**
9. **ap dot11 {5ghz | 24ghz} cac {voice | video} acm**
10. **ap dot11 {5ghz | 24ghz} cac voice sip**
11. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

Step 2	**wlan** *wlan-name*
	
	Example:
	`Switch(config)# wlan qos-wlan`
	`Switch(config-wlan)#`
		Enters WLAN configuration submode.

Step 3	**call-snoop** **Example:** `Switch(config-wlan)# call-snoop`	Enables the call-snooping feature for a particular WLAN.
Step 4	**service-policy** [**client**] **input** *policy-map name* **Example:** `Switch(config-wlan)# service-policy input platinum-up`	Assigns a policy map to WLAN input traffic. Ensure that you provide QoS policy to voice for input traffic.
Step 5	**service-policy** [**client**] **output** *policy-map name* **Example:** `Switch(config-wlan)# service-policy output platinum`	Assigns policy map to WLAN output traffic. Ensure that you provide QoS policy to voice for output traffic.

	Command or Action	Purpose
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.
Step 7	**show wlan** {*wlan-id* | *wlan-name*} **Example:** `Switch# show wlan qos-wlan`	Verifies the configured QoS policy on the WLAN.
Step 8	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 9	**ap dot11 {5ghz | 24ghz} cac {voice | video} acm** **Example:** `Switch(config)# ap dot11 5ghz cac voice acm`	Enables the ACM static on the radio. When enabling SIP snooping, use the static CAC, not the load-based CAC.
Step 10	**ap dot11 {5ghz | 24ghz} cac voice sip** **Example:** `Switch(config)# ap dot11 5ghz cac voice sip`	Configures SIP-based CAC.
Step 11	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring a Preferred Call Number (CLI)

Before You Begin

You must set the following parameters before configuring a preferred call number.

- Set WLAN QoS to voice.
- Enable ACM for the radio.
- Enable SIP call snooping on the WLAN.
- Enable SIP-based CAC.

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name* **qos platinum**
3. **ap dot11 {5ghz | 24ghz} cac {voice | video} acm**
4. **wlan** *wlan-name*
5. **wireless sip preferred-call-no** *call_index call_number*
6. **no wireless sip preferred-call-no** *call_index*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

Step 2	**wlan** *wlan-name* **qos platinum**
	
	Example:
	`Switch(config)# wlan wlan1`
	`Switch(config-wlan)# qos platinum`
		Sets QoS to voice on a particular WLAN.

Step 3	**ap dot11 {5ghz | 24ghz} cac {voice | video} acm** **Example:** `Switch(config)# ap dot11 5ghz cac voice acm`	Enables the static ACM on the radio. When enabling SIP snooping, use the static CAC, not the load-based CAC.

Step 4	**wlan** *wlan-name*
	
	Example:
	`Switch(config)# wlan wlan1`
	`Switch(config-wlan)# call-snoop`
		Enables the call-snooping feature for a particular WLAN.

Step 5	**wireless sip preferred-call-no** *call_index call_number* **Example:** `Switch(config)# wireless sip preferred-call-no 1 555333`	Adds a new preferred call.
Step 6	**no wireless sip preferred-call-no** *call_index* **Example:** `Switch(config)# no wireless sip preferred-call-no 1`	Removes a preferred call.
Step 7	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring EDCA Parameters (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 {5ghz | 24ghz } shutdown**
3. **ap dot11 {5ghz | 24ghz} edca-parameters {custom-voice | optimized-video-voice | optimized-voice | svp-voice | wmm-default}**
4. **show ap dot11 {5ghz | 24ghz} network**
5. **no ap dot11 {5ghz | 24ghz} shutdown**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 {5ghz | 24ghz } shutdown** **Example:** `Switch(config)# ap dot11 5ghz shutdown`	Disables the radio network.
Step 3	**ap dot11 {5ghz | 24ghz} edca-parameters {custom-voice | optimized-video-voice | optimized-voice | svp-voice | wmm-default}** **Example:** `Switch(config)# ap dot11 5ghz edca-parameters optimized-voice`	Enables a specific EDCA parameters for the 802.11a or 802.11b/g network. • custom-voice—Enables custom voice parameters for the 802.11a or 802.11b/g network. • optimized-video-voice—Enables EDCA voice- and video-optimized parameters for the 802.11a or 802.11b/g network. Choose this option when both voice and video services are deployed on your network. • optimized-voice—Enables non-SpectraLink voice-optimized profile parameters for the 802.11a or 802.11b/g network. Choose this option when voice services other than SpectraLink are deployed on your network. • svp-voice—Enables SpectraLink voice priority parameters for the 802.11a or 802.11b/g network. Choose this option if SpectraLink phones are deployed on your network to improve the quality of calls.

	Command or Action	Purpose
		• wmm-default—Enables the Wi-Fi Multimedia (WMM) default parameters for the 802.11a or 802.11b/g network. This is the default value. Choose this option when voice or video services are not deployed on your network.
Step 4	**show ap dot11 {5ghz | 24ghz} network** **Example:** `Switch(config)# show ap dot11 5ghz network`	Displays the current status of MAC optimization for voice.
Step 5	**no ap dot11 {5ghz | 24ghz} shutdown** **Example:** `Switch(config)# no ap dot11 5ghz shutdown`	Reenables the radio network.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Monitoring Voice and Video Parameters

This section describes the new commands for the voice and video parameters.

The following commands can be used to monitor voice and video parameters.

Table 19: Monitoring Voice Parameters Commands

Command	Purpose
show ap dot11 {5ghz | 24ghz} network	Displays the radio-based statistics for voice.
show ap name *ap_name* **dot11 24ghz tsm all**	Displays the TSM voice metrics and current status of MAC optimization for voice.
show ap name *apname* **cac voice**	Displays the information about CAC for a particular access point.
show client detail *client_mac*	Displays the U-APSD status for a particular client.
show policy-map interface wireless client	Displays the video client policy details.
show access-list	Displays the video client dynamic access-list from the switch.

show wireless client voice diag status	Displays information about whether voice diagnostics are enabled or disabled. If enabled, this also displays information about the clients in the watch list and the time remaining for the diagnostics of the voice call. **Note** To work on voice diagnostics CLIs, you need to enter the following command: **debug voice-diagnostic mac-addr** *client_mac_01 client_mac_02*
show wireless client voice diag tspec	Displays the TSPEC information sent from the clients that are enabled for voice diagnostics.
show wireless client voice diag qos-map	Displays information about the QoS/DSCP mapping and packet statistics in each of the four queues: VO, VI, BE, BK. The different DSCP values are also displayed.
show wireless client voice diag rssi	Display the client's RSSI values in the last 5 seconds when voice diagnostics is enabled.
show client voice-diag roam-history	Displays information about the last three roaming calls. The output contains the timestamp, access point associated with roaming, roaming reason, and if there is a roaming failure, reason for roaming-failure.
show policy-map interface wireless mac *mac-address*	Displays information about the voice and video data packet statistics.
show wireless media-stream client summary	Displays a summary of the media stream and video client information.
show controllers d0 | b queue	Displays which queue the packets are going through on an access point.
show platform qos queue stats *interface*	Displays which queue packets are going through from the switch.

You can monitor the video parameters using the following commands.

Table 20: Monitoring Video Parameters Commands

Command	Purpose
show ap join stats summary *ap_mac*	Displays the last join error detail for a specific access point.
show ip igmp snooping wireless mgid	Displays the TSM voice metrics and current status of MAC optimization for voice.

show wireless media-stream multicast-direct state	Displays the media stream multicast-direct parameters.
show wireless media-stream group summary	Displays the summary of the media stream and client information.
show wireless media-stream group detail *group_name*	Displays the details of a specific media-stream group.
show wireless media-stream client summary	Displays the details for a set of media-stream clients.
show wireless media-stream client detail *group_name*	Displays the details for a set of media-stream clients.
show ap dot11 {5ghz | 24ghz) media-stream rrc	Display the details of media stream.
show wireless media-stream message details	Displays information about the message configuration.
show ap name *ap-name* **auto-rf dot11 5ghz | i Util**	Displays the details of channel utilization.
show controllers d0 | b queue	Displays which queue the packets are going through on an access point based on 2.4- and 5-GHz bands.
show controllers d1 | b queue	Displays which queue the packets are going through on an access point based on 2.4- and 5-GHz bands.
show cont d1 | b Media	Displays the video metric details on the band A or B.
show capwap mcast mgid all	Displays information about all of the multicast groups and their corresponding multicast group identifications (MGIDs) associated to the access point.
show capwap mcast mgid id *id*	Displays information about all of the video clients joined to the multicast group in a specific MGID.

Additional References for Voice and Video Parameters

Related Documents

Related Topic	Document Title
Multicast configuration	*Multicast Configuration Guide, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*
VideoStream configuration	*VideoStream Configuration Guide, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information For Performing Voice and Video Parameters Configuration

Release	Feature Information
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 10

Configuring RFID Tag Tracking

Finding Feature Information

Your software release may not support all of the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About Configuring RFID Tag Tracking

The Switch enables you to configure radio-frequency identification (RFID) tag tracking. RFID tags are small wireless devices that are affixed to assets for real-time location tracking. They operate by advertising their location using special 802.11 packets, which are processed by access points, the controller, and the location appliance.

How to Configure RFID Tag Tracking

Configuring RFID Tag Tracking (CLI)

SUMMARY STEPS

1. **location rfid status**
2. (Optional) **no location rfid status**
3. **location rfid timeout** *seconds*
4. **location rfid mobility vendor-name** *name*
5. (Optional) **no location rfid mobility** *name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**location rfid status** **Example:** `Switch(config)# location rfid status`	Enables RFID tag tracking. By default, RFID tag tracking is enabled.
Step 2	(Optional) **no location rfid status** **Example:** `Switch(config)# no location rfid status`	Disables RFID tag tracking.
Step 3	**location rfid timeout** *seconds* **Example:** `Switch(config)# location rfid timeout 1500`	Specifies a static timeout value (between 60 and 7200 seconds). The static timeout value is the amount of time that the switch maintains tags before expiring them. For example, if a tag is configured to beacon every 30 seconds, we recommend that you set the timeout value to 90 seconds (approximately three times the beacon value). The default value is 1200 seconds.
Step 4	**location rfid mobility vendor-name** *name* **Example:** `Switch(config)# location rfid mobility vendor-name Aerosct`	Enables RFID tag mobility for specific tags. When you enter the **location rfid mobility vendor-name** command, tags are unable to obtain a DHCP address for client mode when attempting to select and/or download a configuration. **Note** These commands can be used only for Pango tags. Therefore, the only valid entry for **vendor_name** is "pango" in all lowercase letters.
Step 5	(Optional) **no location rfid mobility** *name* **Example:** `Switch(config)# no location rfid mobility test`	Disables RFID tag mobility for specific tags. When you enter the **no location rfid mobility** command , tags can obtain a DHCP address. If a tag roams from one subnet to another, it obtains a new address rather than retaining the anchor state.

Monitoring RFID Tag Tracking Information

This section describes the new commands for the RFID tag tracking Information.

The following commands can be used to monitor the RFID tag tracking Information on the switch.

Table 21: Monitoring RFID Tag Tracking Information Commands

Command	Purpose
show location rfid config	Displays the current configuration for RFID tag tracking.
show location rfid detail *mac_address*	Displays the detailed information for a specific RFID tag.
show location rfid summary	Displays a list of all RFID tags currently connected to the switch.
show location rfid client	Displays a list of RFID tags that are associated to the switch as clients.

Additional References RFID Tag Tracking

Related Documents

Related Topic	Document Title
System management commands	*System Management Command Reference, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information For Performing RFID Tag Tracking Configuration

Release	Feature Information
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 11

Configuring Location Settings

Finding Feature Information

Your software release may not support all of the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About Configuring Location Settings

The switch determines the location of client devices by gathering Received Signal Strength Indication (RSSI) measurements from access points all around the client of interest. The switch can obtain location reports from up to 16 access points for clients, RFID tags, and rogue access points.

You can configure the path loss measurement (S60) request for normal clients or calibrating clients to improve location accuracy.

How to Configure Location Settings

Configuring Location Settings (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **location plm** {**calibrating** [**multiband** | **uniband**] | **client** *burst_interval*
3. **location rssi-half-life** {**calibrating-client** | **client** | **rogue-aps** | **tags** } *seconds*
4. **location expiry** {**calibrating-client** | **client** | **rogue-aps** | **tags** } *timeout*
5. **location algorithm** {**rssi-average** | **simple**}
6. **location admin-tag** *string*
7. **location civic-location identifier** {*identifier* | **host**}
8. **location custom-location identifier** {*identifier* | **host**}
9. **location geo-location identifier** {*identifier* | **host**}
10. **location prefer** {**cdp** | **lldp-med** | **static**} **weight** *priority_value*
11. **location rfid** {**status** | **timeout** | **vendor-name**}
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**location plm** {**calibrating** [**multiband** | **uniband**] | **client** *burst_interval* **Example:** `Switch(config)# location plm client 100`	Configures the path loss measurement (S60) request for calibrating clients or non-calibrating. The path loss measurement request improves the location accuracy. You can configure the **burst_interval** parameter for the normal, noncalibrating client from zero through 3600 seconds, and the default value is 60 seconds. You can configure the path loss measurement request for calibrating clients on the associated 802.11a or 802.11b/g radio or on the associated 802.11a/b/g radio. If a client does not send probes often or sends them only on a few channels, its location cannot be updated or cannot be updated accurately. The **location plm** command forces clients to send more packets on all channels. When a CCXv4 (or higher) client associates, the Switch sends it a path loss measurement request, which instructs the client to transmit on the bands and channels that the access points are on (typically, channels 1, 6, and 11 for 2.4-GHz-only access points) at a configurable interval (such as 60 seconds) indefinitely.

	Command or Action	Purpose
Step 3	**location rssi-half-life** {**calibrating-client** | **client** | **rogue-aps** | **tags** } *seconds* **Example:** `Switch(config)# location rssi-half-life calibrating-client 60`	Configures the RSSI half life for the clients, calibrating clients, RFID tags, and rogue access points. You can enter the **location rssi-half-life** parameter value for the clients, calibrating clients, RFID tags, and rogue access points as 0, 1, 2, 5, 10, 20, 30, 60, 90, 120, 180, or 300 seconds, and the default value is 0 seconds. Some client devices transmit at reduced power immediately after changing channels, and RF is variable, so RSSI values might vary considerably from packet to packet. The **location rssi-half-life** command increases accuracy by averaging nonuniformly arriving data using a configurable forget period (or half life). **Note** We recommend that you do not use or modify the **location rssi-half-life** command.
Step 4	**location expiry** {**calibrating-client** | **client** | **rogue-aps** | **tags** } *timeout* **Example:** `Switch(config)# location expiry calibrating-client 50`	Configures the RSSI timeout value for the clients, calibrating clients, RFID tags, and rogue access points. You can enter the RSSI timeout value for the clients, RFID tags, and rogue access points from 5 through 3600 seconds, and the default value is 5 seconds. For the calibrating clients, you can enter the RSSI timeout value from 0 through 3600 seconds, and the default value is 5 seconds. Ensuring that recent, strong RSSIs are retained by the CPU is critical to location accuracy. The **location expiry** command enables you to specify the length of time after which old RSSI averages expire. **Note** We recommend that you do not use or modify the **location expiry** command.
Step 5	**location algorithm** {**rssi-average** | **simple**} **Example:** `Switch(config)# location algorithm rssi-average`	Configures the algorithm used to average RSSI and signal-to-noise ratio (SNR) values. You can enter the **location algorithm rssi-average** command to specify a more accurate algorithm but requires more CPU overhead or the **location algorithm simple** command to specify a faster algorithm that requires low CPU overhead but provides less accuracy. **Note** We recommend that you do not use or modify the **location algorithm** command.
Step 6	**location admin-tag** *string* **Example:** `Switch(config)# location admin-tag`	Sets administrative tag or site information for the location of client devices.
Step 7	**location civic-location identifier** {*identifier* | **host**} **Example:** `Switch(config)# location civic-location identifier host`	Specifies civic location information. You can set the civic location identifier either as a string or host.

	Command or Action	Purpose
Step 8	**location custom-location identifier** {*identifier* | **host**} **Example:** `Switch(config)# location custom-location identifier host`	Specifies custom location information. You can set the custom location identifier either as a string or host.
Step 9	**location geo-location identifier** {*identifier* | **host**} **Example:** `Switch(config)# location geo-location identifier host`	Specifies geographical location information of the client devices. You can set the location identifier either as a string or host.
Step 10	**location prefer** {**cdp** | **lldp-med** | **static**} **weight** *priority_value* **Example:** `Switch(config)# location prefer weight cdp 50`	Sets location information source priority. You can enter the priority weight from zero through 255.
Step 11	**location rfid** {**status** | **timeout** | **vendor-name**} **Example:** `Switch(config)# location rfid timeout 100`	Configures RFID tag tracking options such as RFID tag status, RFID timeout value, and RFID tag vendor name. You can enter the RFID timeout value in a range from 60 and 7200 seconds.
Step 12	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Modifying the NMSP Notification Interval for Clients, RFID Tags, and Rogues (CLI)

The Network Mobility Services Protocol (NMSP) manages communication between the mobility services engine and the controller for incoming and outgoing traffic. If your application requires more frequent location updates, you can modify the NMSP notification interval (to a value between 1 and 180 seconds) for clients, active RFID tags, and rogue access points and clients.

Note The TCP port (16113) that the controller and mobility services engine communicate over must be open (not blocked) on any firewall that exists between the controller and the mobility services engine for NMSP to function.

SUMMARY STEPS

1. **configure terminal**
2. **nmsp notification interval** {**attachment** *seconds* | **location** *seconds* | **rssi** [**clients** *interval* | **rfid** *interval* | **rogues** [**ap** | **client**] *interval*]}
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**nmsp notification interval** {**attachment** *seconds* | **location** *seconds* | **rssi** [**clients** *interval* | **rfid** *interval* | **rogues** [**ap** | **client**] *interval*]} **Example:** `Switch(config)# nmsp notification interval rssi rfid 50`	Sets the NMSP notification interval value for clients, RFID tags, and rogue clients and access points. You can enter the NMSP notification interval value for RSSI measurement from 1 through 180 seconds.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Modifying the NMSP Notification threshold for Clients, RFID Tags, and Rogues (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **location notify-threshold** {**clients** | **rogues ap** | **tags** } *threshold*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**location notify-threshold** {**clients** | **rogues ap** | **tags** } *threshold* **Example:** `Switch(config)# location notify-threshold clients 5`	Configures the NMSP notification threshold for clients, RFID tags, and rogue clients and access points. You can enter the RSSI threshold value from zero through 10 db.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Monitoring Location Settings and NMSP Settings

Monitoring Location Settings (CLI)

This section describes the new commands for location settings.

The following commands can be used to monitor location settings on the switch.

Table 22: Monitoring Location Settings Commands

Command	Purpose
show location summary	Displays the current location configuration values.
show location statistics rfid	Displays the location-based RFID statistics.
show location detail *client_mac_addr*	Displays the RSSI table for a particular client.

Monitoring NMSP Settings (CLI)

This section describes the new commands for NMSP settings.

The following commands can be used to monitor NMSP settings on the switch.

Table 23: Monitoring NMSP Settings Commands

Command	Purpose
show nmsp attachment suppress interfaces	Displays the attachment suppress interfaces.

show nmsp capability	Displays the NMSP capabilities.
show nmsp notification interval	Displays the NMSP notification intervals.
show nmsp statistics connection	Displays the connection-specific NMSP counters.
show nmsp statistics summary	Displays the common NMSP counters.
show nmsp status	Displays the status of active NMSP connections.
show nmsp subscription detail	Displays all of the mobility services to which the switch is subscribed.
show nmsp subscription detail *ip_addr*	Displays details only for the mobility services subscribed to by a specific IP address.
show nmsp subscription summary	Displays details for all of the mobility services to which the switch is subscribed.

Examples: Location Settings Configuration

This example shows how to configure the path loss measurement (S60) request for calibrating client on the associated 802.11a or 802.11b/g radio:

```
Switch# configure terminal
Switch(config)# location plm calibrating uniband
Switch(config)# end
Switch# show location summary
```

This example shows how to configure the RSSI half life for a rouge access point:

```
Switch# configure terminal
Switch(config)# location rssi-half-life rogue-aps 20
Switch(config)# end
Switch# show location summary
```

Examples: NMSP Settings Configuration

This example shows how to configure the NMSP notification interval for RFID tags:

```
Switch# configure terminal
Switch(config)# nmsp notification interval rssi rfid 50
Switch(config)# end
Switch# show nmsp notification interval
```

This example shows how to configure the NMSP notification threshold for clients:

```
Switch# configure terminal
Switch(config)# nmsp notify-threshold 5
Switch(config)# end
Switch# show nmsp statistics summary
```

Additional References for Location Settings

Related Documents

Related Topic	Document Title
System management commands	*System Management Command Reference, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information For Performing Location Settings Configuration

Release	Feature Information
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 12

Configuring SDM Templates

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information About Configuring SDM Templates

SDM Templates

You can use SDM templates to configure system resources to optimize support for specific features, depending on how your device is used in the network. You can select a template to provide maximum system usage for some functions.

These templates are supported on your device:

- Advanced—The advanced template is available on all supported images for this release. It maximizes system resources for features like netflow, multicast groups, security ACEs, QoS ACEs, and so on.
- VLAN—The VLAN template is available only on the LAN Base license. The VLAN template disables routing and supports the maximum number of unicast MAC addresses. It would typically be selected for a Layer 2 device.

After you change the template and the system reboots, you can use the **show sdm prefer** privileged EXEC command to verify the new template configuration. If you enter the **show sdm prefer** command before you enter the **reload** privileged EXEC command, the **show sdm prefer** command shows the template currently in use and the template that will become active after a reload.

The default is the advanced template.

Table 24: Approximate Number of Feature Resources Allowed by Templates

Resource	Advanced	VLAN
Number of VLANs	4094	4094
Unicast MAC addresses	32 K	32 K
Overflow unicast MAC addresses	512	512
IGMP groups and multicast routes	4 K	4 K
Overflow IGMP groups and multicast routes	512	512
• Directly connected routes	32 K	32 K
• Indirectly connected IP hosts	8 K	8 K
Policy-based routing ACEs	1024	0
QoS classification ACEs	3 K	3 K
Security ACEs	3 K	3 K
Netflow ACEs	1024	1024
Input Microflow policer ACEs:	256 K	0
Output Microflow policer ACEs:	256 K	0
FSPAN ACEs	256	256
Control Plane Entries:	512	512
Input Netflow flows:	8 K	8 K

Resource	Advanced	VLAN
Output Netflow flows:	16 K	16 K

Note When the switch is used as a Wireless Mobility Agent, the only template allowed is the advanced template.

The tables represent approximate hardware boundaries set when a template is selected. If a section of a hardware resource is full, all processing overflow is sent to the CPU, seriously impacting switch performance.

SDM Templates and Switch Stacks

In a switch stack, all stack members must use the same SDM template that is stored on the active switch. When a new switch is added to a stack, the SDM configuration that is stored on the active switch overrides the template configured on an individual switch.

How to Configure SDM Templates

Configuring SDM Templates

Configuring the Switch SDM Template

Setting the SDM Template

SUMMARY STEPS

1. **configure terminal**
2. **sdm prefer** { **advanced** | **vlan** }
3. **end**
4. **reload**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch> configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**sdm prefer** { **advanced** | **vlan** } **Example:** `Switch(config)# sdm prefer advanced`	Specifies the SDM template to be used on the switch. The keywords have these meanings: • **advanced** —Supports advanced features such as Netflow. • **vlan** —Maximizes VLAN configuration on the switch with no routing supported in hardware. **Note** The **no sdm prefer** command and a default template is not supported.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**reload** **Example:** `Switch# reload`	Reloads the operating system.

Monitoring and Maintaining SDM Templates

Command	Purpose
show sdm prefer	Displays the SDM template in use.
reload	Reloads the switch to activate the newly configured SDM template.
no sdm prefer	Sets the default SDM template.

Configuration Examples for Configuring SDM Templates

Examples: Configuring SDM Templates

This example shows how to configure the VLAN template:

```
Switch(config)# sdm prefer vlan
Switch(config)# exit
Switch# reload
```

```
Proceed with reload? [confirm]
```

Examples: Displaying SDM Templates

This is an example output showing the advanced template information:

```
Switch# show sdm prefer

Showing SDM Template Info

This is the Advanced template.
  Number of VLANs:                                 4094
  Unicast MAC addresses:                           32768
  Overflow Unicast MAC addresses:                  512
  IGMP and Multicast groups:                       8192
  Overflow IGMP and Multicast groups:              512
  Directly connected routes:                       32768
  Indirect routes:                                 8192
  Security Access Control Entries:                 3072
  QoS Access Control Entries:                      2816
  Policy Based Routing ACEs:                       1024
  Netflow ACEs:                                    1024
  Input Microflow policer ACEs:                    256
  Output Microflow policer ACEs:                   256
  Flow SPAN ACEs:                                  256
  Tunnels:                                         256
  Control Plane Entries:                           512
  Input Netflow flows:                             8192
  Output Netflow flows:                            16384
These numbers are typical for L2 and IPv4 features.
Some features such as IPv6, use up double the entry size;
so only half as many entries can be created.


Switch#
```

This is an example output showing the VLAN template information:

```
Switch# show sdm prefer vlan

Showing SDM Template Info

This is the VLAN template for a typical Layer 2 network.
  Number of VLANs:                                 4094
  Unicast MAC addresses:                           32768
  Overflow Unicast MAC addresses:                  512
  IGMP and Multicast groups:                       8192
  Overflow IGMP and Multicast groups:              512
  Directly connected routes:                       32768
  Indirect routes:                                 8192
  Security Access Control Entries:                 3072
  QoS Access Control Entries:                      3072
  Policy Based Routing ACEs:                       0
  Netflow ACEs:                                    1024
  Input Microflow policer ACEs:                    0
  Output Microflow policer ACEs:                   0
  Flow SPAN ACEs:                                  256
  Tunnels:                                         0
  Control Plane Entries:                           512
  Input Netflow flows:                             16384
  Output Netflow flows:                            8192
These numbers are typical for L2 and IPv4 features.
Some features such as IPv6, use up double the entry size;
so only half as many entries can be created.

Switch#
```

Additional References for SDM Templates

Related Documents

Related Topic	Document Title
SDM command reference	*System Management Command Reference (Catalyst 3850 Switches)*
VLAN configuration guide	*VLAN Configuration Guide (Catalyst 3850 Switches)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Configuring SDM Templates

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 13

Configuring System Message Logs

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information About Configuring System Message Logs

System Messsage Logging

By default, a switch sends the output from system messages and **debug** privileged EXEC commands to a logging process. Stack members can trigger system messages. A stack member that generates a system message appends its hostname in the form of hostname-n, where n is a switch range from 1 to 4, and redirects the output to the logging process on the . Though the is a stack member, it does not append its hostname to system messages. The logging process controls the distribution of logging messages to various destinations, such as the logging buffer, terminal lines, or a UNIX syslog server, depending on your configuration. The process also sends messages to the console.

When the logging process is disabled, messages are sent only to the console. The messages are sent as they are generated, so message and debug output are interspersed with prompts or output from other commands. Messages appear on the active consoles after the process that generated them has finished.

You can set the severity level of the messages to control the type of messages displayed on the consoles and each of the destinations. You can time-stamp log messages or set the syslog source address to enhance real-time debugging and management. For information on possible messages, see the system message guide for this release.

You can access logged system messages by using the switch command-line interface (CLI) or by saving them to a properly configured syslog server. The switch software saves syslog messages in an internal buffer on a standalone switch, and in the case of a switch stack, on the . If a standalone switch or the stack master fails, the log is lost unless you had saved it to flash memory.

You can remotely monitor system messages by viewing the logs on a syslog server or by accessing the switch through Telnet, through the console port, or through the Ethernet management port. In a switch stack, all stack member consoles provide the same console output.

Note The syslog format is compatible with 4.3 BSD UNIX.

System Log Message Format

System log messages can contain up to 80 characters and a percent sign (%), which follows the optional sequence number or time-stamp information, if configured. Depending on the switch, messages appear in one of these formats:

- *seq no:timestamp: %facility-severity-MNEMONIC:description (hostname-n)*
- *seq no:timestamp: %facility-severity-MNEMONIC:description*

The part of the message preceding the percent sign depends on the setting of these global configuration commands:

- **service sequence-numbers**
- **service timestamps log datetime**
- **service timestamps log datetime** [**localtime**] [**msec**] [**show-timezone**]
- **service timestamps log uptime**

Table 25: System Log Message Elements

Element	Description
seq no:	Stamps log messages with a sequence number only if the **service sequence-numbers** global configuration command is configured.

Element	Description
timestamp formats: *mm/dd h h:mm:ss* or *hh:mm:ss* (short uptime) or *d h* (long uptime)	Date and time of the message or event. This information appears only if the **service timestamps log** [**datetime** | **log**] global configuration command is configured.
facility	The facility to which the message refers (for example, SNMP, SYS, and so forth).
severity	Single-digit code from 0 to 7 that is the severity of the message.
MNEMONIC	Text string that uniquely describes the message.
description	Text string containing detailed information about the event being reported.
hostname-n	Hostname of a stack member and its switch number in the stack. Though the is a stack member, it does *not* append its hostname to system messages.

Default System Message Logging Settings

Table 26: Default System Message Logging Settings

Feature	Default Setting
System message logging to the console	Enabled.
Console severity	Debugging.
Logging file configuration	No filename specified.
Logging buffer size	4096 bytes.
Logging history size	1 message.
Time stamps	Disabled.
Synchronous logging	Disabled.
Logging server	Disabled.

Feature	Default Setting
Syslog server IP address	None configured.
Server facility	Local7
Server severity	Informational.

Syslog Message Limits

If you enabled syslog message traps to be sent to an SNMP network management station by using the **snmp-server enable trap** global configuration command, you can change the level of messages sent and stored in the switch history table. You also can change the number of messages that are stored in the history table.

Messages are stored in the history table because SNMP traps are not guaranteed to reach their destination. By default, one message of the level **warning** and numerically lower levels are stored in the history table even if syslog traps are not enabled.

When the history table is full (it contains the maximum number of message entries specified with the **logging history size** global configuration command), the oldest message entry is deleted from the table to allow the new message entry to be stored.

The history table lists the level keywords and severity level. For SNMP usage, the severity level values increase by 1. For example, *emergencies* equal 1, not 0, and *critical* equals 3, not 2.

How to Configure System Message Logs

Setting the Message Display Destination Device

If message logging is enabled, you can send messages to specific locations in addition to the console.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **logging buffered** [*size*]
3. **logging** *host*
4. **logging file flash:** *filename* [*max-file-size* [*min-file-size*]] [*severity-level-number* | *type*]
5. **end**
6. **terminal monitor**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**logging buffered** [*size*] **Example:** `Switch(config)# logging buffered 8192`	Logs messages to an internal buffer on the switch or on a standalone switch or, in the case of a switch stack, on the . The range is 4096 to 2147483647 bytes. The default buffer size is 4096 bytes. If a standalone switch or the fails, the log file is lost unless you previously saved it to flash memory. See Step 4. **Note** Do not make the buffer size too large because the switch could run out of memory for other tasks. Use the **show memory** privileged EXEC command to view the free processor memory on the switch. However, this value is the maximum available, and the buffer size should *not* be set to this amount.
Step 3	**logging** *host* **Example:** `Switch(config)# logging 125.1.1.100`	Logs messages to a UNIX syslog server host. *host* specifies the name or IP address of the host to be used as the syslog server. To build a list of syslog servers that receive logging messages, enter this command more than once.
Step 4	**logging file flash:** *filename* [*max-file-size* [*min-file-size*]] [*severity-level-number* | *type*] **Example:** `Switch(config)# logging file flash:log_msg.txt 40960 4096 3`	Stores log messages in a file in flash memory on a standalone switch or, in the case of a switch stack, on the . • *filename*—Enters the log message filename. • (Optional) **max-file-size** —Specifies the maximum logging file size. The range is 4096 to 2147483647. The default is 4096 bytes. • (Optional) *min-file-size*—Specifies the minimum logging file size. The range is 1024 to 2147483647. The default is 2048 bytes. • (Optional) *severity-level-number* | *type*—Specifies either the logging severity level or the logging type. The severity range is 0 to 7.
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 6	**terminal monitor**	Logs messages to a nonconsole terminal during the current session.

	Command or Action	Purpose
	Example: `Switch# terminal monitor`	Terminal parameter-setting commands are set locally and do not remain in effect after the session has ended. You must perform this step for each session to see the debugging messages.

Synchronizing Log Messages

You can synchronize unsolicited messages and **debug** privileged EXEC command output with solicited device output and prompts for a specific console port line or virtual terminal line. You can identify the types of messages to be output asynchronously based on the level of severity. You can also configure the maximum number of buffers for storing asynchronous messages for the terminal after which messages are dropped.

When synchronous logging of unsolicited messages and **debug** command output is enabled, unsolicited device output appears on the console or printed after solicited device output appears or is printed. Unsolicited messages and **debug** command output appears on the console after the prompt for user input is returned. Therefore, unsolicited messages and **debug** command output are not interspersed with solicited device output and prompts. After the unsolicited messages appear, the console again displays the user prompt.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **line** [**console** | **vty**] *line-number* [*ending-line-number*]
3. **logging synchronous** [**level** [*severity-level* | **all**] | **limit** *number-of-buffers*]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**line** [**console** | **vty**] *line-number* [*ending-line-number*] **Example:** `Switch(config)# line console`	Specifies the line to be configured for synchronous logging of messages. • **console** —Specifies configurations that occur through the switch console port or the Ethernet management port. • **line vty** *line-number*—Specifies which vty lines are to have synchronous logging enabled. You use a vty connection for configurations that occur through a Telnet session. The range of line numbers is from 0 to 15.

	Command or Action	Purpose
		You can change the setting of all 16 vty lines at once by entering: line vty 0 15 You can also change the setting of the single vty line being used for your current connection. For example, to change the setting for vty line 2, enter: line vty 2 When you enter this command, the mode changes to line configuration.
Step 3	**logging synchronous** [**level** [*severity-level* | **all**] | **limit** *number-of-buffers*] **Example:** `Switch(config)# logging synchronous level 3 limit 1000`	Enables synchronous logging of messages. • (Optional) **level** *severity-level***—**Specifies the message severity level. Messages with a severity level equal to or higher than this value are printed asynchronously. Low numbers mean greater severity and high numbers mean lesser severity. The default is 2. • (Optional) **level all—**Specifies that all messages are printed asynchronously regardless of the severity level. • (Optional) **limit** *number-of-buffers***—**Specifies the number of buffers to be queued for the terminal after which new messages are dropped. The range is 0 to 2147483647. The default is 20.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Disabling Message Logging

Message logging is enabled by default. It must be enabled to send messages to any destination other than the console. When enabled, log messages are sent to a logging process, which logs messages to designated locations asynchronously to the processes that generated the messages.

Disabling the logging process can slow down the switch because a process must wait until the messages are written to the console before continuing. When the logging process is disabled, messages appear on the console as soon as they are produced, often appearing in the middle of command output.

The **logging synchronous** global configuration command also affects the display of messages to the console. When this command is enabled, messages appear only after you press **Return**.

To reenable message logging after it has been disabled, use the **logging on** global configuration command.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **no logging console**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**no logging console** **Example:** `Switch(config)# no logging console`	Disables message logging.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Enabling and Disabling Time Stamps on Log Messages

By default, log messages are not time-stamped.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. Use one of these commands:
    - **service timestamps log uptime**
    - **service timestamps log datetime**[**msec** | **localtime** | **show-timezone**]
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

Step 2	Use one of these commands:
	• **service timestamps log uptime**
	• **service timestamps log datetime**[**msec** | **localtime** | **show-timezone**]
	
	Example:
	`Switch(config)# service timestamps log uptime`
	
	or
	
	`Switch(config)# service timestamps log datetime`
		Enables log time stamps.
		• **log uptime—**Enables time stamps on log messages, showing the time since the system was rebooted.
		• **log datetime—**Enables time stamps on log messages. Depending on the options selected, the time stamp can include the date, time in milliseconds relative to the local time zone, and the time zone name.

Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Enabling and Disabling Sequence Numbers in Log Messages

If there is more than one log message with the same time stamp, you can display messages with sequence numbers to view these messages. By default, sequence numbers in log messages are not displayed.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **service sequence-numbers**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**service sequence-numbers** **Example:** `Switch(config)# service sequence-numbers`	Enables sequence numbers.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Defining the Message Severity Level

Limit messages displayed to the selected device by specifying the severity level of the message.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **logging console** *level*
3. **logging monitor** *level*
4. **logging trap** *level*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**logging console** *level*	Limits messages logged to the console.

	Command or Action	Purpose
	Example: `Switch(config)# logging console 3`	By default, the console receives debugging messages and numerically lower levels.
Step 3	**logging monitor** *level* **Example:** `Switch(config)# logging monitor 3`	Limits messages logged to the terminal lines. By default, the terminal receives debugging messages and numerically lower levels.
Step 4	**logging trap** *level* **Example:** `Switch(config)# logging trap 3`	Limits messages logged to the syslog servers. By default, syslog servers receive informational messages and numerically lower levels.
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Limiting Syslog Messages Sent to the History Table and to SNMP

This task explains how to limit syslog messages that are sent to the history table and to SNMP.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **logging history** *level*
3. **logging history size** *number*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**logging history** *level* **Example:** `Switch(config)# logging history 3`	Changes the default level of syslog messages stored in the history file and sent to the SNMP server. By default, **warnings**, **errors**, **critical**, **alerts**, and **emergencies** messages are sent.
Step 3	**logging history size** *number* **Example:** `Switch(config)# logging history size 200`	Specifies the number of syslog messages that can be stored in the history table. The default is to store one message. The range is 0 to 500 messages.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Logging Messages to a UNIX Syslog Daemon

This task is optional.

Note Some recent versions of UNIX syslog daemons no longer accept by default syslog packets from the network. If this is the case with your system, use the UNIX **man syslogd** command to decide what options must be added to or removed from the syslog command line to enable logging of remote syslog messages.

Before You Begin

- Log in as root.
- Before you can send system log messages to a UNIX syslog server, you must configure the syslog daemon on a UNIX server.

SUMMARY STEPS

1. Add a line to the file /etc/syslog.conf.
2. Enter these commands at the UNIX shell prompt.
3. Make sure the syslog daemon reads the new changes.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Add a line to the file /etc/syslog.conf. **Example:** `local7.debug /usr/adm/logs/`*cisco.log*	• **local7**—Specifies the logging facility. • **debug**—Specifies the syslog level. The file must already exist, and the syslog daemon must have permission to write to it.
Step 2	Enter these commands at the UNIX shell prompt. **Example:** `$ touch /var/log/`*cisco.log* `$ chmod 666 /var/log/`*cisco.log*	Creates the log file. The syslog daemon sends messages at this level or at a more severe level to this file.
Step 3	Make sure the syslog daemon reads the new changes. **Example:** ``$ kill -HUP `cat /etc/syslog.pid` ``	For more information, see the **man syslog.conf** and **man syslogd** commands on your UNIX system.

Monitoring and Maintaining System Message Logs

Monitoring Configuration Archive Logs

Command	Purpose
show archive log config {**all** | **number** [*end-number*] | **user** *username* [**session** *number*] *number* [*end-number*] | **statistics**} [**provisioning**]	Displays the entire configuration log or the log for specified parameters.

Configuration Examples for System Message Logs

Example: Stacking System Message

This example shows a partial switch system message for and a stack member (hostname *Switch-2*):

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/2, changed state to up
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet1/0/1, changed
state to down 2
*Mar  1 18:46:11: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
18:47:02: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
*Mar  1 18:48:50.483 UTC: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)

00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up (Switch-2)
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet2/0/1, changed state to up (Switch-2)
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet2/0/2, changed state to up (Switch-2)
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
(Switch-2)
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet2/0/1, changed
state to down 2 (Switch-2)
```

Example: Switch System Message

This example shows a partial switch system message on a switch:

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/2, changed state to up
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state
 to down 2
*Mar  1 18:46:11: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
18:47:02: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
*Mar  1 18:48:50.483 UTC: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
```

Additional References for System Message Logs

Related Documents

Related Topic	Document Title
System management commands	*System Management Command Reference (Catalyst 3850 Switches)*
Platform-independent command references	*Configuration Fundamentals Command Reference, Cisco IOS XE Release 3S (Catalyst 3850 Switches)*

Related Topic	Document Title
Platform-independent configuration information	*Configuration Fundamentals Configuration Guide, Cisco IOS XE Release 3S (Catalyst 3850 Switches)* *IP Addressing Configuration Guide Library, Cisco IOS XE Release 3S (Catalyst 3850 Switches)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information For System Message Logs

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 14

Configuring Online Diagnostics

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information About Configuring Online Diagnostics

Online Diagnostics

With online diagnostics, you can test and verify the hardware functionality of the switch while the switch is connected to a live network.

The online diagnostics contain packet switching tests that check different hardware components and verify the data path and the control signals.

The online diagnostics detect problems in these areas:

- Hardware components
- Interfaces (Ethernet ports and so forth)
- Solder joints

Online diagnostics are categorized as on-demand, scheduled, or health-monitoring diagnostics. On-demand diagnostics run from the CLI; scheduled diagnostics run at user-designated intervals or at specified times when the switch is connected to a live network; and health-monitoring runs in the background with user-defined intervals. By default, the health-monitoring test runs for every 30 seconds.

After you configure online diagnostics, you can manually start diagnostic tests or display the test results. You can also see which tests are configured for the switch or switch stack and the diagnostic tests that have already run.

How to Configure Online Diagnostics

Starting Online Diagnostic Tests

After you configure diagnostic tests to run on the switch, use the **diagnostic start** privileged EXEC command to begin diagnostic testing.

After starting the tests, you cannot stop the testing process.

Use this privileged EXEC command to manually start online diagnostic testing:

SUMMARY STEPS

1. **diagnostic start switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **complete** | **minimal** | **non-disruptive** | **per-port**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**diagnostic start switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **complete** | **minimal** | **non-disruptive** | **per-port**} **Example:** `Switch# diagnostic start switch 2 test basic`	Starts the diagnostic tests. The **switch** *number* keyword is supported only on stacking switches. The range is from 1 to 4. You can specify the tests by using one of these options: • *name*—Enters the name of the test. • *test-id*—Enters the ID number of the test. • *test-id-range*—Enters the range of test IDs by using integers separated by a comma and a hyphen. • **all**—Starts all of the tests. • **basic**— Starts the basic test suite. • **complete**—Starts the complete test suite.

	Command or Action	Purpose
		• **minimal—**Starts the minimal bootup test suite. • **non-disruptive—**Starts the non-disruptive test suite. • **per-port—**Starts the per-port test suite.

Configuring Online Diagnostics

You must configure the failure threshold and the interval between tests before enabling diagnostic monitoring.

Scheduling Online Diagnostics

You can schedule online diagnostics to run at a designated time of day or on a daily, weekly, or monthly basis for a switch. Use the **no** form of this command to remove the scheduling.

SUMMARY STEPS

1. **configure terminal**
2. **diagnostic schedule switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **complete** | **minimal** | **non-disruptive** | **per-port**} {**daily** | **on** *mm dd yyyy hh:mm* | **port** *inter-port-number port-number-list* | **weekly** *day-of-week hh:mm*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**diagnostic schedule switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **complete** | **minimal** | **non-disruptive** | **per-port**} {**daily** | **on** *mm dd yyyy hh:mm* | **port** *inter-port-number port-number-list* | **weekly** *day-of-week hh:mm*} **Example:** `Switch(config)# diagnostic schedule switch 3 test 1-5 on July 3 2013 23:10`	Schedules on-demand diagnostic tests for a specific day and time. The **switch** *number* keyword is supported only on stacking switches. The range is from 1 to 4. When specifying the tests to be scheduled, use these options: • *name*—Name of the test that appears in the **show diagnostic content** command output. • *test-id*—ID number of the test that appears in the **show diagnostic content** command output. • *test-id-range*—ID numbers of the tests that appear in the **show diagnostic content** command output.

	Command or Action	Purpose
		• **all**—All test IDs. • **basic**—Starts the basic on-demand diagnostic tests. • **complete**—Starts the complete test suite. • **minimal**—Starts the minimal bootup test suite. • **non-disruptive**—Starts the non-disruptive test suite. • **per-port**—Starts the per-port test suite. You can schedule the tests as follows: • Daily—Use the **daily** *hh:mm* parameter. • Specific day and time—Use the **on** *mm dd yyyy hh:mm* parameter. • Weekly—Use the **weekly** *day-of-week hh:mm* parameter.

Configuring Health-Monitoring Diagnostics

You can configure health-monitoring diagnostic testing on a switch while it is connected to a live network. You can configure the execution interval for each health-monitoring test, enable the switch to generate a syslog message because of a test failure, and enable a specific test.

By default, health monitoring is disabled, but the switch generates a syslog message when a test fails.

SUMMARY STEPS

1. **configure terminal**
2. **diagnostic monitor interval switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} *hh:mm:ss milliseconds day*
3. **diagnostic monitor syslog**
4. **diagnostic monitor threshold switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} **failure count** *count*
5. **diagnostic monitor switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**}
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**diagnostic monitor interval switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} *hh:mm:ss milliseconds day* **Example:** `Switch(config)# `**`diagnostic monitor interval switch 2 test 1 12:30:00 750 5`**	Configures the health-monitoring interval of the specified tests. The **switch** *number* keyword is supported only on stacking switches. When specifying the tests, use one of these parameters: • *name*—Name of the test that appears in the **show diagnostic content** command output. • *test-id*—ID number of the test that appears in the **show diagnostic content** command output. • *test-id-range*—ID numbers of the tests that appear in the **show diagnostic content** command output. • **all**—All of the diagnostic tests. When specifying the interval, set these parameters: • *hh:mm:ss*—Monitoring interval in hours, minutes, and seconds. The range for *hh* is 0 to 24, and the range for *mm* and *ss* is 0 to 60. • *milliseconds*—Monitoring interval in milliseconds (ms). The range is from 0 to 999. • *day*—Monitoring interval in the number of days. The range is from 0 to 20.
Step 3	**diagnostic monitor syslog** **Example:** `Switch(config)# `**`diagnostic monitor syslog`**	(Optional) Configures the switch to generate a syslog message when a health-monitoring test fails.
Step 4	**diagnostic monitor threshold switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} **failure count** *count* **Example:** `Switch(config)# `**`diagnostic monitor threshold switch 2 test 1 failure count 20`**	(Optional) Sets the failure threshold for the health-monitoring tests. When specifying the tests, use one of these parameters: • *name*—Name of the test that appears in the **show diagnostic content** command output. • *test-id*—ID number of the test that appears in the **show diagnostic content** command output. • *test-id-range*—ID numbers of the tests that appear in the **show diagnostic content** command output. • **all**—All of the diagnostic tests. The range for the failure threshold *count* is 0 to 99.
Step 5	**diagnostic monitor switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**}	Enables the specified health-monitoring tests. The **switch** *number* keyword is supported only on stacking switches. The range is from 1 to 9.

	Command or Action	Purpose
	Example: `Switch(config)# diagnostic monitor switch 2 test 1`	When specifying the tests, use one of these parameters: • *name*—Name of the test that appears in the **show diagnostic content** command output. • *test-id*—ID number of the test that appears in the **show diagnostic content** command output. • *test-id-range*—ID numbers of the tests that appear in the **show diagnostic content** command output. • **all**—All of the diagnostic tests.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Monitoring and Maintaining Online Diagnostics

Displaying Online Diagnostic Tests and Test Results

You can display the online diagnostic tests that are configured for the switch ro switch stack and check the test results by using the privileged EXEC **show** commands in this table:

Table 27: Commands for Diagnostic Test Configuration and Results

Command	Purpose
show diagnostic content switch [*number* | **all**]	Displays the online diagnostics configured for a switch. The **switch** [*number* | **all**] parameter is supported only on stacking switches.
show diagnostic status	Displays the currently running diagnostic tests.
show diagnostic result switch [*number* | **all**] [**detail** | **test** {*name* | *test-id* | *test-id-range* | **all**} [**detail**]]	Displays the online diagnostics test results. The **switch** [*number* | **all**] parameter is supported only on stacking switches.
show diagnostic switch [*number* | **all**] [**detail**]	Displays the online diagnostics test results. The **switch** [*number* | **all**] parameter is supported only on stacking switches.

Command	Purpose
show diagnostic schedule switch [*number* | **all**]	Displays the online diagnostics test schedule. The **switch** [*number* | **all**] parameter is supported only on stacking switches.
show diagnostic post	Displays the POST results. (The output is the same as the **show post** command output.)

Configuration Examples for Online Diagnostic Tests

Examples: Start Diagnostic Tests

This example shows how to start a diagnostic test by using the test name:

```
Switch# diagnostic start switch 2 test TestInlinePwrCtlr
```

This example shows how to start all of the basic diagnostic tests:

```
Switch# diagnostic start switch 1 test all
```

Example: Configure a Health Monitoring Test

This example shows how to configure a health-monitoring test:

```
Switch(config)# diagnostic monitor threshold switch 1 test 1 failure count 50
Switch(config)# diagnostic monitor interval switch 1 test TestPortAsicStackPortLoopback
```

Examples: Schedule Diagnostic Test

This example shows how to schedule diagnostic testing for a specific day and time on a specific switch:

```
Switch(config)# diagnostic schedule test DiagThermalTest on June 3 2013  22:25
```

This example shows how to schedule diagnostic testing to occur weekly at a certain time on a specific switch:

```
Switch(config)# diagnostic schedule switch 1 test 1,2,4-6 weekly saturday 10:30
```

Examples: Displaying Online Diagnostics

This example shows how to display on demand diagnostic settings:

```
Switch# show diagnostic ondemand settings
```

```
Test iterations = 1
Action on test failure = continue
```

This example shows how to display diagnostic events for errors:

```
Switch# show diagnostic events event-type error

Diagnostic events (storage for 500 events, 0 events recorded)
Number of events matching above criteria = 0

No diagnostic log entry exists.
```

This example shows how to display the description for a diagnostic test:

```
Switch# show diagnostic description switch 1 test all

DiagGoldPktTest :
        The GOLD packet Loopback test verifies the MAC level loopback
        functionality. In this test, a GOLD packet, for which doppler
        provides the support in hardware, is sent. The packet loops back
        at MAC level and is matched against the stored packet. It is a non
        -disruptive test.

DiagThermalTest :
        This test verifies the temperature reading from the sensor is below the yellow
        temperature threshold. It is a non-disruptive test and can be run as a health
monitoring test.

DiagFanTest :
        This test verifies all fan modules have been inserted and working properly on the
board
        It is a non-disruptive test and can be run as a health monitoring test.

DiagPhyLoopbackTest :
        The PHY Loopback test verifies the PHY level loopback
        functionality. In this test, a packet is sent which loops back
        at PHY level and is matched against the stored packet. It is a
        disruptive test and cannot be run as a health monitoring test.

DiagScratchRegisterTest :
        The Scratch Register test monitors the health of application-specific
        integrated circuits (ASICs) by writing values into registers and reading
        back the values from these registers. It is a non-disruptive test and can
        be run as a health monitoring test.

DiagPoETest :
        This test checks the PoE controller functionality. This is a disruptive test
        and should not be performed during normal switch operation.

DiagStackCableTest :
        This test verifies the stack ring loopback functionality
        in the stacking environment. It is a disruptive test and
        cannot be run as a health monitoring test.

DiagMemoryTest :
        This test runs the exhaustive ASIC memory test during normal switch operation
        NG3K utilizes mbist for this test. Memory test is very disruptive
        in nature and requires switch reboot after the test.

Switch#
```

This example shows how to display the boot up level:

```
Switch# show diagnostic bootup level

Current bootup diagnostic level: minimal
```

```
Switch#
```

Additional References for Online Diagnostics

Related Documents

Related Topic	Document Title
System management commands	*System Management Command Reference (Catalyst 3850 Switches)*
Platform-independent command reference	*Configuration Fundamentals Command Reference, Cisco IOS XE Release 3S (Catalyst 3850 Switches)*
Platform-independent configuration information	*Configuration Fundamentals Configuration Guide, Cisco IOS XE Release 3S (Catalyst 3850 Switches)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Configuring Online Diagnostics

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 15

Troubleshooting the Software Configuration

This chapter describes how to identify and resolve software problems related to the Cisco IOS software on the switch. Depending on the nature of the problem, you can use the command-line interface (CLI), Device Manager, or Network Assistant to identify and solve problems.

Additional troubleshooting information, such as LED descriptions, is provided in the hardware installation guide.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information About Troubleshooting the Software Configuration

Software Failure on a Switch

Switch software can be corrupted during an upgrade by downloading the incorrect file to the switch, and by deleting the image file. In all of these cases, the switch does not pass the power-on self-test (POST), and there is no connectivity.

Related Topics

Recovering from a Software Failure, on page 215

Lost or Forgotten Password on a Switch

The default configuration for the switch allows an end user with physical access to the switch to recover from a lost password by interrupting the boot process during power-on and by entering a new password. These recovery procedures require that you have physical access to the switch.

Note On these switches, a system administrator can disable some of the functionality of this feature by allowing an end user to reset a password only by agreeing to return to the default configuration. If you are an end user trying to reset a password when password recovery has been disabled, a status message reminds you to return to the default configuration during the recovery process.

Related Topics

Recovering from a Lost or Forgotten Password, on page 217

Power over Ethernet Ports

A Power over Ethernet (PoE) switch port automatically supplies power to one of these connected devices if the switch detects that there is no power on the circuit:

- a Cisco pre-standard powered device (such as a Cisco IP Phone or a Cisco Aironet Access Point)
- an IEEE 802.3af-compliant powered device
- an IEEE 802.3at-compliant powered device

A powered device can receive redundant power when it is connected to a PoE switch port and to an AC power source. The device does not receive redundant power when it is only connected to the PoE port.

After the switch detects a powered device, the switch determines the device power requirements and then grants or denies power to the device. The switch can also detect the real-time power consumption of the device by monitoring and policing the power usage.

For more information, see the "Configuring PoE" chapter in the *Interface and Hardware Component Configuration Guide (Catalyst 3850 Switches)* .

Related Topics

Scenarios to Troubleshoot Power over Ethernet (PoE), on page 229

Disabled Port Caused by Power Loss

If a powered device (such as a Cisco IP Phone 7910) that is connected to a PoE switch port and powered by an AC power source loses power from the AC power source, the device might enter an error-disabled state. To recover from an error-disabled state, enter the **shutdown** interface configuration command, and then enter the **no shutdown** interface command. You can also configure automatic recovery on the switch to recover from the error-disabled state.

On a switch, the **errdisable recovery cause loopback** and the **errdisable recovery interval** *seconds* global configuration commands automatically take the interface out of the error-disabled state after the specified period of time.

Disabled Port Caused by False Link-Up

If a Cisco powered device is connected to a port and you configure the port by using the **power inline never** interface configuration command, a false link-up can occur, placing the port into an error-disabled state. To take the port out of the error-disabled state, enter the **shutdown** and the **no shutdown** interface configuration commands.

You should not connect a Cisco powered device to a port that has been configured with the **power inline never** command.

Ping

The switch supports IP ping, which you can use to test connectivity to remote hosts. Ping sends an echo request packet to an address and waits for a reply. Ping returns one of these responses:

- Normal response—The normal response (hostname is alive) occurs in 1 to 10 seconds, depending on network traffic.
- Destination does not respond—If the host does not respond, a no-answer message is returned.
- Unknown host—If the host does not exist, an unknown host message is returned.
- Destination unreachable—If the default gateway cannot reach the specified network, a destination-unreachable message is returned.
- Network or host unreachable—If there is no entry in the route table for the host or network, a network or host unreachable message is returned.

Related Topics

Executing Ping, on page 224

Example: Pinging an IP Host, on page 231

Layer 2 Traceroute

The Layer 2 traceroute feature allows the switch to identify the physical path that a packet takes from a source device to a destination device. Layer 2 traceroute supports only unicast source and destination MAC addresses.

Traceroute finds the path by using the MAC address tables of the switches in the path. When the switch detects a device in the path that does not support Layer 2 traceroute, the switch continues to send Layer 2 trace queries and lets them time out.

The switch can only identify the path from the source device to the destination device. It cannot identify the path that a packet takes from source host to the source device or from the destination device to the destination host.

Layer 2 Traceroute Guidelines

- Cisco Discovery Protocol (CDP) must be enabled on all the devices in the network. For Layer 2 traceroute to function properly, do not disable CDP.

  If any devices in the physical path are transparent to CDP, the switch cannot identify the path through these devices.

- A switch is reachable from another switch when you can test connectivity by using the **ping** privileged EXEC command. All switches in the physical path must be reachable from each other.
- The maximum number of hops identified in the path is ten.
- You can enter the **traceroute mac** or the **traceroute mac ip** privileged EXEC command on a switch that is not in the physical path from the source device to the destination device. All switches in the path must be reachable from this switch.
- The **traceroute mac** command output shows the Layer 2 path only when the specified source and destination MAC addresses belong to the same VLAN. If you specify source and destination MAC addresses that belong to different VLANs, the Layer 2 path is not identified, and an error message appears.
- If you specify a multicast source or destination MAC address, the path is not identified, and an error message appears.
- If the source or destination MAC address belongs to multiple VLANs, you must specify the VLAN to which both the source and destination MAC addresses belong. If the VLAN is not specified, the path is not identified, and an error message appears.
- The **traceroute mac ip** command output shows the Layer 2 path when the specified source and destination IP addresses belong to the same subnet. When you specify the IP addresses, the switch uses the Address Resolution Protocol (ARP) to associate the IP addresses with the corresponding MAC addresses and the VLAN IDs.
  - If an ARP entry exists for the specified IP address, the switch uses the associated MAC address and identifies the physical path.
  - If an ARP entry does not exist, the switch sends an ARP query and tries to resolve the IP address. If the IP address is not resolved, the path is not identified, and an error message appears.
- When multiple devices are attached to one port through hubs (for example, multiple CDP neighbors are detected on a port), the Layer 2 traceroute feature is not supported. When more than one CDP neighbor is detected on a port, the Layer 2 path is not identified, and an error message appears.
- This feature is not supported in Token Ring VLANs.

IP Traceroute

You can use IP traceroute to identify the path that packets take through the network on a hop-by-hop basis. The command output displays all network layer (Layer 3) devices, such as routers, that the traffic passes through on the way to the destination.

Your switches can participate as the source or destination of the **traceroute** privileged EXEC command and might or might not appear as a hop in the **traceroute** command output. If the switch is the destination of the traceroute, it is displayed as the final destination in the traceroute output. Intermediate switches do not show up in the traceroute output if they are only bridging the packet from one port to another within the same VLAN. However, if the intermediate switch is a multilayer switch that is routing a particular packet, this switch shows up as a hop in the traceroute output.

The **traceroute** privileged EXEC command uses the Time To Live (TTL) field in the IP header to cause routers and servers to generate specific return messages. Traceroute starts by sending a User Datagram Protocol (UDP) datagram to the destination host with the TTL field set to 1. If a router finds a TTL value of 1 or 0, it drops the datagram and sends an Internet Control Message Protocol (ICMP) time-to-live-exceeded message to the sender. Traceroute finds the address of the first hop by examining the source address field of the ICMP time-to-live-exceeded message.

To identify the next hop, traceroute sends a UDP packet with a TTL value of 2. The first router decrements the TTL field by 1 and sends the datagram to the next router. The second router sees a TTL value of 1, discards the datagram, and returns the time-to-live-exceeded message to the source. This process continues until the TTL is incremented to a value large enough for the datagram to reach the destination host (or until the maximum TTL is reached).

To learn when a datagram reaches its destination, traceroute sets the UDP destination port number in the datagram to a very large value that the destination host is unlikely to be using. When a host receives a datagram destined to itself containing a destination port number that is unused locally, it sends an ICMP *port-unreachable* error to the source. Because all errors except port-unreachable errors come from intermediate hops, the receipt of a port-unreachable error means that this message was sent by the destination port.

Related Topics

Executing IP Traceroute, on page 225

Example: Performing a Traceroute to an IP Host, on page 232

Time Domain Reflector Guidelines

You can use the Time Domain Reflector (TDR) feature to diagnose and resolve cabling problems. When running TDR, a local device sends a signal through a cable and compares the reflected signal to the initial signal.

TDR is supported only on 10/100/1000 copper Ethernet ports. It is not supported on 10-Gigabit Ethernet ports and on SFP module ports.

TDR can detect these cabling problems:

- Open, broken, or cut twisted-pair wires—The wires are not connected to the wires from the remote device.
- Shorted twisted-pair wires—The wires are touching each other or the wires from the remote device. For example, a shorted twisted pair can occur if one wire of the twisted pair is soldered to the other wire.

If one of the twisted-pair wires is open, TDR can find the length at which the wire is open.

Use TDR to diagnose and resolve cabling problems in these situations:

- Replacing a switch
- Setting up a wiring closet
- Troubleshooting a connection between two devices when a link cannot be established or when it is not operating properly

When you run TDR, the switch reports accurate information in these situations:

- The cable for the gigabit link is a solid-core cable.
- The open-ended cable is not terminated.

When you run TDR, the switch does not report accurate information in these situations:

- The cable for the gigabit link is a twisted-pair cable or is in series with a solid-core cable.
- The link is a 10-megabit or a 100-megabit link.
- The cable is a stranded cable.
- The link partner is a Cisco IP Phone.
- The link partner is not IEEE 802.3 compliant.

Debug Commands

Caution Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. It is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

All **debug** commands are entered in privileged EXEC mode, and most **debug** commands take no arguments.

Related Topics

Redirecting Debug and Error Message Output, on page 225

Example: Enabling All System Diagnostics, on page 233

Crashinfo Files

The crashinfo files save information that helps Cisco technical support representatives to debug problems that caused the Cisco IOS image to fail (crash). The switch generates two files at the time of the failure: full core and crashinfo.

The information in the crashinfo file includes the Cisco IOS image name and version that failed, a list of the processor registers, and a stack trace. You can provide this information to the Cisco technical support representative by using the **show tech-support** privileged EXEC command.

The file names have the following format:

```
[fullcore | crashinfo]_[process that crashed]_[date]-[timestamp]-UTC
```

From IOS, you can view the crashinfo files on each switch by using the following command:

```
Switch#  dir crashinfo?
crashinfo-1: crashinfo-2: crashinfo-3: crashinfo:
Switch#
```

For example, to access the crashinfo directory for switch 1, enter
`Switch dir crashinfo-1`

From the ROMMON prompt, you can view the crashinfo files by using the **dir** command:

```
Switch: dir sda1
```

The following is sample output of a crashinfo file

```
Switch# dir crashinfo:

Directory of crashinfo:/

   12  -rwx        2768  Dec 31 1969 16:00:15 -08:00  koops.dat
   15  -rwx           0  Jan 12 2000 22:53:40 -08:00  deleted_crash_files
   16  -rwx    4246576  Jan 12 2000 22:53:40 -08:00  crashinfo_stack-mgr_20000113-065250-UTC

   17  -rwx          50   Oct 2 2012 03:18:42 -08:00  last_crashinfo
   26  -rwx          39  Jan 22 2013 14:14:14 -08:00  last_systemreport
   18  -rwx    2866565  Jan 12 2000 22:53:41 -08:00  fullcore_stack-mgr_20000113-065250-UTC

   20  -rwx    4391796   Feb 1 2000 17:50:44 -08:00  crashinfo_stack-mgr_20000202-014954-UTC

   21  -rwx    2920325   Feb 1 2000 17:50:45 -08:00  fullcore_stack-mgr_20000202-014954-UTC
34817  -rw-    1050209  Jan 10 2013 20:26:23 -08:00  system-report_1_20130111-042535-UTC.gz
18434  -rw-    1016913  Jan 11 2013 10:35:28 -08:00  system-report_1_20130111-183440-UTC.gz
18435  -rw-    1136167  Jan 22 2013 14:14:11 -08:00  system-report_1_20130122-221322-UTC.gz
34821  -rw-    1094631   Jan 2 2013 17:59:23 -08:00  system-report_1_20130103-015835-UTC.gz

 6147  -rw-     967429   Jan 3 2013 10:32:44 -08:00  system-report_1_20130103-183156-UTC.gz
34824  -rwx          50  Jan 22 2013 14:14:14 -08:00  deleted_sysreport_files
6155  -rwx         373  Jan 22 2013 14:14:13 -08:00  last_systemreport_log

145898496 bytes total (18569216 bytes free)
stack3#

The file name of the most recent crashinfo file is stored in last_crashinfo.
The file name of the most recent system report is stored in last_systemreport.


Switch#
```

System Reports

When a switch crashes, a system report is automatically generated for each switch in the switch stack. The system report file captures all the trace buffers, and other system-wide logs found on the switch. System reports are located in the crashinfo directory in the following format:
`system-report_[switch number]_[date]-[timestamp]-UTC.gz`

After a switch crash, you should check if a system report file was generated. The name of the most recently generated system report file is stored in the last_systemreport file under the crashinfo directory. The system report and crashinfo files assist TAC when troubleshooting your issue.

Onboard Failure Logging on the Switch

You can use the onboard failure logging (OBFL) feature to collect information about the switch. The information includes uptime, temperature, and voltage information and helps Cisco technical support representatives to troubleshoot switch problems. We recommend that you keep OBFL enabled and do not erase the data stored in the flash memory.

By default, OBFL is enabled. It collects information about the switch and small form-factor pluggable (SFP) modules. The switch stores this information in the flash memory:

- CLI commands—Record of the OBFL CLI commands that are entered on a standalone switch or a switch stack member.
- Environment data—Unique device identifier (UDI) information for a standalone switch or a stack member and for all the connected FRU devices: the product identification (PID), the version identification (VID), and the serial number.
- Message—Record of the hardware-related system messages generated by a standalone switch or a stack member.
- Power over Ethernet (PoE)—Record of the power consumption of PoE ports on a standalone switch or a stack member.
- Temperature—Temperature of a standalone switch or a stack member.
- Uptime data—Time when a standalone switch or a stack member starts, the reason the switch restarts, and the length of time the switch has been running since it last restarted.
- Voltage—System voltages of a standalone switch or a stack member.

You should manually set the system clock or configure it by using Network Time Protocol (NTP).

When the switch is running, you can retrieve the OBFL data by using the **show logging onboard** privileged EXEC commands. If the switch fails, contact your Cisco technical support representative to find out how to retrieve the data.

Related Topics

Configuring OBFL, on page 226

Displaying OBFL Information, on page 227

Fan Failures

By default, the feature is disabled. When more than one of the fans fails in a field-replaceable unit (FRU) or in a power supply, the switch does not shut down, and this error message appears:

```
Multiple fan(FRU/PS) failure detected. System may get overheated. Change fan quickly.
```

The switch might overheat and shut down.

To enable the fan failures feature, enter the **system env fan-fail-action shut** privileged EXEC command. If more than one fan in the switch fails, the switch automatically shuts down, and this error message appears:

```
Faulty (FRU/PS) fans detected, shutting down system!
```

After the first fan shuts down, if the switch detects a second fan failure, the switch waits for 20 seconds before it shuts down.

To restart the switch, it must be power cycled.

Possible Symptoms of High CPU Utilization

Excessive CPU utilization might result in these symptoms, but the symptoms might also result from other causes:

- Spanning tree topology changes
- EtherChannel links brought down due to loss of communication
- Failure to respond to management requests (ICMP ping, SNMP timeouts, slow Telnet or SSH sessions)
- UDLD flapping
- IP SLAs failures because of SLAs responses beyond an acceptable threshold
- DHCP or IEEE 802.1x failures if the switch does not forward or respond to requests

How to Troubleshoot the Software Configuration

Recovering from a Software Failure

Before You Begin

This recovery procedure requires that you have physical access to the switch.

This procedure uses boot loader commands and TFTP to recover from a corrupted or incorrect image file.

Step 1 From your PC, download the software image file (*image.bin*) from Cisco.com.

Step 2 Load the software image to your TFTP server.

Step 3 Connect your PC to the switch Ethernet management port.

Step 4 Unplug the switch power cord.

Step 5 Press the **Mode** button, and at the same time, reconnect the power cord to the switch.

Step 6 From the bootloader (ROMMON) prompt, ensure that you can ping your TFTP server.

a) Set the IP address **switch: set IP_ADDR** *ip_address subnet_mask*

Example:

```
switch: set IP_ADDR 192.0.2.123/255.255.255.0
```

b) Set the default router IP address **switch: set DEFAULT_ROUTER** *ip_address*

Example:

```
switch: set DEFAULT_ROUTER 192.0.2.1
```

c) Verify that you can ping the TFTP server **switch: ping** *ip_address_of_TFTP_server*

Example:

```
switch: ping 192.0.2.15
ping 192.0.2.1 with 32 bytes of data...
Host 192.0.2.1 is alive.
switch:
```

Step 7 Verify that you have a recovery image in your recovery partition (sda9:).
This recovery image is required for recovery using the emergency-install feature.

Example:

```
switch: dir sda9:
Directory of sda9:/

    2  drwx  1024       .
    2  drwx  1024       ..
   11  -rw-  18923068   c3850-recovery.bin

36939776 bytes available (20830208 bytes used)
switch:
```

Step 8 From the bootloader (ROMMON) prompt, initiate the emergency-install feature that assists you in recovering the software image on your switch.
WARNING: The emergency install command will erase your entire boot flash!

Example:

```
Switch#
emergency-install
tftp://192.0.2.47/cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin

The bootflash will be erased during install operation, continue (y/n)?y
Starting emergency recovery
(tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SPA.03.02.00.SE.150-1.EX.bin)...
Reading full image into memory......................done
Nova Bundle Image
--------------------------------------
Kernel Address : 0x6042e5cc
Kernel Size : 0x318261/3244641
Initramfs Address : 0x60746830
Initramfs Size : 0xdb0fb9/14356409
Compression Format: .mzip

Bootable image at @ ram:0x6042e5cc
Bootable image segment 0 address range [0x81100000, 0x81b80000] is in range [0x80180000, 0x90000000].
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
File "sda9:c3850-recovery.bin" uncompressed and installed, entry point: 0x811060f0
Loading Linux kernel with entry point 0x811060f0 ...
Bootloader: Done loading app on core_mask: 0xf

### Launching Linux Kernel (flags = 0x5)



Initiating Emergency Installation of bundle
tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin


Downloading bundle
tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin...
Validating bundle
tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin...
Installing bundle
tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin...
```

```
Verifying bundle
tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin...
Package cat3k_caa-base..pkg is Digitally Signed
Package cat3k_caa-drivers.SPA.03.02.00.SE.pkg is Digitally Signed
Package cat3k_caa-infra.SPA.03.02.00.SE.pkg is Digitally Signed
Package cat3k_caa-iosd-universalk9.SPA.03.02.00.SE.pkg is Digitally Signed
Package cat3k_caa-platform.SPA.03.02.00.SE.pkg is Digitally Signed
Package cat3k_caa-wcm.SPA.03.02.00.SE.pkg is Digitally Signed
Preparing flash...
Syncing device...
Emergency Install successful... Rebooting
Restarting system.


Booting...(use DDR clock 667 MHz)Initializing and Testing RAM +++@@@@####...++@@++@@++@@++@
```

Related Topics

Software Failure on a Switch, on page 208

Recovering from a Lost or Forgotten Password

The default configuration for the switch allows an end user with physical access to the switch to recover from a lost password by interrupting the boot process during power-on and by entering a new password. These recovery procedures require that you have physical access to the switch.

Note On these switches, a system administrator can disable some of the functionality of this feature by allowing an end user to reset a password only by agreeing to return to the default configuration. If you are an end user trying to reset a password when password recovery has been disabled, a status message shows this during the recovery process.

SUMMARY STEPS

1. Connect a terminal or PC to the switch.
2. Set the line speed on the emulation software to 9600 baud.
3. Power off the standalone switch or the entire switch stack.
4. Reconnect the power cord to the or the . Within 15 seconds, press the **Mode** button while the System LED is still flashing green. Continue pressing the **Mode** button until all the system LEDs turn on and remain solid; then release the **Mode** button.
5. After recovering the password, reload the switch or the .
6. Power on the remaining switches in the stack.

DETAILED STEPS

Step 1 Connect a terminal or PC to the switch.

- Connect a terminal or a PC with terminal-emulation software to the switch console port. If you are recovering the password for a switch stack, connect to the console port of the or
- Connect a PC to the Ethernet management port. If you are recovering the password for a switch stack, connect to the Ethernet management port of a stack member .

Step 2 Set the line speed on the emulation software to 9600 baud.

Step 3 Power off the standalone switch or the entire switch stack.

Step 4 Reconnect the power cord to the or the . Within 15 seconds, press the **Mode** button while the System LED is still flashing green. Continue pressing the **Mode** button until all the system LEDs turn on and remain solid; then release the **Mode** button.

-

```
Switch:
Xmodem file system is available.
Base ethernet MAC Address: 20:37:06:4d:e9:80
Verifying bootloader digital signature.


The system has been interrupted prior to loading the operating
system software, console will be reset to 9600 baud rate.
```

proceed to the *Procedure with Password Recovery Enabled* section, and follow the steps.

Step 5 After recovering the password, reload the switch or the .
On a switch:

```
Switch> reload
Proceed with reload? [confirm] y
```

On the active switch:

```
Switch> reload slot <stack-active-member-number>
Proceed with reload? [confirm] y
```

Step 6 Power on the remaining switches in the stack.

Related Topics

Lost or Forgotten Password on a Switch, on page 208

Procedure with Password Recovery Enabled

If the password-recovery operation is enabled, this message appears:

Step 1 Initialize the flash file system.

```
Switch: flash_init
```

Step 2 Ignore the startup configuration with the following command:

```
Switch: SWITCH_IGNORE_STARTUP_CFG=1
```

Step 3 Boot the switch with the *packages.conf* file from flash.

```
Switch: boot flash:packages.conf
```

Step 4 Terminate the initial configuration dialog by answering **No**.

```
Would you like to enter the initial configuration dialog? [yes/no]: No
```

Step 5 At the switch prompt, enter privileged EXEC mode.

```
Switch> enable
Switch#
```

Step 6 Copy the startup configuration to running configuration.

```
Switch# copy startup-config running-config Destination filename [running-config]?
```

Press Return in response to the confirmation prompts. The configuration file is now reloaded, and you can change the password.

Step 7 Enter global configuration mode and change the **enable** password.

```
Switch# configure terminal
Switch(config)#
```

Step 8 Write the running configuration to the startup configuration file.

```
Switch# copy running-config startup-config
```

Step 9 Confirm that manual boot mode is enabled.

```
Switch# show boot

 BOOT variable = flash:packages.conf;
```

```
 Manual Boot = yes
 Enable Break = yes
```

Step 10 Reload the switch.

```
Switch# reload
```

Step 11 Return the Bootloader parameters (previously changed in Steps 2 and 3) to their original values.

```
Switch: switch: SWITCH_IGNORE_STARTUP_CFG=0
```

Step 12 Boot the switch with the *packages.conf* file from flash.

```
Switch: boot flash:packages.conf
```

Step 13 After the switch boots up, disable manual boot on the switch.

```
Switch(config)# no boot manual
```

Procedure with Password Recovery Disabled

If the password-recovery mechanism is disabled, this message appears:

```
The password-recovery mechanism has been triggered, but
is currently disabled.  Access to the boot loader prompt
through the password-recovery mechanism is disallowed at
this point.  However, if you agree to let the system be
reset back to the default system configuration, access
to the boot loader prompt can still be allowed.

Would you like to reset the system back to the default configuration (y/n)?
```

Caution Returning the switch to the default configuration results in the loss of all existing configurations. We recommend that you contact your system administrator to verify if there are backup switch and VLAN configuration files.

- If you enter **n** (no), the normal boot process continues as if the **Mode** button had not been pressed; you cannot access the boot loader prompt, and you cannot enter a new password. You see the message:

```
Press Enter to continue........
```

- If you enter **y** (yes), the configuration file in flash memory and the VLAN database file are deleted. When the default configuration loads, you can reset the password.

Step 1 Choose to continue with password recovery and delete the existing configuration:

```
Would you like to reset the system back to the default configuration (y/n)? Y
```

Step 2 Display the contents of flash memory:

```
Switch: dir flash:
```

The switch file system appears.

```
Directory of flash:/
.
.
.i'
15494  drwx        4096   Jan 1 2000 00:20:20 +00:00  kirch
15508  -rw-   258065648   Sep 4 2013 14:19:03 +00:00
cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin
162196684
```

Step 3 Boot up the system:

```
Switch: boot
```

You are prompted to start the setup program. To continue with password recovery, enter **N** at the prompt:

```
Continue with the configuration dialog? [yes/no]: N
```

Step 4 At the switch prompt, enter privileged EXEC mode:

```
Switch> enable
```

Step 5 Enter global configuration mode:

```
Switch# configure terminal
```

Step 6 Change the password:

```
Switch(config)# enable secret password
```

The secret password can be from 1 to 25 alphanumeric characters, can start with a number, is case sensitive, and allows spaces but ignores leading spaces.

Step 7 Return to privileged EXEC mode:

```
Switch(config)# exit
Switch#
```

Note Before continuing to Step 9, power on any connected stack members and wait until they have completely initialized.

Step 8 Write the running configuration to the startup configuration file:

```
Switch# copy running-config startup-config
```

The new password is now in the startup configuration.

Step 9 You must now reconfigure the switch. If the system administrator has the backup switch and VLAN configuration files available, you should use those.

Preventing Switch Stack Problems

To prevent switch stack problems, you should do the following:

- Make sure that the switches that you add to or remove from the switch stack are powered off. For all powering considerations in switch stacks, see the "Switch Installation" chapter in the hardware installation guide.
- Press the **Mode** button on a stack member until the Stack mode LED is on. The last two port LEDs on the switch should be green. Depending on the switch model, the last two ports are either 10/100/1000 ports or small form-factor pluggable (SFP) module. If one or both of the last two port LEDs are not green, the stack is not operating at full bandwidth.
- We recommend using only one CLI session when managing the switch stack. Be careful when using multiple CLI sessions to the . Commands that you enter in one session are not displayed in the other sessions. Therefore, it is possible that you might not be able to identify the session from which you entered a command.
- Manually assigning stack member numbers according to the placement of the switches in the stack can make it easier to remotely troubleshoot the switch stack. However, you need to remember that the switches have manually assigned numbers if you add, remove, or rearrange switches later. Use the **switch** *current-stack-member-number* **renumber** *new-stack-member-number* global configuration command to manually assign a stack member number.

If you replace a stack member with an identical model, the new switch functions with the exact same configuration as the replaced switch. This is also assuming the new switch is using the same member number as the replaced switch.

Removing powered-on stack members causes the switch stack to divide (partition) into two or more switch stacks, each with the same configuration. If you want the switch stacks to remain separate, change the IP address or addresses of the newly created switch stacks. To recover from a partitioned switch stack, follow these steps:

1. Power off the newly created switch stacks.
2. Reconnect them to the original switch stack through their StackWise Plus ports.
3. Power on the switches.

Preventing Autonegotiation Mismatches

The IEEE 802.3ab autonegotiation protocol manages the switch settings for speed (10 Mb/s, 100 Mb/s, and 1000 Mb/s, excluding SFP module ports) and duplex (half or full). There are situations when this protocol can incorrectly align these settings, reducing performance. A mismatch occurs under these circumstances:

- A manually set speed or duplex parameter is different from the manually set speed or duplex parameter on the connected port.
- A port is set to autonegotiate, and the connected port is set to full duplex with no autonegotiation.

To maximize switch performance and ensure a link, follow one of these guidelines when changing the settings for duplex and speed:

- Let both ports autonegotiate both speed and duplex.
- Manually set the speed and duplex parameters for the ports on both ends of the connection.

Note If a remote device does not autonegotiate, configure the duplex settings on the two ports to match. The speed parameter can adjust itself even if the connected port does not autonegotiate.

Troubleshooting SFP Module Security and Identification

Cisco small form-factor pluggable (SFP) modules have a serial EEPROM that contains the module serial number, the vendor name and ID, a unique security code, and cyclic redundancy check (CRC). When an SFP module is inserted in the switch, the switch software reads the EEPROM to verify the serial number, vendor name and vendor ID, and recompute the security code and CRC. If the serial number, the vendor name or vendor ID, the security code, or CRC is invalid, the software generates a security error message and places the interface in an error-disabled state.

Note The security error message references the GBIC_SECURITY facility. The switch supports SFP modules and does not support GBIC modules. Although the error message text refers to GBIC interfaces and modules, the security messages actually refer to the SFP modules and module interfaces.

If you are using a non-Cisco SFP module, remove the SFP module from the switch, and replace it with a Cisco module. After inserting a Cisco SFP module, use the **errdisable recovery cause gbic-invalid** global configuration command to verify the port status, and enter a time interval for recovering from the error-disabled state. After the elapsed interval, the switch brings the interface out of the error-disabled state and retries the operation.

If the module is identified as a Cisco SFP module, but the system is unable to read vendor-data information to verify its accuracy, an SFP module error message is generated. In this case, you should remove and reinsert the SFP module. If it continues to fail, the SFP module might be defective.

Monitoring SFP Module Status

You can check the physical or operational status of an SFP module by using the **show interfaces transceiver** privileged EXEC command. This command shows the operational status, such as the temperature and the current for an SFP module on a specific interface and the alarm status. You can also use the command to check the speed and the duplex settings on an SFP module.

Executing Ping

If you attempt to ping a host in a different IP subnetwork, you must define a static route to the network or have IP routing configured to route between those subnets.

IP routing is disabled by default on all switches.

Note Though other protocol keywords are available with the **ping** command, they are not supported in this release.

Use this command to ping another device on the network from the switch:

Command	Purpose
ping ip *host* | *address* `Switch# ping 172.20.52.3`	Pings a remote host through IP or by supplying the hostname or network address.

Related Topics

Ping, on page 209
Example: Pinging an IP Host, on page 231

Monitoring Temperature

The switch monitors the temperature conditions and uses the temperature information to control the fans.

Use the **show env temperature status** privileged EXEC command to display the temperature value, state, and thresholds. The temperature value is the temperature in the switch (not the external temperature).You can configure only the yellow threshold level (in Celsius) by using the **system env temperature threshold yellow** *value* global configuration command to set the difference between the yellow and red thresholds. You cannot configure the green or red thresholds.

Monitoring the Physical Path

You can monitor the physical path that a packet takes from a source device to a destination device.

Table 28: Monitoring the Physical Path

Command	Purpose
tracetroute mac [**interface** *interface-id*] {*source-mac-address*} [**interface** *interface-id*] {*destination-mac-address*} [**vlan** *vlan-id*] [**detail**]	Displays the Layer 2 path taken by the packets from the specified source MAC address to the specified destination MAC address.

Command	Purpose
tracetroute mac ip {*source-ip-address* | *source-hostname*} {*destination-ip-address* | *destination-hostname*} [**detail**]	Displays the Layer 2 path taken by the packets from the specified source IP address or hostname to the specified destination IP address or hostname.

Executing IP Traceroute

Note Though other protocol keywords are available with the **traceroute** privileged EXEC command, they are not supported in this release.

Command	Purpose
traceroute ip *host* `Switch# traceroute ip 192.51.100.1`	Traces the path that packets take through the network.

Related Topics

IP Traceroute , on page 211
Example: Performing a Traceroute to an IP Host, on page 232

Running TDR and Displaying the Results

When you run TDR on an interface, you can run it on the or a stack member.

To run TDR, enter the **test cable-diagnostics tdr interface** *interface-id* privileged EXEC command.

To display the results, enter the **show cable-diagnostics tdr interface** *interface-id* privileged EXEC command.

Redirecting Debug and Error Message Output

By default, the network server sends the output from **debug** commands and system error messages to the console. If you use this default, you can use a virtual terminal connection to monitor debug output instead of connecting to the console port or the Ethernet management port.

Possible destinations include the console, virtual terminals, internal buffer, and UNIX hosts running a syslog server. The syslog format is compatible with 4.3 Berkeley Standard Distribution (BSD) UNIX and its derivatives.

Note Be aware that the debugging destination you use affects system overhead. When you log messages to the console, very high overhead occurs. When you log messages to a virtual terminal, less overhead occurs. Logging messages to a syslog server produces even less, and logging to an internal buffer produces the least overhead of any method.

Related Topics

Debug Commands, on page 212

Using the show platform forward Command

The output from the **show platform forward** privileged EXEC command provides some useful information about the forwarding results if a packet entering an interface is sent through the system. Depending upon the parameters entered about the packet, the output provides lookup table results and port maps used to calculate forwarding destinations, bitmaps, and egress information.

Most of the information in the output from the command is useful mainly for technical support personnel, who have access to detailed information about the switch application-specific integrated circuits (ASICs). However, packet forwarding information can also be helpful in troubleshooting.

Configuring OBFL

Caution We recommend that you do not disable OBFL and that you do not remove the data stored in the flash memory.

- To enable OBFL, use the **hw-switch switch** [*switch-number*] **logging onboard** [**message level** *level*] global configuration command. On switches, the range for *switch-number* is from 1 to 9. Use the **message level** *level* parameter to specify the severity of the hardware-related messages that the switch generates and stores in the flash memory.
- To copy the OBFL data to the local network or a specific file system, use the **copy onboard switch** *switch-number* **url** *url-destination* privileged EXEC command.
- To disable OBFL, use the **no hw-switch switch** [*switch-number*] **logging onboard** [**message level**] global configuration command.
- To clear all the OBFL data in the flash memory except for the uptime and CLI command information, use the **clear onboard switch** *switch-number* privileged EXEC command.
- In a switch stack, you can enable OBFL on a standalone switch or on all stack members by using the **hw-switch switch** [*switch-number*] **logging onboard** [**message level** *level*] global configuration command.
- You can enable or disable OBFL on a member switch from the .

Related Topics

Onboard Failure Logging on the Switch, on page 214
Displaying OBFL Information, on page 227

Verifying Troubleshooting of the Software Configuration

Displaying OBFL Information

Table 29: Commands for Displaying OBFL Information

Command	Purpose
show onboard switch *switch-number* **clilog** `Switch# show onboard switch 1 clilog`	Displays the OBFL CLI commands that were entered on a standalone switch or the specified stack members.
show onboard switch *switch-number* **environment** `Switch# show onboard switch 1 environment`	Displays the UDI information for a standalone switch or the specified stack members and for all the connected FRU devices: the PID, the VID, and the serial number.
show onboard switch *switch-number* **message** `Switch# show onboard switch 1 message`	Displays the hardware-related messages generated by a standalone switch or the specified stack members.
show onboard switch *switch-number* **counter** `Switch# show onboard switch 1 counter`	Displays the counter information on a standalone switch or the specified stack members.
show onboard switch *switch-number* **temperature** `Switch# show onboard switch 1 temperature`	Displays the temperature of a standalone switch or the specified switch stack members.
show onboard switch *switch-number* **uptime** `Switch# show onboard switch 1 uptime`	Displays the time when a standalone switch or the specified stack members start, the reason the standalone switch or specified stack members restart, and the length of time that the standalone switch or specified stack members have been running since they last restarted.
show onboard switch *switch-number* **voltage** `Switch# show onboard switch 1 voltage`	Displays the system voltages of a standalone switch or the specified stack members.
show onboard switch *switch-number* **status** `Switch# show onboard switch 1 status`	Displays the status of a standalone switch or the specified stack members.

Related Topics

Onboard Failure Logging on the Switch, on page 214

Configuring OBFL, on page 226

Example: Verifying the Problem and Cause for High CPU Utilization

To determine if high CPU utilization is a problem, enter the **show processes cpu sorted** privileged EXEC command. Note the underlined information in the first line of the output example.

```
Switch# show processes cpu sorted
CPU utilization for five seconds: 8%/0%; one minute: 7%; five minutes: 8%
PID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min TTY Process
309 42289103 752750 56180 1.75% 1.20% 1.22% 0 RIP Timers
140 8820183 4942081 1784 0.63% 0.37% 0.30% 0 HRPC qos request
100 3427318 16150534 212 0.47% 0.14% 0.11% 0 HRPC pm-counters
192 3093252 14081112 219 0.31% 0.14% 0.11% 0 Spanning Tree
143 8 37 216 0.15% 0.01% 0.00% 0 Exec
...
<output truncated>
```

This example shows normal CPU utilization. The output shows that utilization for the last 5 seconds is 8%/0%, which has this meaning:

- The total CPU utilization is 8 percent, including both time running Cisco IOS processes and time spent handling interrupts.
- The time spent handling interrupts is zero percent.

Table 30: Troubleshooting CPU Utilization Problems

Type of Problem	Cause	Corrective Action
Interrupt percentage value is almost as high as total CPU utilization value.	The CPU is receiving too many packets from the network.	Determine the source of the network packet. Stop the flow, or change the switch configuration. See the section on “Analyzing Network Traffic.”
Total CPU utilization is greater than 50% with minimal time spent on interrupts.	One or more Cisco IOS process is consuming too much CPU time. This is usually triggered by an event that activated the process.	Identify the unusual event, and troubleshoot the root cause. See the section on “Debugging Active Processes.”

Scenarios for Troubleshooting the Software Configuration

Scenarios to Troubleshoot Power over Ethernet (PoE)

Table 31: Power over Ethernet Troubleshooting Scenarios

Symptom or Problem	Possible Cause and Solution
Only one port does not have PoE. Trouble is on only one switch port. PoE and non-PoE devices do not work on this port, but do on other ports.	Verify that the powered device works on another PoE port. Use the **show run**, or **show interface status** user EXEC commands to verify that the port is not shut down or error-disabled. **Note** Most switches turn off port power when the port is shut down, even though the IEEE specifications make this optional. Verify that the Ethernet cable from the powered device to the switch port is good: Connect a known good non-PoE Ethernet device to the Ethernet cable, and make sure that the powered device establishes a link and exchanges traffic with another host. Verify that the total cable length from the switch front panel to the powered device is not more than 100 meters. Disconnect the Ethernet cable from the switch port. Use a short Ethernet cable to connect a known good Ethernet device directly to this port on the switch front panel (not on a patch panel). Verify that it can establish an Ethernet link and exchange traffic with another host, or ping the port VLAN SVI. Next, connect a powered device to this port, and verify that it powers on. If a powered device does not power on when connected with a patch cord to the switch port, compare the total number of connected powered devices to the switch power budget (available PoE). Use the **show inline power** command to verify the amount of available power.

Symptom or Problem	Possible Cause and Solution
No PoE on all ports or a group of ports. Trouble is on all switch ports. Nonpowered Ethernet devices cannot establish an Ethernet link on any port, and PoE devices do not power on.	If there is a continuous, intermittent, or reoccurring alarm related to power, replace the power supply if possible it is a field-replaceable unit. Otherwise, replace the switch. If the problem is on a consecutive group of ports but not all ports, the power supply is probably not defective, and the problem could be related to PoE regulators in the switch. Use the **show log** privileged EXEC command to review alarms or system messages that previously reported PoE conditions or status changes. If there are no alarms, use the **show interface status** command to verify that the ports are not shut down or error-disabled. If ports are error-disabled, use the **shut** and **no shut** interface configuration commands to reenable the ports. Use the **show env power** and **show power inline** privileged EXEC commands to review the PoE status and power budget (available PoE). Review the running configuration to verify that **power inline never** is not configured on the ports. Connect a nonpowered Ethernet device directly to a switch port. Use only a short patch cord. Do not use the existing distribution cables. Enter the **shut** and **no shut** interface configuration commands, and verify that an Ethernet link is established. If this connection is good, use a short patch cord to connect a powered device to this port and verify that it powers on. If the device powers on, verify that all intermediate patch panels are correctly connected. Disconnect all but one of the Ethernet cables from switch ports. Using a short patch cord, connect a powered device to only one PoE port. Verify the powered device does not require more power than can be delivered by the switch port. Use the **show power inline** privileged EXEC command to verify that the powered device can receive power when the port is not shut down. Alternatively, watch the powered device to verify that it powers on. If a powered device can power on when only one powered device is connected to the switch, enter the **shut** and **no shut** interface configuration commands on the remaining ports, and then reconnect the Ethernet cables one at a time to the switch PoE ports. Use the **show interface status** and **show power inline** privileged EXEC commands to monitor inline power statistics and port status. If there is still no PoE at any port, a fuse might be open in the PoE section of the power supply. This normally produces an alarm. Check the log again for alarms reported earlier by system messages.

Symptom or Problem	Possible Cause and Solution
Cisco IP Phone disconnects or resets. After working normally, a Cisco phone or wireless access point intermittently reloads or disconnects from PoE.	Verify all electrical connections from the switch to the powered device. Any unreliable connection results in power interruptions and irregular powered device functioning such as erratic powered device disconnects and reloads. Verify that the cable length is not more than 100 meters from the switch port to the powered device. Notice what changes in the electrical environment at the switch location or what happens at the powered device when the disconnect occurs. Notice whether any error messages appear at the same time a disconnect occurs. Use the **show log** privileged EXEC command to review error messages. Verify that an IP phone is not losing access to the Call Manager immediately before the reload occurs. (It might be a network problem and not a PoE problem.) Replace the powered device with a non-PoE device, and verify that the device works correctly. If a non-PoE device has link problems or a high error rate, the problem might be an unreliable cable connection between the switch port and the powered device.
Non-Cisco powered device does not work on Cisco PoE switch. A non-Cisco powered device is connected to a Cisco PoE switch, but never powers on or powers on and then quickly powers off. Non-PoE devices work normally.	Use the **show power inline** command to verify that the switch power budget (available PoE) is not depleted before or after the powered device is connected. Verify that sufficient power is available for the powered device type before you connect it. Use the **show interface status** command to verify that the switch detects the connected powered device. Use the **show log** command to review system messages that reported an overcurrent condition on the port. Identify the symptom precisely: Does the powered device initially power on, but then disconnect? If so, the problem might be an initial surge-in (or *inrush*) current that exceeds a current-limit threshold for the port.

Related Topics

Power over Ethernet Ports, on page 208

Configuration Examples for Troubleshooting Software

Example: Pinging an IP Host

This example shows how to ping an IP host:

```
Switch# ping 172.20.52.3
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echoes to 172.20.52.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Switch#
```

Table 32: Ping Output Display Characters

Character	Description
!	Each exclamation point means receipt of a reply.
.	Each period means the network server timed out while waiting for a reply.
U	A destination unreachable error PDU was received.
C	A congestion experienced packet was received.
I	User interrupted test.
?	Unknown packet type.
&	Packet lifetime exceeded.

To end a ping session, enter the escape sequence (**Ctrl-^ X** by default). Simultaneously press and release the **Ctrl**, **Shift**, and **6** keys and then press the **X** key.

Related Topics

Ping, on page 209
Executing Ping, on page 224

Example: Performing a Traceroute to an IP Host

This example shows how to perform a **traceroute** to an IP host:

```
Switch# traceroute ip 192.0.2.10

Type escape sequence to abort.
Tracing the route to 192.0.2.10

  1 192.0.2.1 0 msec 0 msec 4 msec
  2 192.0.2.203 12 msec 8 msec 0 msec
  3 192.0.2.100 4 msec 0 msec 0 msec
  4 192.0.2.10 0 msec 4 msec 0 msec
```

The display shows the hop count, the IP address of the router, and the round-trip time in milliseconds for each of the three probes that are sent.

Table 33: Traceroute Output Display Characters

Character	Description
*	The probe timed out.
?	Unknown packet type.
A	Administratively unreachable. Usually, this output means that an access list is blocking traffic.
H	Host unreachable.
N	Network unreachable.
P	Protocol unreachable.
Q	Source quench.
U	Port unreachable.

To end a trace in progress, enter the escape sequence (**Ctrl-^ X** by default). Simultaneously press and release the **Ctrl**, **Shift**, and **6** keys and then press the **X** key.

Related Topics

IP Traceroute , on page 211
Executing IP Traceroute, on page 225

Example: Enabling All System Diagnostics

Caution Because debugging output takes priority over other network traffic, and because the **debug all** privileged EXEC command generates more output than any other **debug** command, it can severely diminish switch performance or even render it unusable. In virtually all cases, it is best to use more specific **debug** commands.

This command disables all-system diagnostics:

```
Switch# debug all
```

The **no debug all** privileged EXEC command disables all diagnostic output. Using the **no debug all** command is a convenient way to ensure that you have not accidentally left any **debug** commands enabled.

Related Topics

Debug Commands, on page 212

Additional References for Troubleshooting Software Configuration

Related Documents

Related Topic	Document Title
System management commands	*System Management Command Reference (Catalyst 3850 Switches)*
Platform-independent command reference	*Configuration Fundamentals Command Reference, Cisco IOS XE Release 3S (Catalyst 3850 Switches)*
Platform_independent configuration information	*Configuration Fundamentals Configuration Guide, Cisco IOS XE Release 3S (Catalyst 3850 Switches)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Troubleshooting Software Configuration

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

Related Topics

Finding Feature Information, on page 15

PART II

QoS

CHAPTER 16

Configuring QoS

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for QoS

Before configuring standard QoS, you must have a thorough understanding of these items:

- Standard QoS concepts.
- Wireless concepts and network topologies.

- Classic Cisco IOS QoS.
- Modular QoS CLI (MQC).
- Understanding of QoS implementation.
- The types of applications used and the traffic patterns on your network.
- Traffic characteristics and needs of your network. For example, is the traffic on your network bursty? Do you need to reserve bandwidth for voice and video streams?
- Bandwidth requirements and speed of the network.
- Location of congestion points in the network.

Restrictions for QoS on Wired Targets

A target is an entity where a policy is applied. You can apply a policy to either a wired or wireless target. A wired target can be either a port or VLAN. A wireless target can be either a port, radio, SSID, or client. Only port, SSID, and client policies are user configurable. Radio polices are not user configurable. Wireless QoS policies for port, radio, SSID, and client are applied in the downstream direction, and for upstream only SSID and client targets are supported. Downstream indicates that traffic is flowing from the switch to the wireless client. Upstream indicates that traffic is flowing from wireless client to the switch.

The following are restrictions for applying QoS features on the switch for the wired target:

- A maximum of 8 queuing classes are supported on the switch port for the wired target.
- A maximum of 63 policers are supported per policy on the wired port for the wired target.
- No more than two levels are supported in a QoS hierarchy.
- In a hierarchical policy, overlapping actions between parent and child are not allowed, except when a policy has the port shaper in the parent and queueing features in the child policy.
- A QoS policy cannot be attached to any EtherChannel interface.
- Policing in both the parent and child is not supported in a QoS hierarchy.
- Marking in both the parent and child is not supported in a QoS hierarchy.
- A mixture of queue limit and queue buffer in the same policy is not supported.

Note The queue-limit percent is not supported on the switch because the **queue-buffer** command handles this functionality. Queue limit is only supported with the DSCP and CoS extensions.

- The classification sequence for all wired queuing-based policies should be the same across all wired upstream ports (10-Gigabit Ethernet), and the same for all downstream wired ports (1-Gigabit Ethernet).
- Empty classes are not supported.
- Class-maps with empty actions are not supported.
- A maximum of 256 classes are supported per policy on the wired port for the wired target.
- The actions under a policer within a policy map have the following restrictions:

- ◦ The conform action must be transmit.
- ◦ The exceed/violate action for markdown type can only be cos2cos, prec2prec, dscp2dscp.
- ◦ The markdown types must be the same within a policy.

- Classification counters have the following specific restrictions:
  - ◦ Classification counters count packets instead of bytes.
  - ◦ Only QoS configurations with marking or policing trigger the classification counter.
  - ◦ The classification counter is not port based. This means that the classification counter aggregates all packets belonging to the same class of the same policy which attach to different interfaces.
  - ◦ As long as there is policing or marking action in the policy, the class-default will have classification counters.
  - ◦ When there are multiple match statements in a class, then the classification counter only shows the traffic counter for one of the match statements.

- Table maps have the following specific restrictions:
  - ◦ Only one table map for policing exceeding the markdown and one table map for policing violating the markdown per direction per target is supported.
  - ◦ Table maps must be configured under the class-default; table maps are unsupported for a user-defined class.

- Hierarchical policies are required for the following:
  - ◦ Port-shapers
  - ◦ Aggregate policers
  - ◦ PV policy
  - ◦ Parent shaping and child marking/policing

- For ports with wired targets, these are the only supported hierarchical policies:
  - ◦ Police chaining in the same policy is unsupported, except for wireless client.
  - ◦ Hierarchical queueing is unsupported in the same policy (port shaper is the exception).
  - ◦ In a parent class, all filters must have the same type. The child filter type must match the parent filter type with the following exceptions:
    - If the parent class is configured to match IP, then the child class can be configured to match the ACL.
    - If the parent class is configured to match CoS, then the child class can be configured to match the ACL.

- The **trust device** *device_type* interface configuration command is only supported in an auto-QoS configuration, and not as a stand-alone command on the switch. When using the **trust device** *device_type* interface configuration command in an auto-QoS configuration, if the connected peer device is not a corresponding device (defined as a device matching your trust policy), both CoS and DSCP values are set to "0" and any input policy will not take effect.

The following are restrictions for applying QoS features on the VLAN to the wired target:

- For a flat or nonhierarchical policy, only marking or a table map is supported.

The following are restrictions and considerations for applying QoS features on EtherChannel and channel member interfaces:

- QoS is not supported on an EtherChannel interface.
- QoS is supported on EtherChannel member interfaces in both ingress and egression directions. All EtherChannel members must have the same QoS policy applied. If the QoS policy is not the same, each individual policy on the different link acts independently.
- On attaching a service policy to channel members, the following warning message appears to remind the user to make sure the same policy is attached to all ports in the EtherChannel: ' Warning: add service policy will cause inconsistency with port xxx in ether channel xxx. '.
- Auto QoS is not supported on EtherChannel members.

Note On attaching a service policy to an EtherChannel, the following message appears on the console: ' Warning: add service policy will cause inconsistency with port xxx in ether channel xxx. '. This warning message should be expected. This warning message is a reminder to attach the same policy to other ports in the same EtherChannel. The same message will be seen during boot up. This message does not mean there is a discrepancy between the EtherChannel member ports.

Related Topics

Restrictions for QoS on Wireless Targets, on page 354

Information About QoS

QoS Overview

By configuring the quality of service (QoS), you can provide preferential treatment to specific types of traffic at the expense of other traffic types. Without QoS, the switch offers best-effort service to each packet, regardless of the packet contents or size. The switch sends the packets without any assurance of reliability, delay bounds, or throughput.

The following are specific features provided by QoS:

- Low latency
- Bandwidth guarantee
- Buffering capabilities and dropping disciplines
- Traffic policing
- Enables the changing of the attribute of the frame or packet header

- Relative services

Modular QoS Command-Line Interface

With the switch, QoS features are enabled through the Modular QoS command-line interface (MQC). The MQC is a command-line interface (CLI) structure that allows you to create traffic policies and attach these policies to interfaces. A traffic policy contains a traffic class and one or more QoS features. A traffic class is used to classify traffic, while the QoS features in the traffic policy determine how to treat the classified traffic. One of the main goals of MQC is to provide a platform-independent interface for configuring QoS across Cisco platforms.

QoS and IPv6 for Wireless

From this release onwards, the switch supports QoS for both IPv4 and IPv6 traffic, and client policies can now have IPv4 and IPv6 filters.

Supported QoS Features for Wired Access

The following table describes the supported QoS features for wired access.

Table 34: Supported QoS Features for Wired Access

Feature	Description
Supported targets	• Gigabit Ethernet • 10-Gigabit Ethernet • VLAN
Configuration sequence	QoS policy installed using the **service-policy** command.
Supported number of queues at port level	Up to 8 queues supported on a port. No Approximate Fair Dropping or Discard (AFD) support for wired targets.

Supported classification mechanism	• DSCP
	• IP precedence
	• CoS
	• QoS-group
	• ACL membership including:
	  ◦ IPv4 ACLs
	  ◦ IPv6 ACLS
	  ◦ MAC ACLs

Hierarchical QoS

The switch supports hierarchical QoS (HQoS). HQoS allows you to perform:

- Hierarchical classification— Traffic classification is based upon other classes.
- Hierarchical policing—The process of having the policing configuration at multiple levels in a hierarchical policy.
- Hierarchical shaping—Shaping can also be configured at multiple levels in the hierarchy.

Note Hierarchical shaping is only supported for the port shaper, where for the parent you only have a configuration for the class default, and the only action for the class default is shaping.

Related Topics

Examples: Hierarchical Classification, on page 314

QoS Implementation

Typically, networks operate on a best-effort delivery basis, which means that all traffic has equal priority and an equal chance of being delivered in a timely manner. When congestion occurs, all traffic has an equal chance of being dropped.

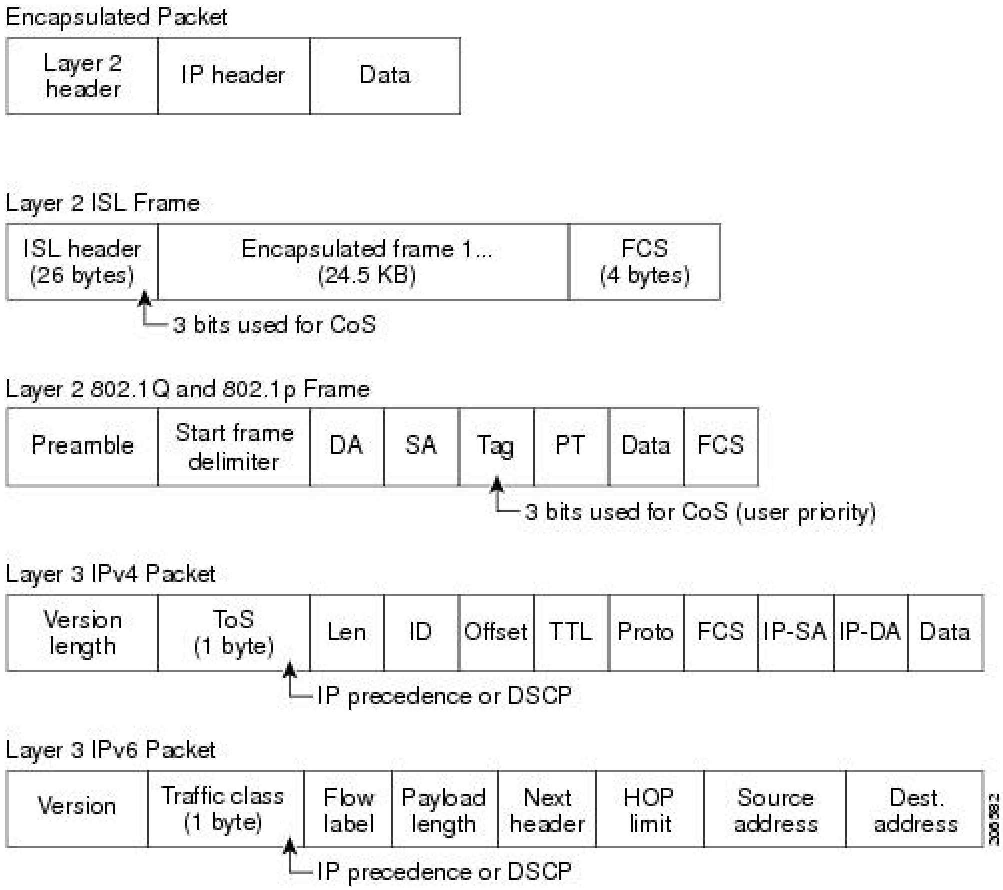
When you configure the QoS feature, you can select specific network traffic, prioritize it according to its relative importance, and use congestion-management and congestion-avoidance techniques to provide preferential treatment. Implementing QoS in your network makes network performance more predictable and bandwidth utilization more effective.

The QoS implementation is based on the Differentiated Services (Diff-Serv) architecture, a standard from the Internet Engineering Task Force (IETF). This architecture specifies that each packet is classified upon entry into the network.

The classification is carried in the IP packet header, using 6 bits from the deprecated IP type of service (ToS) field to carry the classification (*class*) information. Classification can also be carried in the Layer 2 frame.

The special bits in the Layer 2 frame or a Layer 3 packet are shown in the following figure:

Figure 3: QoS Classification Layers in Frames and Packets



Layer 2 Frame Prioritization Bits

Layer 2 Inter-Switch Link (ISL) frame headers have a 1-byte User field that carries an IEEE 802.1p class of service (CoS) value in the three least-significant bits. On ports configured as Layer 2 ISL trunks, all traffic is in ISL frames.

Layer 2 802.1Q frame headers have a 2-byte Tag Control Information field that carries the CoS value in the three most-significant bits, which are called the User Priority bits. On ports configured as Layer 2 802.1Q trunks, all traffic is in 802.1Q frames except for traffic in the native VLAN.

Other frame types cannot carry Layer 2 CoS values.

Layer 2 CoS values range from 0 for low priority to 7 for high priority.

Layer 3 Packet Prioritization Bits

Layer 3 IP packets can carry either an IP precedence value or a Differentiated Services Code Point (DSCP) value. QoS supports the use of either value because DSCP values are backward-compatible with IP precedence values.

IP precedence values range from 0 to 7. DSCP values range from 0 to 63.

End-to-End QoS Solution Using Classification

All switches and routers that access the Internet rely on the class information to provide the same forwarding treatment to packets with the same class information and different treatment to packets with different class information. The class information in the packet can be assigned by end hosts or by switches or routers along the way, based on a configured policy, detailed examination of the packet, or both. Detailed examination of the packet is expected to occur closer to the edge of the network, so that the core switches and routers are not overloaded with this task.

Switches and routers along the path can use the class information to limit the amount of resources allocated per traffic class. The behavior of an individual device when handling traffic in the Diff-Serv architecture is called per-hop behavior. If all devices along a path provide a consistent per-hop behavior, you can construct an end-to-end QoS solution.

Implementing QoS in your network can be a simple task or complex task and depends on the QoS features offered by your internetworking devices, the traffic types and patterns in your network, and the granularity of control that you need over incoming and outgoing traffic.

Packet Classification

Packet classification is the process of identifying a packet as belonging to one of several classes in a defined policy, based on certain criteria. The Modular QoS CLI (MQC) is a policy-class based language. The policy class language is used to define the following:

- Class-map template with one or several match criteria
- Policy-map template with one or several classes associated to the policy map

The policy map template is then associated to one or several interfaces on the switch.

Packet classification is the process of identifying a packet as belonging to one of the classes defined in the policy map. The process of classification will exit when the packet being processed matches a specific filter in a class. This is referred to as first-match exit. If a packet matches multiple classes in a policy, irrespective of the order of classes in the policy map, it would still exit the classification process after matching the first class.

If a packet does not match any of the classes in the policy, it would be classified into the default class in the policy. Every policy map has a default class, which is a system-defined class to match packets that do not match any of the user-defined classes.

Packet classification can be categorized into the following types:

- Classification based on information that is propagated with the packet
- Classification based on information that is switch specific
- Hierarchical classification

Classification Based on Information That is Propagated with the Packet

Classification that is based on information that is part of the packet and propagated either end-to-end or between hops, typically includes the following:

- Classification based on Layer 3 or 4 headers
- Classification based on Layer 2 information

Classification Based on Layer 3 or Layer 4 Header

This is the most common deployment scenario. Numerous fields in the Layer 3 and Layer 4 headers can be used for packet classification.

At the most granular level, this classification methodology can be used to match an entire flow. For this deployment type, an access control list (ACLs) can be used. ACLs can also be used to match based on various subsets of the flow (for example, source IP address only, or destination IP address only, or a combination of both).

Classification can also be done based on the precedence or DSCP values in the IP header. The IP precedence field is used to indicate the relative priority with which a particular packet needs to be handled. It is made up of three bits in the IP header's type of service (ToS) byte.

The following table shows the different IP precedence bit values and their names.

Note IP precedence is not supported for wireless QoS.

Table 35: IP Precedence Values and Names

IP Precedence Value	IP Precedence Bits	IP Precedence Names
0	000	Routine
1	001	Priority
2	010	Immediate
3	011	Flash
4	100	Flash Override
5	101	Critical
6	110	Internetwork control
7	111	Network control

Note All routing control traffic in the network uses IP precedence value 6 by default. IP precedence value 7 also is reserved for network control traffic. Therefore, the use of IP precedence values 6 and 7 is not recommended for user traffic.

The DSCP field is made up of 6 bits in the IP header and is being standardized by the Internet Engineering Task Force (IETF) Differentiated Services Working Group. The original ToS byte contained the DSCP bits has been renamed the DSCP byte. The DSCP field is part of the IP header, similar to IP precedence. The DSCP field is a super set of the IP precedence field. Therefore, the DSCP field is used and is set in ways similar to what was described with respect to IP precedence.

Note The DSCP field definition is backward-compatible with the IP precedence values.

Classification Based on Layer 2 Header

A variety of methods can be used to perform classification based on the Layer 2 header information. The most common methods include the following:

- MAC address-based classification (only for access groups)—Classification is based upon the source MAC address (for policies in the input direction) and destination MAC address (for policies in the output direction).
- Class-of-Service—Classification is based on the 3 bits in the Layer 2 header based on the IEEE 802.1p standard. This usually maps to the ToS byte in the IP header.
- VLAN ID—Classification is based on the VLAN ID of the packet.

Note Some of these fields in the Layer 2 header can also be set using a policy.

Classification Based on Information that is Device Specific (QoS Groups)

The switch also provides classification mechanisms that are available where classification is not based on information in the packet header or payload.

At times you might be required to aggregate traffic coming from multiple input interfaces into a specific class in the output interface. For example, multiple customer edge routers might be going into the same access switch on different interfaces. The service provider might want to police all the aggregate voice traffic going into the core to a specific rate. However, the voice traffic coming in from the different customers could have a different ToS settings. QoS group-based classification is a feature that is useful in these scenarios.

Policies configured on the input interfaces set the QoS group to a specific value, which can then be used to classify packets in the policy enabled on output interface.

The QoS group is a field in the packet data structure internal to the switch. It is important to note that a QoS group is an internal label to the switch and is not part of the packet header.

Hierarchical Classification

The switch permits you to perform a classification based on other classes. Typically, this action may be required when there is a need to combine the classification mechanisms (that is, filters) from two or more classes into a single class map.

QoS Wired Model

To implement QoS, the switch must perform the following tasks:

- Traffic classification—Distinguishes packets or flows from one another.
- Traffic marking and policing—Assigns a label to indicate the given quality of service as the packets move through the switch, and then make the packets comply with the configured resource usage limits.
- Queuing and scheduling—Provides different treatment in all situations where resource contention exists.
- Shaping—Ensures that traffic sent from the switch meets a specific traffic profile.

Ingress Port Activity

The following activities occur at the ingress port of the switch:

- Classification—Classifying a distinct path for a packet by associating it with a QoS label. For example, the switch maps the CoS or DSCP in the packet to a QoS label to distinguish one type of traffic from another. The QoS label that is generated identifies all future QoS actions to be performed on this packet.
- Policing—Policing determines whether a packet is in or out of profile by comparing the rate of the incoming traffic to the configured policer. The policer limits the bandwidth consumed by a flow of traffic. The result is passed to the marker.
- Marking—Marking evaluates the policer and configuration information for the action to be taken when a packet is out of profile and determines what to do with the packet (pass through a packet without modification, mark down the QoS label in the packet, or drop the packet).

Note Applying polices on the wireless ingress port is not supported on the switch.

Egress Port Activity

The following activities occur at the egress port of the switch:

- Policing—Policing determines whether a packet is in or out of profile by comparing the rate of the incoming traffic to the configured policer. The policer limits the bandwidth consumed by a flow of traffic. The result is passed to the marker.
- Marking—Marking evaluates the policer and configuration information for the action to be taken when a packet is out of profile and determines what to do with the packet (pass through a packet without modification, mark down the QoS label in the packet, or drop the packet).
- Queueing—Queueing evaluates the QoS packet label and the corresponding DSCP or CoS value before selecting which of the egress queues to use. Because congestion can occur when multiple ingress ports simultaneously send data to an egress port, Weighted Tail Drop (WTD) differentiates traffic classes and subjects the packets to different thresholds based on the QoS label. If the threshold is exceeded, the packet is dropped.

Classification

Classification is the process of distinguishing one kind of traffic from another by examining the fields in the packet. Classification is enabled only if QoS is enabled on the switch. By default, QoS is enabled on the switch.

During classification, the switch performs a lookup and assigns a QoS label to the packet. The QoS label identifies all QoS actions to be performed on the packet and from which queue the packet is sent.

Access Control Lists

You can use IP standard, IP extended, or Layer 2 MAC ACLs to define a group of packets with the same characteristics (class). You can also classify IP traffic based on IPv6 ACLs.

In the QoS context, the permit and deny actions in the access control entries (ACEs) have different meanings from security ACLs:

- If a match with a permit action is encountered (first-match principle), the specified QoS-related action is taken.
- If a match with a deny action is encountered, the ACL being processed is skipped, and the next ACL is processed.
- If no match with a permit action is encountered and all the ACEs have been examined, no QoS processing occurs on the packet, and the switch offers best-effort service to the packet.
- If multiple ACLs are configured on a port, the lookup stops after the packet matches the first ACL with a permit action, and QoS processing begins.

Note When creating an access list, note that by default the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.

After a traffic class has been defined with the ACL, you can attach a policy to it. A policy might contain multiple classes with actions specified for each one of them. A policy might include commands to classify the class as a particular aggregate (for example, assign a DSCP) or rate-limit the class. This policy is then attached to a particular port on which it becomes effective.

You implement IP ACLs to classify IP traffic by using the **access-list** global configuration command; you implement Layer 2 MAC ACLs to classify non-IP traffic by using the **mac access-list extended** global configuration command.

Class Maps

A class map is a mechanism that you use to name a specific traffic flow (or class) and isolate it from all other traffic. The class map defines the criteria used to match against a specific traffic flow to further classify it. The criteria can include matching the access group defined by the ACL or matching a specific list of DSCP or IP precedence values. If you have more than one type of traffic that you want to classify, you can create another class map and use a different name. After a packet is matched against the class-map criteria, you further classify it through the use of a policy map.

You create a class map by using the **class-map** global configuration command or the **class** policy-map configuration command. You should use the **class-map** command when the map is shared among many ports. When you enter the **class-map** command, the switch enters the class-map configuration mode. In this mode, you define the match criterion for the traffic by using the **match** class-map configuration command.

You can create a default class by using the **class class-default** policy-map configuration command. The default class is system-defined and cannot be configured. Unclassified traffic (traffic that does not meet the match criteria specified in the traffic classes) is treated as default traffic.

Related Topics

Creating a Traffic Class , on page 265

Examples: Classification by Access Control Lists, on page 313

Policy Maps

A policy map specifies which traffic class to act on. Actions can include the following:

- Setting a specific DSCP or IP precedence value in the traffic class

- Setting a CoS value in the traffic class
- Setting a QoS group
- Setting a wireless LAN (WLAN) value in the traffic class
- Specifying the traffic bandwidth limitations and the action to take when the traffic is out of profile

Before a policy map can be effective, you must attach it to a port.

You create and name a policy map using the **policy-map** global configuration command. When you enter this command, the switch enters the policy-map configuration mode. In this mode, you specify the actions to take on a specific traffic class by using the **class** or **set** policy-map configuration and policy-map class configuration commands.

The policy map can also be configured using the **police** and **bandwidth** policy-map class configuration commands, which define the policer, the bandwidth limitations of the traffic, and the action to take if the limits are exceeded. In addition, the policy-map can further be configured using the **priority** policy-map class configuration command, to schedule priority for the class or the queueing policy-map class configuration commands, **queue-buffers** and **queue-limit**.

To enable the policy map, you attach it to a port by using the **service-policy** interface configuration command.

Related Topics

Creating a Traffic Policy , on page 268

Policy Map on Physical Port

You can configure a nonhierarchical policy map on a physical port that specifies which traffic class to act on. Actions can include setting a specific DSCP or IP precedence value in the traffic class, specifying the traffic bandwidth limitations for each matched traffic class (policer), and taking action when the traffic is out of profile (marking).

A policy map also has these characteristics:

- A policy map can contain multiple class statements, each with different match criteria and policers.
- A policy map can contain a predefined default traffic class explicitly placed at the end of the map.

  When you configure a default traffic class by using the **class class-default** policy-map configuration command, unclassified traffic (traffic that does not meet the match criteria specified in the traffic classes) is treated as the default traffic class (**class-default**).
- A separate policy-map class can exist for each type of traffic received through a port.

Related Topics

Attaching a Traffic Policy to an Interface , on page 277

Policy Map on VLANs

The switch supports a VLAN QoS feature that allows the user to perform QoS treatment at the VLAN level (classification and QoS actions) using the incoming frame's VLAN information. In VLAN-based QoS, a service policy is applied to an SVI interface. All physical interfaces belonging to a VLAN policy map then need to be programmed to refer to the VLAN-based policy maps instead of the port-based policy map.

Although the policy map is applied to the VLAN SVI, any policing (rate-limiting) action can only be performed on a per-port basis. You cannot configure the policer to take account of the sum of traffic from a number of physical ports. Each port needs to have a separate policer governing the traffic coming into that port.

Related Topics

Classifying, Policing, and Marking Traffic on SVIs by Using Policy Maps , on page 282
Examples: Policer VLAN Configuration, on page 319

Policing

After a packet is classified and has a DSCP-based, CoS-based, or QoS-group label assigned to it, the policing and marking process can begin.

Policing involves creating a policer that specifies the bandwidth limits for the traffic. Packets that exceed the limits are *out of profile* or *nonconforming*. Each policer decides on a packet-by-packet basis whether the packet is in or out of profile and specifies the actions on the packet. These actions, carried out by the marker, include passing through the packet without modification, dropping the packet, or modifying (marking down) the assigned DSCP or CoS value of the packet and allowing the packet to pass through.

To avoid out-of-order packets, both conform and nonconforming traffic typically exit the same queue.

Note All traffic, regardless of whether it is bridged or routed, is subjected to a policer, if one is configured. As a result, bridged packets might be dropped or might have their DSCP or CoS fields modified when they are policed and marked.

You can only configure policing on a physical port.

After you configure the policy map and policing actions, attach the policy to an ingress port or SVI by using the **service-policy** interface configuration command.

Related Topics

Configuring Police , on page 299
Examples: Policing Action Configuration, on page 318

Token-Bucket Algorithm

Policing uses a token-bucket algorithm. As each frame is received by the switch, a token is added to the bucket. The bucket has a hole in it and leaks at a rate that you specify as the average traffic rate in bits per second. Each time a token is added to the bucket, the switch verifies that there is enough room in the bucket. If there is not enough room, the packet is marked as nonconforming, and the specified policer action is taken (dropped or marked down).

How quickly the bucket fills is a function of the bucket depth (burst-byte), the rate at which the tokens are removed (rate-bps), and the duration of the burst above the average rate. The size of the bucket imposes an upper limit on the burst length and limits the number of frames that can be transmitted back-to-back. If the burst is short, the bucket does not overflow, and no action is taken against the traffic flow. However, if a burst is long and at a higher rate, the bucket overflows, and the policing actions are taken against the frames in that burst.

You configure the bucket depth (the maximum burst that is tolerated before the bucket overflows) by using the burst-byte option of the **police** policy-map class configuration command. You configure how fast (the average rate) that the tokens are removed from the bucket by using the rate option of the **police** policy-map class configuration command.

Related Topics

Configuring Police , on page 299

Examples: Policing Units, on page 320

Marking

Marking is used to convey specific information to a downstream device in the network, or to carry information from one interface in a switch to another.

Marking can be used to set certain field/bits in the packet headers, or marking can also be used to set certain fields in the packet structure that is internal to the switch. Additionally, the marking feature can be used to define mapping between fields. The following marking methods are available for QoS:

- Packet header
- Device (switch) specific information
- Table maps

Packet Header Marking

Marking on fields in the packet header can be classified into two general categories:

- IPv4/v6 header bit marking
- Layer 2 header bit marking

The marking feature at the IP level is used to set the precedence or the DSCP in the IP header to a specific value to get a specific per-hop behavior at the downstream device (switch or router), or it can also be used to aggregate traffic from different input interfaces into a single class in the output interface. The functionality is currently supported on both the IPv4 and IPv6 headers.

Marking in the Layer 2 headers is typically used to influence dropping behavior in the downstream devices (switch or router). It works in tandem with the match on the Layer 2 headers. The bits in the Layer 2 header that can be set using a policy map are class of service.

Switch Specific Information Marking

This form of marking includes marking of fields in the packet data structure that are not part of the packets header, so that the marking can be used later in the data path. This is not propagated between the switches. Marking of QoS-group falls into this category. This form of marking is only supported in policies that are enabled on the input interfaces. The corresponding matching mechanism can be enabled on the output interfaces on the same switch and an appropriate QoS action can be applied.

Table Map Marking

Table map marking enables the mapping and conversion from one field to another using a conversion table. This conversion table is called a table map.

Depending upon the table map attached to an interface, CoS, DSCP, and UP values (UP specific to wireless packets) of the packet are rewritten. The switch allows configuring both ingress table map policies and egress table map policies.

Note The switch stack supports a total of 14 table maps. Only one table map is supported per wired port, per direction.

As an example, a table map can be used to map the Layer 2 CoS setting to a precedence value in Layer 3. This feature enables combining multiple **set** commands into a single table, which indicates the method to perform the mapping. This table can be referenced in multiple policies, or multiple times in the same policy.

The following table shows the currently supported forms of mapping:

Table 36: Packet-Marking Types Used for Establishing a To-From Relationship

The To Packet-Marking Type	The From Packet-Marking Type
Precedence	CoS
Precedence	QoS Group
DSCP	CoS
DSCP	QoS Group
CoS	Precedence
CoS	DSCP
QoS Group	Precedence
QoS Group	DSCP

A table map-based policy supports the following capabilities:

- Mutation—You can have a table map that maps from one DSCP value set to another DSCP value set, and this can be attached to an egress port.
- Rewrite—Packets coming in are rewritten depending upon the configured table map.
- Mapping—Table map based policies can be used instead of set policies.

The following steps are required for table map marking:

1. Define the table map—Use the **table-map** global configuration command to map the values. The table does not know of the policies or classes within which it will be used. The default command in the table map is used to indicate the value to be copied into the to field when there is no matching from field.
2. Define the policy map—You must define the policy map where the table map will be used.
3. Associate the policy to an interface.

Note A table map policy on an input port changes the trust setting of that port to the from type of qos-marking.

Related Topics

Configuring Table Maps , on page 285

Examples: Table Map Marking Configuration, on page 321

Traffic Conditioning

To support QoS in a network, traffic entering the service provider network needs to be policed on the network boundary routers to ensure that the traffic rate stays within the service limit. Even if a few routers at the network boundary start sending more traffic than what the network core is provisioned to handle, the increased traffic load leads to network congestion. The degraded performance in the network makes it difficult to deliver QoS for all the network traffic.

Traffic policing functions (using the police feature) and shaping functions (using the traffic shaping feature) manage the traffic rate, but differ in how they treat traffic when tokens are exhausted. The concept of tokens comes from the token bucket scheme, a traffic metering function.

Note When running QoS tests on network traffic, you may see different results for the shaper and policing data. Network traffic data from shaping provides more accurate results.

This table compares the policing and shaping functions.

Table 37: Comparison Between Policing and Shaping Functions

Policing Function	Shaping Function
Sends conforming traffic up to the line rate and allows bursts.	Smooths traffic and sends it out at a constant rate.
When tokens are exhausted, action is taken immediately.	When tokens are exhausted, it buffers packets and sends them out later, when tokens are available. A class with shaping has a queue associated with it which will be used to buffer the packets.
Policing has multiple units of configuration – in bits per second, packets per second and cells per second.	Shaping has only one unit of configuration - in bits per second.
Policing has multiple possible actions associated with an event, marking and dropping being example of such actions.	Shaping does not have the provision to mark packets that do not meet the profile.
Works for both input and output traffic.	Implemented for output traffic only.

Policing Function	Shaping Function
Transmission Control Protocol (TCP) detects the line at line speed but adapts to the configured rate when a packet drop occurs by lowering its window size.	TCP can detect that it has a lower speed line and adapt its retransmission timer accordingly. This results in less scope of retransmissions and is TCP-friendly.

Policing

The QoS policing feature is used to impose a maximum rate on a traffic class. The QoS policing feature can also be used with the priority feature to restrict priority traffic. If the rate is exceeded, then a specific action is taken as soon as the event occurs. The rate (committed information rate [CIR] and peak information rate [PIR]) and the burst parameters (conformed burst size [B_c] and extended burst size [B_e]) are all configured in bytes per second.

The following policing forms or policers are supported for QoS:

- Single-rate two-color policing
- Dual-rate three-color policing

Note Single-rate three-color policing is not supported.

Single-Rate Two-Color Policing

Single-rate two-color policer is the mode in which you configure only a CIR and a B_c.

The B_c is an optional parameter, and if it is not specified it is computed by default. In this mode, when an incoming packet has enough tokens available, the packet is considered to be conforming. If at the time of packet arrival, enough tokens are not available within the bounds of B_c, the packet is considered to have exceeded the configured rate.

Note For information about the token-bucket algorithm, see Token-Bucket Algorithm, on page 252.

Related Topics

Configuring Police , on page 299
Examples: Single-Rate Two-Color Policing Configuration, on page 320

Dual-Rate Three-Color Policing

With the dual rate policer, the switch supports only color-blind mode. In this mode, you configure a committed information rate (CIR) and a peak information rate (PIR). As the name suggests, there are two token buckets in this case, one for the peak rate, and one for the conformed rate.

Note For information about the token-bucket algorithm, see Token-Bucket Algorithm, on page 252.

In the color-blind mode, the incoming packet is first checked against the peak rate bucket. If there are not enough tokens available, the packet is said to violate the rate. If there are enough tokens available, then the tokens in the conformed rate buckets are checked to determine if there are enough tokens available. The tokens in the peak rate bucket are decremented by the size of the packet. If it does not have enough tokens available, the packet is said to have exceeded the configured rate. If there are enough tokens available, then the packet is said to conform, and the tokens in both the buckets are decremented by the size of the packet.

The rate at which tokens are replenished depends on the packet arrival. Assume that a packet comes in at time T1 and the next one comes in at time T2. The time interval between T1 and T2 determines the number of tokens that need to be added to the token bucket. This is calculated as:

Time interval between packets (T2-T1) * CIR)/8 bytes

Related Topics

Configuring Police , on page 299

Examples: Dual-Rate Three-Color Policing Configuration, on page 321

Shaping

Shaping is the process of imposing a maximum rate of traffic, while regulating the traffic rate in such a way that the downstream switches and routers are not subjected to congestion. Shaping in the most common form is used to limit the traffic sent from a physical or logical interface.

Shaping has a buffer associated with it that ensures that packets which do not have enough tokens are buffered as opposed to being immediately dropped. The number of buffers available to the subset of traffic being shaped is limited and is computed based on a variety of factors. The number of buffers available can also be tuned using specific QoS commands. Packets are buffered as buffers are available, beyond which they are dropped.

Class-Based Traffic Shaping

The switch uses class-based traffic shaping. This shaping feature is enabled on a class in a policy that is associated to an interface. A class that has shaping configured is allocated a number of buffers to hold the packets that do not have tokens. The buffered packets are sent out from the class using FIFO. In the most common form of usage, class-based shaping is used to impose a maximum rate for an physical interface or logical interface as a whole. The following shaping forms are supported in a class:

- Average rate shaping
- Hierarchical shaping

Shaping is implemented using a token bucket. The values of CIR, B_c and B_e determine the rate at which the packets are sent out and the rate at which the tokens are replenished.

Note For information about the token-bucket algorithm, see Token-Bucket Algorithm, on page 252.

Average Rate Shaping

You use the **shape average** policy-map class command to configure average rate shaping.

This command configures a maximum bandwidth for a particular class. The queue bandwidth is restricted to this value even though the port has more bandwidth available. The switch supports configuring shape average by either a percentage or by a target bit rate value.

Related Topics

Configuring Shaping , on page 309

Examples: Average Rate Shaping Configuration, on page 317

Hierarchical Shaping

Shaping can also be configured at multiple levels in a hierarchy. This is accomplished by creating a parent policy with shaping configured, and then attaching child policies with additional shaping configurations to the parent policy.

There are two supported types of hierarchical shaping:

- Port shaper
- User-configured shaping

The port shaper uses the class default and the only action permitted in the parent is shaping. The queueing action is in the child with the port shaper. With the user configured shaping, you cannot have queueing action in the child.

Related Topics

Configuring Shaping , on page 309

Queueing and Scheduling

The switch uses both queueing and scheduling to help prevent traffic congestion. The switch supports the following queueing and scheduling features:

- Bandwidth
- Weighted Tail Drop
- Priority queues
- Queue buffers

Bandwidth

The switch supports the following bandwidth configurations:

- Bandwidth percent
- Bandwidth remaining ratio

Related Topics

Configuring Bandwidth , on page 297

Bandwidth Percent

You can use the **bandwidth percent** policy-map class command to allocate a minimum bandwidth to a particular class. The total sum cannot exceed 100 percent and in case the total sum is less than 100 percent, then the rest of the bandwidth is divided equally among all bandwidth queues.

Note A queue can oversubscribe bandwidth in case the other queues do not utilize the entire port bandwidth.

You cannot mix bandwidth types on a policy map. For example, you cannot configure bandwidth in a single policy map using both a bandwidth percent and in kilobits per second.

Bandwidth Remaining Ratio

You use the **bandwidth remaining ratio** policy-map class command to create a ratio for sharing unused bandwidth in specified queues. Any unused bandwidth will be used by these specific queues in the ratio that is specified by the configuration. Use this command when the **priority** command is also used for certain queues in the policy.

When you assign ratios, the queues will be assigned certain weights which are inline with these ratios.

You can specify ratios using a range from 0 to 100. For example, you can configure a bandwidth remaining ration of 2 on one class, and another queue with a bandwidth remaining ratio of 4 on another class. The bandwidth remaining ratio of 4 will be scheduled twice as often as the bandwidth remaining ratio of 2.

The total bandwidth ratio allocation for the policy can exceed 100. For example, you can configure a queue with a bandwidth remaining ratio of 50, and another queue with a bandwidth remaining ratio of 100.

Weighted Tail Drop

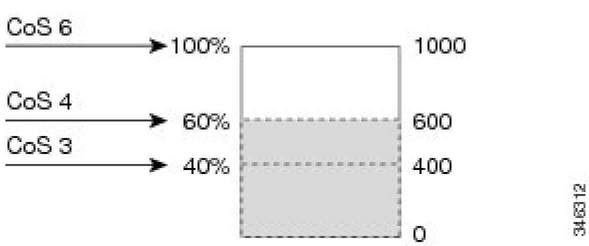
The switch egress queues use an enhanced version of the tail-drop congestion-avoidance mechanism called weighted tail drop (WTD). WTD is implemented on queues to manage the queue lengths and to provide drop precedences for different traffic classifications.

As a frame is enqueued to a particular queue, WTD uses the frame's assigned QoS label to subject it to different thresholds. If the threshold is exceeded for that QoS label (the space available in the destination queue is less than the size of the frame), the switch drops the frame.

Each queue has three configurable threshold values. The QoS label determines which of the three threshold values is subjected to the frame.

The following figure shows an example of WTD operating on a queue whose size is 1000 frames. Three drop percentages are configured: 40 percent (400 frames), 60 percent (600 frames), and 100 percent (1000 frames). These percentages indicate that up to 400 frames can be queued at the 40-percent threshold, up to 600 frames at the 60-percent threshold, and up to 1000 frames at the 100-percent threshold.

Figure 4: WTD and Queue Operation



In the example, CoS value 6 has a greater importance than the other CoS values, and is assigned to the 100-percent drop threshold (queue-full state). CoS values 4 is assigned to the 60-percent threshold, and CoS values 3 is assigned to the 40-percent threshold. All of these threshold values are assigned using the **queue-limit cos** command.

Assuming the queue is already filled with 600 frames, and a new frame arrives. It contains CoS value 4 and is subjected to the 60-percent threshold. If this frame is added to the queue, the threshold will be exceeded, so the switch drops it.

Related Topics

Configuring Queue Limits , on page 306

Examples: Queue-limit Configuration, on page 317

Weighted Tail Drop Default Values

The following are the Weighted Tail Drop (WTD) default values and the rules for configuring WTD threshold values.

- If you configure less than three queue-limit percentages for WTD, then WTD default values are assigned to these thresholds.

  The following are the WTD threshold default values:

  Table 38: WTD Threshold Default Values

Threshold	Default Value Percentage
0	80
1	90
2	400

- If 3 different WTD thresholds are configured, then the queues are programmed as configured.
- If 2 WTD thresholds are configured, then the maximum value percentage will be 400.
- If a WTD single threshold is configured as x, then the maximum value percentage will be 400.
  - If the value of x is less than 90, then threshold1=90 and threshold 0= x.
  - If the value of x equals 90, then threshold1=90, threshold 0=80.
  - If the value x is greater than 90, then threshold1=x, threshold 0=80.

Priority Queues

Each port supports eight egress queues, of which two can be given a priority.

You use the **priority level** policy class-map command to configure the priority for two classes. One of the classes has to be configured with a priority queue level 1, and the other class has to be configured with a priority queue level 2. Packets on these two queues are subjected to less latency with respect to other queues.

Related Topics

Configuring Priority, on page 302

Queue Buffer

Each 1-gigabit port on the switch is allocated 168 buffers for a wireless port and 300 buffers for a wired port. Each 10-gigabit port is allocated 1800 buffers. At boot time, when there is no policy map enabled on the wired port, there are two queues created by default. Wired ports can have a maximum of 8 queues configured using MQC-based policies. The following table shows which packets go into which one of the queues:

Table 39: DSCP, Precedence, and CoS - Queue Threshold Mapping Table

DSCP, Precedence or CoS	Queue	Threshold
Control Packets	0	2
Rest of Packets	1	2

Note You can guarantee the availability of buffers, set drop thresholds, and configure the maximum memory allocation for a queue. You use the **queue-buffers** policy-map class command to configure the queue buffers. You use the **queue-limit** policy-map class command to configure the maximum thresholds.

There are two types of buffer allocations: hard buffers, which are explicitly reserved for the queue, and soft buffers, which are available for other ports when unused by a given port.

For the wireless port default, Queue 0 will be given 40 percent of the buffers that are available for the interface as hard buffers, that is 67 buffers are allocated for Queue 0 in the context of 1-gigabit ports. The soft maximum for this queue is set to 268 (calculated as 67 * 400/100) for 1-gigabit ports, where 400 is the default maximum threshold that is configured for any queue.

For the wired port default, Queue 0 will be given 40 percent of the buffers that are available for the interface as hard buffers, that is 120 buffers are allocated for Queue 0 in the context of 1-gigabit ports, and 720 buffers in the context of 10-gigabit ports. The soft maximum for this queue is set to 480 (calculated as 120 * 400/100) for 1-gigabit ports and 2880 for 10-gigabit ports, where 400 is the default maximum threshold that is configured for any queue.

Queue 1 does not have any hard buffers allocated. The default soft buffer limit is set to 400 (which is the maximum threshold). The threshold would determine the maximum number of soft buffers that can be borrowed from the common pool.

Queue Buffer Allocation

The buffer allocation to any queue can be tuned using the **queue-buffers ratio** policy-map class configuration command.

Related Topics

Configuring Queue Buffers , on page 304

Examples: Queue Buffers Configuration, on page 318

Dynamic Threshold and Scaling

Traditionally, reserved buffers are statically allocated for each queue. No matter whether the queue is active or not, its buffers are held up by the queue. In addition, as the number of queues increases, the portion of the reserved buffers allocated for each queue can become smaller and smaller. Eventually, a situation may occur where there are not enough reserved buffers to support a jumbo frame for all queues.

The switch supports Dynamic Thresholding and Scaling (DTS), which is a feature that provides a fair and efficient allocation of buffer resources. When congestion occurs, this DTS mechanism provides an elastic buffer allocation for the incoming data based on the occupancy of the global/port resources. Conceptually, DTS scales down the queue buffer allocation gradually as the resources are used up to leave room for other queues, and vice versa. This flexible method allows the buffers to be more efficiently and fairly utilized.

As mentioned in the previous sections, there are two limits configured on a queue—a hard limit and a soft limit.

Hard limits are not part of DTS. These buffers are available only for that queue. The sum of the hard limits should be less than the globally set up hard maximum limit. The global hard limit configured for egress queuing is currently set to 5705. In the default scenario when there are no MQC policies configured, the 24 1-gigabit ports would take up 24 * 67 = 1608, and the 4 10-gigabit ports would take up 4 * 720 = 2880, for a total of 4488 buffers, allowing room for more hard buffers to be allocated based upon the configuration.

Soft limit buffers participate in the DTS process. Additionally, some of the soft buffer allocations can exceed the global soft limit allocation. The global soft limit allocation for egress queuing is currently set to 7607. The sum of the hard and soft limits add up to 13312, which in turn translates to 3.4 MB. Because the sum of the soft buffer allocations can exceed the global limit, it allows a specific queue to use a large number of buffers when the system is lightly loaded. The DTS process dynamically adjusts the per-queue allocation as the system becomes more heavily loaded.

Trust Behavior

Trust Behavior for Wired Ports

For wired ports that are connected to the switch (end points such as IP phones, laptops, cameras, telepresence units, or other devices), their DSCP, precedence, or CoS values coming in from these end points are trusted by the switch and therefore are retained in the absence of any explicit policy configuration.

The packets are enqueued to the appropriate queue per the default initial configuration.

In scenarios where the incoming packet type differs from the outgoing packet type, the trust behavior and the queuing behavior are explained in the following table. Note that the default trust mode for a wired port is DSCP based. The trust mode 'falls back' to CoS if the incoming packet is a pure Layer 2 packet. You can also change the trust setting from DSCP to CoS. This is accomplished by using an MQC policy that has a class default with a 'set cos cos table default default-cos' action, where default-cos is the name of the table map created (which only performs a default copy).

Table 40: Trust and Queueing Behavior

Incoming Packet	Outgoing Packet	Trust Behavior	Queuing Behavior
Layer 3	Layer 3	Preserve DSCP/Precedence	Based on DSCP
Layer 2	Layer 2	Not applicable	Based on CoS
Tagged	Tagged	Preserve DSCP and CoS	Based on DSCP (trust DSCP takes precedence)

Incoming Packet	Outgoing Packet	Trust Behavior	Queuing Behavior
Layer 3	Tagged	Preserve DSCP, CoS is set to 0	Based on DSCP

Port Security on a Trusted Boundary for Cisco IP Phones

In a typical network, you connect a Cisco IP Phone to a switch port and cascade devices that generate data packets from the back of the telephone. The Cisco IP Phone guarantees the voice quality through a shared data link by marking the CoS level of the voice packets as high priority (CoS = 5) and by marking the data packets as low priority (CoS = 0). Traffic sent from the telephone to the switch is typically marked with a tag that uses the 802.1Q header. The header contains the VLAN information and the class of service (CoS) 3-bit field, which is the priority of the packet.

For most Cisco IP Phone configurations, the traffic sent from the telephone to the switch should be trusted to ensure that voice traffic is properly prioritized over other types of traffic in the network. By using the **trust device** interface configuration command, you configure the switch port to which the telephone is connected to trust the traffic received on that port.

Note The **trust device** *device_type* interface configuration command is only supported in an auto-QoS configuration, and not as a stand-alone command on the switch. When using the **trust device** *device_type* interface configuration command in an auto-QoS configuration, if the connected peer device is not a corresponding device (defined as a device matching your trust policy), both CoS and DSCP values are set to "0" and any input policy will not take effect.

With the trusted setting, you also can use the trusted boundary feature to prevent misuse of a high-priority queue if a user bypasses the telephone and connects the PC directly to the switch. Without trusted boundary, the CoS labels generated by the PC are trusted by the switch (because of the trusted CoS setting). By contrast, trusted boundary uses CDP to detect the presence of a Cisco IP Phone (such as the Cisco IP Phone 7910, 7935, 7940, and 7960) on a switch port. If the telephone is not detected, the trusted boundary feature disables the trusted setting on the switch port and prevents misuse of a high-priority queue. Note that the trusted boundary feature is not effective if the PC and Cisco IP Phone are connected to a hub that is connected to the switch.

Related Topics

Configuring Trust Behavior for the Device Type, on page 288

Standard QoS Default Settings

Default Wired QoS Configuration

There are two queues configured by default on each wired interface on the switch. All control traffic traverses and is processed through queue 0. All other traffic traverses and is processed through queue 1.

DSCP Maps

Default CoS-to-DSCP Map

You use the CoS-to-DSCP map to map CoS values in incoming packets to a DSCP value that QoS uses internally to represent the priority of the traffic. The following table shows the default CoS-to-DSCP map. If these values are not appropriate for your network, you need to modify them.

Table 41: Default CoS-to-DSCP Map

CoS Value	DSCP Value
0	0
1	8
2	16
3	24
4	32
5	40
6	48
7	56

Default IP-Precedence-to-DSCP Map

You use the IP-precedence-to-DSCP map to map IP precedence values in incoming packets to a DSCP value that QoS uses internally to represent the priority of the traffic. The following table shows the default IP-precedence-to-DSCP map. If these values are not appropriate for your network, you need to modify them.

Table 42: Default IP-Precedence-to-DSCP Map

IP Precedence Value	DSCP Value
0	0
1	8
2	16
3	24
4	32
5	40
6	48
7	56

Default DSCP-to-CoS Map

You use the DSCP-to-CoS map to generate a CoS value, which is used to select one of the four egress queues. The following table shows the default DSCP-to-CoS map. If these values are not appropriate for your network, you need to modify them.

Table 43: Default DSCP-to-CoS Map

DSCP Value	CoS Value
0–7	0
8–15	1
16–23	2
24–31	3
32–39	4
40–47	5
48–55	6
56–63	7

How to Configure QoS

Configuring Class, Policy, and Table Maps

Creating a Traffic Class

To create a traffic class containing match criteria, use the **class-map** command to specify the traffic class name, and then use the following **match** commands in class-map configuration mode, as needed.

Before You Begin

All match commands specified in this configuration task are considered optional, but you must configure at least one match criterion for a class.

SUMMARY STEPS

1. **configure terminal**
2. **class-map** {*class-map name* | **match-any**}
3. **match access-group** {*index number* | *name*}
4. **match class-map** *class-map name*
5. **match cos** *cos value*
6. **match dscp** *dscp value*
7. **match ip** {**dscp** *dscp value* | **precedence** *precedence value* }
8. **match non-client-nrt**
9. **match qos-group** *qos group value*
10. **match vlan** *vlan value*
11. **match wlan user-priority** *wlan value*
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**class-map** {*class-map name* | **match-any**}
	
	Example:
	`Switch(config)# class-map test_1000`
	`Switch(config-cmap)#`
		Enters class map configuration mode.
		• Creates a class map to be used for matching packets to the class whose name you specify.
		• If you specify **match-any**, one of the match criteria must be met for traffic entering the traffic class to be classified as part of the traffic class. This is the default.

Step 3	**match access-group** {*index number* | *name*}
	
	Example:
	`Switch(config-cmap)# match access-group 100`
	`Switch(config-cmap)#`
		The following parameters are available for this command:
		• access-group
		• class-map
		• cos
		• dscp
		• ip
		• non-client-nrt
		• precedence
		• qos-group

	Command or Action	Purpose
		• vlan • wlan user priority (Optional) For this example, enter the access-group ID: • Access list index (value from 1 to 2799) • Named access list
Step 4	**match class-map** *class-map name* **Example:** `Switch(config-cmap)#` **`match class-map test_2000`** `Switch(config-cmap)#`	(Optional) Matches to another class-map name.
Step 5	**match cos** *cos value* **Example:** `Switch(config-cmap)#` **`match cos 2 3 4 5`** `Switch(config-cmap)#`	(Optional) Matches IEEE 802.1Q or ISL class of service (user) priority values. • Enters up to 4 CoS values separated by spaces (0 to 7).
Step 6	**match dscp** *dscp value* **Example:** `Switch(config-cmap)#` **`match dscp af11 af12`** `Switch(config-cmap)#`	(Optional) Matches the DSCP values in IPv4 and IPv6 packets.
Step 7	**match ip** {**dscp** *dscp value* | **precedence** *precedence value* } **Example:** `Switch(config-cmap)#` **`match ip dscp af11 af12`** `Switch(config-cmap)#`	(Optional) Matches IP values including the following: • **dscp—**Matches IP DSCP (DiffServ codepoints). • **precedence—**Matches IP precedence (0 to 7).
Step 8	**match non-client-nrt** **Example:** `Switch(config-cmap)#` **`match non-client-nrt`** `Switch(config-cmap)#`	(Optional) Matches non-client NRT (Non-Real-Time). **Note** This match is applicable only for policies on a wireless port. It carries all the multi-destination and AP (non-client) bound traffic.
Step 9	**match qos-group** *qos group value* **Example:** `Switch(config-cmap)#` **`match qos-group 10`**	(Optional) Matches QoS group value (from 0 to 31).

	Command or Action	Purpose
	`Switch(config-cmap)#`	
Step 10	**match vlan** *vlan value* **Example:** `Switch(config-cmap)#` **`match vlan 210`** `Switch(config-cmap)#`	(Optional) Matches a VLAN ID (from 1 to 4095).
Step 11	**match wlan user-priority** *wlan value* **Example:** `Switch(config-cmap)#` **`match wlan user priority 7`** `Switch(config-cmap)#`	(Optional) Matches 802.11e specific values. Enter the user priority 802.11e user priority (0 to 7).
Step 12	**end** **Example:** `Switch(config-cmap)#` **`end`**	Saves the configuration changes.

What to Do Next

Configure the policy map.

Related Topics

Class Maps, on page 250

Examples: Classification by Access Control Lists, on page 313

Creating a Traffic Policy

To create a traffic policy, use the **policy-map** global configuration command to specify the traffic policy name.

The traffic class is associated with the traffic policy when the **class** command is used. The **class** command must be entered after you enter the policy map configuration mode. After entering the **class** command, the switch is automatically in policy map class configuration mode, which is where the QoS policies for the traffic policy are defined.

The following policy map class-actions are supported:

- admit—Admits the request for Call Admission Control (CAC).
- bandwidth—Bandwidth configuration options.
- exit—Exits from the QoS class action configuration mode.

- no—Negates or sets default values for the command.
- police—Policer configuration options.
- priority—Strict scheduling priority configuration options for this class.
- queue-buffers—Queue buffer configuration options.
- queue-limit—Queue maximum threshold for Weighted Tail Drop (WTD) configuration options.
- service-policy—Configures the QoS service policy.
- set—Sets QoS values using the following options:
    - CoS values
    - DSCP values
    - Precedence values
    - QoS group values
    - WLAN values
- shape—Traffic-shaping configuration options.

Before You Begin

You should have first created a class map.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** *policy-map name*
3. **class** {*class-name* | **class-default**}
4. **admit**
5. **bandwidth** {**kb/s** *kb/s value* | **percent** *percentage* | **remaining** {*percent* | *ratio*}}
6. **exit**
7. **no**
8. **police** {*target_bit_rate* | **cir** | **rate**}
9. **priority** {*kb/s* | **level** *level value* | **percent** *percentage value*}
10. **queue-buffers ratio** *ratio limit*
11. **queue-limit** {*packets* | **cos** | **dscp** | **percent**}
12. **service-policy** *policy-map name*
13. **set** {**cos** | **dscp** | **ip** | **precedence** | **qos-group** | **wlan**}
14. **shape average** {*target _bit_rate* | **percent**}
15. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**policy-map** *policy-map name*
	
	Example:
	
	`Switch(config)# policy-map test_2000`
	`Switch(config-pmap)#`
		Enters policy map configuration mode.
		
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.

Step 3	**class** {*class-name* | **class-default**}
	
	Example:
	
	`Switch(config-pmap)# class test_1000`
	`Switch(config-pmap-c)#`
		Specifies the name of the class whose policy you want to create or change.
		
		You can also create a system default class for unclassified packets.

Step 4	**admit**
	
	Example:
	
	`Switch(config-pmap-c)# admit cac wmm-tspec`
	`Switch(config-pmap-c)#`
		(Optional) Admits the request for Call Admission Control (CAC). For a more detailed example of this command and its usage, see Configuring Call Admission Control , on page 290.
		
		Note This command only configures CAC for wireless QoS.

Step 5	**bandwidth** {**kb/s** *kb/s value* | **percent** *percentage* | **remaining** {*percent* | *ratio*}}
	
	Example:
	
	`Switch(config-pmap-c)# bandwidth 50`
	`Switch(config-pmap-c)#`
		(Optional) Sets the bandwidth using one of the following:
		
		• **kb/s—**Kilobits per second, enter a value between 20000 and 10000000 for Kb/s.
		• **percent—**Enter the percentage of the total bandwidth to be used for this policy map.
		• **remaining—**Enter the percentage ratio of the remaining bandwidth.
		
		For a more detailed example of this command and its usage, see Configuring Bandwidth , on page 297.

Step 6	**exit**
	
	Example:
	
	`Switch(config-pmap-c)# exit`
	`Switch(config-pmap-c)#`
		(Optional) Exits from QoS class action configuration mode.

Step 7	**no**	(Optional) Negates the command.

	Command or Action	Purpose

	Example:
	`Switch(config-pmap-c)# no`
	`Switch(config-pmap-c)#`	

Step 8	**police** {*target_bit_rate* | **cir** | **rate**}
	Example:
	`Switch(config-pmap-c)# police 100000`
	`Switch(config-pmap-c)#`
		(Optional) Configures the policer:
		• *target_bit_rate*—Enter the bit rate per second, enter a value between 8000 and 10000000000.
		• **cir**—Committed Information Rate
		• **rate**—Specify police rate, PCR for hierarchical policies or SCR for single-level ATM 4.0 policer policies.
		For a more detailed example of this command and its usage, see Configuring Police , on page 299.

Step 9	**priority** {*kb/s* | **level** *level value* | **percent** *percentage value*}
	Example:
	`Switch(config-pmap-c)# priority percent 50`
	`Switch(config-pmap-c)#`
		(Optional) Sets the strict scheduling priority for this class. Command options include:
		• *kb/s*—Kilobits per second, enter a value between 1 and 2000000.
		• **level**—Establishes a multi-level priority queue. Enter a value (1 or 2).
		• **percent**—Enter a percent of the total bandwidth for this priority.
		For a more detailed example of this command and its usage, see Configuring Priority, on page 302.

Step 10	**queue-buffers ratio** *ratio limit*
	Example:
	`Switch(config-pmap-c)# queue-buffers ratio 10`
	`Switch(config-pmap-c)#`
		(Optional) Configures the queue buffer for the class. Enter the queue buffers ratio limit (0 to 100).
		For a more detailed example of this command and its usage, see Configuring Queue Buffers , on page 304.

Step 11	**queue-limit** {*packets* | **cos** | **dscp** | **percent**}
	Example:
	`Switch(config-pmap-c)# queue-limit cos 7 percent 50`
	`Switch(config-pmap-c)#`
		(Optional) Specifies the queue maximum threshold for the tail drop:
		• *packets*—Packets by default, enter a value between 1 to 2000000.
		• **cos**—Enter the parameters for each COS value.
		• **dscp**—Enter the parameters for each DSCP value.
		• **percent**—Enter the percentage for the threshold.
		For a more detailed example of this command and its usage, see Configuring Queue Limits , on page 306.

Step 12	**service-policy** *policy-map name*	(Optional) Configures the QoS service policy.

	Command or Action	Purpose

	Example:
	`Switch(config-pmap-c)# service-policy test_2000`
	`Switch(config-pmap-c)#`	

Step 13	**set** {**cos** | **dscp** | **ip** | **precedence** | **qos-group** | **wlan**}
	Example:
	`Switch(config-pmap-c)# set cos 7`
	`Switch(config-pmap-c)#`
		(Optional) Sets the QoS values. Possible QoS configuration values include:
		• **cos**—Sets the IEEE 802.1Q/ISL class of service/user priority.
		• **dscp**—Sets DSCP in IP(v4) and IPv6 packets.
		• **ip**—Sets IP specific values.
		• **precedence**—Sets precedence in IP(v4) and IPv6 packet.
		• **qos-group**—Sets the QoS Group.
		• **wlan**—Sets the WLAN user-priority.

Step 14	**shape average** {*target _bit_rate* | **percent**}
	Example:
	`Switch(config-pmap-c) #shape average percent 50`
	`Switch(config-pmap-c) #`
		(Optional) Sets the traffic shaping. Command parameters include:
		• *target_bit_rate*—Target bit rate.
		• **percent**—Percentage of interface bandwidth for Committed Information Rate.
		For a more detailed example of this command and its usage, see Configuring Shaping , on page 309.

Step 15	**end**
	Example:
	`Switch(config-pmap-c) #end`
	`Switch(config-pmap-c) #`
		Saves the configuration changes.

What to Do Next

Configure the interface.

Related Topics

Policy Maps, on page 250

Configuring Class-Based Packet Marking

This procedure explains how to configure the following class-based packet marking features on your switch:

- CoS value
- DSCP value
- IP value
- Precedence value
- QoS group value
- WLAN value

Before You Begin

You should have created a class map and a policy map before beginning this procedure.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** *policy name*
3. **class** *class name*
4. **set cos** {*cos value* | **cos table** *table-map name* | **dscp table** *table-map name* | **precedence table** *table-map name* | **qos-group table** *table-map name* | **wlan user-priority table** *table-map name*}
5. **set dscp** {*dscp value* | **default** | **dscp table** *table-map name* | **ef** | **precedence table** *table-map name* | **qos-group table** *table-map name* | **wlan user-priority table** *table-map name*}
6. **set ip** {**dscp** | **precedence**}
7. **set precedence** {*precedence value* | **cos table** *table-map name* | **dscp table** *table-map name* | **precedence table** *table-map name* | **qos-group table** *table-map name*}
8. **set qos-group** {*qos-group value* | **dscp table** *table-map name* | **precedence table** *table-map name*}
9. **set wlan user-priority** {*wlan user-priority value* | **cos table** *table-map name* | **dscp table** *table-map name* | **qos-group table** *table-map name* | **wlan table** *table-map name*}
10. **end**
11. **show policy-map**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**policy-map** *policy name*
	
	Example:
	
	`Switch(config)# policy-map policy1`
	`Switch(config-pmap)#`
		Enters policy map configuration mode.
		
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.

	Command or Action	Purpose

Step 3	**class** *class name*
	
	Example:
	
	`Switch(config-pmap)# class class1`
	`Switch(config-pmap-c)#`
		Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change.
		
		Command options for policy class map configuration mode include the following:
		
		• **admit—**Admits the request for Call Admission Control (CAC).
		• **bandwidth—**Bandwidth configuration options.
		• **exit—**Exits from the QoS class action configuration mode.
		• **no—**Negates or sets default values for the command.
		• **police—**Policer configuration options.
		• **priority—**Strict scheduling priority configuration options for this class.
		• **queue-buffers—**Queue buffer configuration options.
		• **queue-limit—**Queue maximum threshold for Weighted Tail Drop (WTD) configuration options.
		• **service-policy—**Configures the QoS service policy.
		• **set—**Sets QoS values using the following options:
		    ◦ CoS values
		    ◦ DSCP values
		    ◦ Precedence values
		    ◦ QoS group values
		    ◦ WLAN values
		• **shape—**Traffic-shaping configuration options.
		
		Note This procedure describes the available configurations using **set** command options. The other command options (**admit**, **bandwidth**, etc.) are described in other sections of this guide. Although this task lists all of the possible **set** commands, only one **set** command is supported per class.

Step 4	**set cos** {*cos value* | **cos table** *table-map name* | **dscp table** *table-map name* | **precedence table** *table-map name* | **qos-group table** *table-map name* | **wlan user-priority table** *table-map name*}
	
	Example:
	
	`Switch(config-pmap)# set cos 5`
	`Switch(config-pmap)#`
		(Optional) Sets the specific IEEE 802.1Q Layer 2 CoS value of an outgoing packet. Values are from 0 to7.
		
		You can also set the following values using the **set cos** command:
		
		• **cos table—**Sets the CoS value based on a table map.
		• **dscp table—**Sets the code point value based on a table map.
		• **precedence table—**Sets the code point value based on a table map.
		• **qos-group table—**Sets the CoS value from QoS group based on a table map.

	Command or Action	Purpose
		• **wlan user-priority table—**Sets the CoS value from the WLAN user priority based on a table map.

Step 5	**set dscp** {*dscp value* | **default** | **dscp table** *table-map name* | **ef** | **precedence table** *table-map name* | **qos-group table** *table-map name* | **wlan user-priority table** *table-map name*}
	
	Example:
	
	`Switch(config-pmap)# set dscp af11`
	`Switch(config-pmap)#`
		(Optional) Sets the DSCP value.
		
		In addition to setting specific DSCP values, you can also set the following using the **set dscp** command:
		
		• **default—**Matches packets with default DSCP value (000000).
		• **dscp table—**Sets the packet DSCP value from DSCP based on a table map.
		• **ef—**Matches packets with EF DSCP value (101110).
		• **precedence table—**Sets the packet DSCP value from precedence based on a table map.
		• **qos-group table—**Sets the packet DSCP value from a QoS group based upon a table map.
		• **wlan user-priority table—**Sets the packet DSCP value based upon a WLAN user-priority based upon a table map.

Step 6	**set ip** {**dscp** | **precedence**}
	
	Example:
	
	`Switch(config-pmap)# set ip dscp c3`
	`Switch(config-pmap)#`
		(Optional) Sets IP specific values. These values are either IP DSCP or IP precedence values.
		
		You can set the following values using the **set ip dscp** command:
		
		• *dscp value*—Sets a specific DSCP value.
		• **default—**Matches packets with default DSCP value (000000).
		• **dscp table—**Sets the packet DSCP value from DSCP based on a table map.
		• **ef—**Matches packets with EF DSCP value (101110).
		• **precedence table—**Sets the packet DSCP value from precedence based on a table map.
		• **qos-group table—**Sets the packet DSCP value from a QoS group based upon a table map.
		• **wlan user-priority table—**Sets the packet DSCP value based upon a WLAN user-priority based upon a table map.
		
		You can set the following values using the **set ip precedence** command:
		
		• *precedence value*—Sets the precedence value (from 0 to 7) .
		• **cos table—**Sets the packet precedence value from Layer 2 CoS based on a table map.
		• **dscp table—**Sets the packet precedence from DSCP value based on a table map.

	Command or Action	Purpose
		• **precedence table**—Sets the precedence value from precedence based on a table map • **qos-group table**—Sets the precedence value from a QoS group based upon a table map.

Step 7	**set precedence** {*precedence value* | **cos table** *table-map name* | **dscp table** *table-map name* | **precedence table** *table-map name* | **qos-group table** *table-map name*}
	
	Example:
	
	`Switch(config-pmap)# set precedence 5`
	`Switch(config-pmap)#`
		(Optional) Sets precedence values in IPv4 and IPv6 packets.
		You can set the following values using the **set precedence** command:
		• *precedence value*—Sets the precedence value (from 0 to 7) .
		• **cos table**—Sets the packet precedence value from Layer 2 CoS on a table map.
		• **dscp table**—Sets the packet precedence from DSCP value on a table map.
		• **precedence table**—Sets the precedence value from precedence based on a table map.
		• **qos-group table**—Sets the precedence value from a QoS group based upon a table map.

Step 8	**set qos-group** {*qos-group value* | **dscp table** *table-map name* | **precedence table** *table-map name*}
	
	Example:
	
	`Switch(config-pmap)# set qos-group 10`
	`Switch(config-pmap)#`
		(Optional) Sets QoS group values. You can set the following values using this command:
		• *qos-group value*—A number from 1 to 31.
		• **dscp table**—Sets the code point value from DSCP based on a table map.
		• **precedence table**—Sets the code point value from precedence based on a table map.

Step 9	**set wlan user-priority** {*wlan user-priority value* | **cos table** *table-map name* | **dscp table** *table-map name* | **qos-group table** *table-map name* | **wlan table** *table-map name*}
	
	Example:
	
	`Switch(config-pmap)# set wlan user-priority 1`
	`Switch(config-pmap)#`
		(Optional) Sets the WLAN user priority value. You can set the following values using this command:
		• *wlan user-priority value*—A value between 0 to 7.
		• **cos table**—Sets the WLAN user priority value from CoS based on a table map.
		• **dscp table**—Sets the WLAN user priority value from DSCP based on a table map.
		• **qos-group table**—Sets the WLAN user priority value from QoS group based on a table map.
		• **wlan table**—Sets the WLAN user priority value from the WLAN user priority based on a table map.

Step 10	**end**	Saves configuration changes.

	Command or Action	Purpose

	Example:
	`Switch(config-pmap)# end`
	`Switch#`	

Step 11	**show policy-map** **Example:** `Switch# show policy-map`	(Optional) Displays policy configuration information for all classes configured for all service policies.

What to Do Next

Attach the traffic policy to an interface using the **service-policy** command.

Attaching a Traffic Policy to an Interface

After the traffic class and traffic policy are created, you must use the **service-policy** interface configuration command to attach a traffic policy to an interface, and to specify the direction in which the policy should be applied (either on packets coming into the interface or packets leaving the interface).

Before You Begin

A traffic class and traffic policy must be created before attaching a traffic policy to an interface.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type*
3. **service-policy** {**input** *policy-map* | **output** *policy-map* }
4. **end**
5. **show policy map**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *type*	Enters interface configuration mode and configures an interface.

	Command or Action	Purpose

	Example:
	`Switch(config)# interface GigabitEthernet1/0/1`
	`Switch(config-if)#`
		Command parameters for the interface configuration include:
		• **Auto Template—** Auto-template interface
		• **Capwap—**CAPWAP tunnel interface
		• **GigabitEthernet—**Gigabit Ethernet IEEE 802
		• **GroupVI—**Group virtual interface
		• **Internal Interface—** Internal interface
		• **Loopback—**Loopback interface
		• **Null—**Null interface
		• **Port-channel—**Ethernet Channel of interface
		• **TenGigabitEthernet—**10-Gigabit Ethernet
		• **Tunnel—**Tunnel interface
		• **Vlan—**Catalyst VLANs
		• **Range—**Interface range

Step 3	**service-policy** {**input** *policy-map* | **output** *policy-map* }
	Example:
	`Switch(config-if)# service-policy output policy_map_01`
	`Switch(config-if)#`
		Attaches a policy map to an input or output interface. This policy map is then used as the service policy for that interface.
		In this example, the traffic policy evaluates all traffic leaving that interface.

Step 4	**end**
	Example:
	`Switch(config-if)# end`
	`Switch#`
		Saves configuration changes.

Step 5	**show policy map** **Example:** `Switch# show policy map`	(Optional) Displays statistics for the policy on the specified interface.

What to Do Next

Proceed to attach any other traffic policy to an interface, and to specify the direction in which the policy should be applied.

Related Topics

Policy Map on Physical Port, on page 251

Classifying, Policing, and Marking Traffic on Physical Ports by Using Policy Maps

You can configure a nonhierarchical policy map on a physical port that specifies which traffic class to act on. Actions supported are remarking and policing.

Before You Begin

You should have already decided upon the classification, policing, and marking of your network traffic by policy maps prior to beginning this procedure.

SUMMARY STEPS

1. **configure terminal**
2. **class-map** {*class-map name* | **match-any** }
3. **match access-group** { *access list index* | *access list name* }
4. **policy-map** *policy-map-name*
5. **class** {*class-map-name* | **class-default**}
6. **set** {**cos** | **dscp** | **ip** | **precedence** | **qos-group** | **wlan user-priority**}
7. **police** {*target_bit_rate* | **cir** | **rate** }
8. **exit**
9. **exit**
10. **interface** *interface-id*
11. **service-policy input** *policy-map-name*
12. **end**
13. **show policy-map** [*policy-map-name* [**class** *class-map-name*]]
14. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**class-map** {*class-map name* | **match-any** }
	
	Example:
	`Switch(config)# class-map ipclass1`
	`Switch(config-cmap)# exit`
	`Switch(config)#`
		Enters class map configuration mode.
		• Creates a class map to be used for matching packets to the class whose name you specify.
		• If you specify **match-any**, one of the match criteria must be met for traffic entering the traffic class to be classified as part of the traffic class. This is the default.

	Command or Action	Purpose

Step 3	**match access-group** { *access list index* | *access list name* }
	
	Example:
	
	`Switch(config-cmap)# match access-group 1000`
	`Switch(config-cmap)# exit`
	`Switch(config)#`
		Specifies the classification criteria to match to the class map. You can match on the following criteria:
		
		• **access-group—**Matches to access group.
		• **class-map—**Matches to another class map.
		• **cos—**Matches to a CoS value.
		• **dscp—**Matches to a DSCP value.
		• **ip—**Matches to a specific IP value.
		• **non-client-nrt—**Matches non-client NRT.
		• **precedence—**Matches precedence in IPv4 and IPv6 packets.
		• **qos-group—**Matches to a QoS group.
		• **vlan—**Matches to a VLAN.
		• **wlan—**Matches to a wireless LAN.

Step 4	**policy-map** *policy-map-name*
	
	Example:
	
	`Switch(config)# policy-map flowit`
	`Switch(config-pmap)#`
		Creates a policy map by entering the policy map name, and enters policy-map configuration mode.
		
		By default, no policy maps are defined.

Step 5	**class** {*class-map-name* | **class-default**}
	
	Example:
	
	`Switch(config-pmap)# class ipclass1`
	`Switch(config-pmap-c)#`
		Defines a traffic classification, and enter policy-map class configuration mode.
		
		By default, no policy map class-maps are defined.
		
		If a traffic class has already been defined by using the **class-map** global configuration command, specify its name for *class-map-name* in this command.
		
		A **class-default** traffic class is predefined and can be added to any policy. It is always placed at the end of a policy map. With an implied **match any** included in the **class-default** class, all packets that have not already matched the other traffic classes will match **class-default**.

Step 6	**set** {**cos** | **dscp** | **ip** | **precedence** | **qos-group** | **wlan user-priority**}
	
	Example:
	
	`Switch(config-pmap-c)# set dscp 45`
	`Switch(config-pmap-c)#`
		(Optional) Sets the QoS values. Possible QoS configuration values include:
		
		• **cos—**Sets the IEEE 802.1Q/ISL class of service/user priority.
		• **dscp—**Sets DSCP in IP(v4) and IPv6 packets.
		• **ip—**Sets IP specific values.
		• **precedence—**Sets precedence in IP(v4) and IPv6 packet.
		• **qos-group—**Sets QoS group.

	Command or Action	Purpose
		• **wlan user-priority**—Sets WLAN user priority. In this example, the **set dscp** command classifies the IP traffic by setting a new DSCP value in the packet.
Step 7	**police** {*target_bit_rate* | **cir** | **rate** } **Example:** `Switch(config-pmap-c)#` **`police 100000 conform-action transmit exceed-action drop`** `Switch(config-pmap-c)#`	(Optional) Configures the policer: • *target_bit_rate*—Specifies the bit rate per second, enter a value between 8000 and 10000000000. • **cir**—Committed Information Rate. • **rate**—Specifies the police rate, PCR for hierarchical policies, or SCR for single-level ATM 4.0 policer policies. In this example, the **police** command adds a policer to the class where any traffic beyond the 100000 set target bit rate is dropped.
Step 8	**exit** **Example:** `Switch(config-pmap-c)#` **`exit`**	Returns to policy map configuration mode.
Step 9	**exit** **Example:** `Switch(config-pmap)#` **`exit`**	Returns to global configuration mode.
Step 10	**interface** *interface-id* **Example:** `Switch(config)#` **`interface gigabitethernet 2/0/1`**	Specifies the port to attach to the policy map, and enters interface configuration mode. Valid interfaces include physical ports.
Step 11	**service-policy input** *policy-map-name* **Example:** `Switch(config-if)#` **`service-policy input flowit`**	Specifies the policy-map name, and applies it to an ingress port. Only one policy map per ingress port is supported.
Step 12	**end** **Example:** `Switch(config-if)#` **`end`**	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 13	**show policy-map** [*policy-map-name* [**class** *class-map-name*]] **Example:** `Switch# show policy-map`	(Optional) Verifies your entries.
Step 14	**copy running-config startup-config** **Example:** `Switch# copy-running-config startup-config`	(Optional) Saves your entries in the configuration file.

What to Do Next

If applicable to your QoS configuration, configure classification, policing, and marking of traffic on SVIs by using policy maps.

Classifying, Policing, and Marking Traffic on SVIs by Using Policy Maps

Before You Begin

You should have already decided upon the classification, policing, and marking of your network traffic by using policy maps prior to beginning this procedure.

SUMMARY STEPS

1. **configure terminal**
2. **class-map** {*class-map name* | **match-any** }
3. **match vlan** *vlan number*
4. **policy-map** *policy-map-name*
5. **description** *description*
6. **class** {*class-map-name* | **class-default**}
7. **set** {**cos** | **dscp** | **ip** | **precedence** | **qos-group** | **wlan user-priority**}
8. **police** {*target_bit_rate* | **cir** | **rate**}
9. **exit**
10. **exit**
11. **interface** *interface-id*
12. **service-policy input** *policy-map-name*
13. **end**
14. **show policy-map** [*policy-map-name* [**class** *class-map-name*]]
15. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**class-map** {*class-map name* | **match-any** } **Example:** `Switch(config)# class-map class_vlan100`	Enters class map configuration mode. • Creates a class map to be used for matching packets to the class whose name you specify. • If you specify **match-any**, one of the match criteria must be met for traffic entering the traffic class to be classified as part of the traffic class. This is the default.

Step 3	**match vlan** *vlan number*
	
	Example:
	
	`Switch(config-cmap)# match vlan 100`
	`Switch(config-cmap)# exit`
	`Switch(config)#`
		Specifies the VLAN to match to the class map.

Step 4	**policy-map** *policy-map-name*
	
	Example:
	
	`Switch(config)# policy-map policy_vlan100`
	`Switch(config-pmap)#`
		Creates a policy map by entering the policy map name, and enters policy-map configuration mode.
		
		By default, no policy maps are defined.

Step 5	**description** *description* **Example:** `Switch(config-pmap)# description vlan 100`	(Optional) Enters a description of the policy map.

Step 6	**class** {*class-map-name* | **class-default**}
	
	Example:
	
	`Switch(config-pmap)# class class_vlan100`
	`Switch(config-pmap-c)#`
		Defines a traffic classification, and enters the policy-map class configuration mode.
		
		By default, no policy map class-maps are defined.
		
		If a traffic class has already been defined by using the **class-map** global configuration command, specify its name for *class-map-name* in this command.
		
		A **class-default** traffic class is predefined and can be added to any policy. It is always placed at the end of a policy map. With an implied **match any** included in the **class-default** class, all packets that have not already matched the other traffic classes will match **class-default**.

	Command or Action	Purpose

Step 7	**set** {**cos** | **dscp** | **ip** | **precedence** | **qos-group** | **wlan user-priority**}
	
	Example:
	
	`Switch(config-pmap-c)# set dscp af23`
	`Switch(config-pmap-c)#`
		(Optional) Sets the QoS values. Possible QoS configuration values include:
		
		• **cos—**Sets the IEEE 802.1Q/ISL class of service/user priority.
		• **dscp—**Sets DSCP in IP(v4) and IPv6 packets.
		• **ip—**Sets IP specific values.
		• **precedence—**Sets precedence in IP(v4) and IPv6 packet.
		• **qos-group—**Sets QoS group.
		• **wlan user-priority—**Sets WLAN user-priority.
		
		In this example, the **set dscp** command classifies the IP traffic by matching the packets with a DSCP value of AF23 (010010).

Step 8	**police** {*target_bit_rate* | **cir** | **rate**}
	
	Example:
	
	`Switch(config-pmap-c)# police 200000`
	`conform-action transmit`
	`exceed-action drop`
	`Switch(config-pmap-c)#`
		(Optional) Configures the policer:
		
		• *target_bit_rate***—**Specifies the bit rate per second. Enter a value between 8000 and 10000000000.
		• **cir—**Committed Information Rate.
		• **rate—**Specifies the police rate, PCR for hierarchical policies, or SCR for single-level ATM 4.0 policer policies.
		
		In this example, the **police** command adds a policer to the class where any traffic beyond the 200000 set target bit rate is dropped.

Step 9	**exit** **Example:** `Switch(config-pmap-c)# exit`	Returns to policy map configuration mode.
Step 10	**exit** **Example:** `Switch(config-pmap)# exit`	Returns to global configuration mode.

Step 11	**interface** *interface-id*
	
	Example:
	
	`Switch(config)# interface`
	`gigabitethernet 1/0/3`
		Specifies the port to attach to the policy map, and enters interface configuration mode.
		
		Valid interfaces include physical ports.

	Command or Action	Purpose
Step 12	**service-policy input** *policy-map-name* **Example:** `Switch(config-if)# service-policy input policy_vlan100`	Specifies the policy-map name, and applies it to an ingress port. Only one policy map per ingress port is supported.
Step 13	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 14	**show policy-map** [*policy-map-name* [**class** *class-map-name*]] **Example:** `Switch# show policy-map`	(Optional) Verifies your entries.
Step 15	**copy running-config startup-config** **Example:** `Switch# copy-running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Policy Map on VLANs, on page 251

Examples: Policer VLAN Configuration, on page 319

Configuring Table Maps

Table maps are a form of marking, and also enable the mapping and conversion of one field to another using a table. For example, a table map can be used to map and convert a Layer 2 CoS setting to a precedence value in Layer 3.

Note A table map can be referenced in multiple policies or multiple times in the same policy.

SUMMARY STEPS

1. **configure terminal**
2. **table-map** *name* {**default** {*default value* | **copy** | **ignore**} | **exit** | **map** {**from** *from value* **to** *to value* } | **no**}
3. **map from** *value* **to** *value*
4. **exit**
5. **exit**
6. **show table-map**
7. **configure terminal**
8. **policy-map**
9. **class class-default**
10. **set cos dscp table** *table map name*
11. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**table-map** *name* {**default** {*default value* | **copy** | **ignore**} | **exit** | **map** {**from** *from value* **to** *to value* } | **no**}
	
	Example:
	
	`Switch(config)# table-map table01`
	`Switch(config-tablemap)#`
		Creates a table map and enters the table map configuration mode. In table map configuration mode, you can perform the following tasks:
		
		• **default—**Configures the table map default value, or sets the default behavior for a value not found in the table map to copy or ignore.
		
		• **exit—**Exits from the table map configuration mode.
		
		• **map—**Maps a *from* to a *to* value in the table map.
		
		• **no—**Negates or sets the default values of the command.

Step 3	**map from** *value* **to** *value*
	
	Example:
	
	`Switch(config-tablemap)# map from 0 to 2`
	`Switch(config-tablemap)# map from 1 to 4`
	`Switch(config-tablemap)# map from 24 to 3`
	`Switch(config-tablemap)# map from 40 to 6`
	`Switch(config-tablemap)# default 0`
	`Switch(config-tablemap)#`
		In this step, packets with DSCP values 0 are marked to the CoS value 2, DSCP value 1 to the CoS value 4, DSCP value 24 to the CoS value 3, DSCP value 40 to the CoS value 6 and all others to the CoS value 0.
		
		Note The mapping from CoS values to DSCP values in this example is configured by using the **set** policy map class configuration command as described in a later step in this procedure.

	Command or Action	Purpose

Step 4	**exit**
	
	Example:
	
	`Switch(config-tablemap)# exit`
	`Switch(config)#`
		Returns to global configuration mode.

Step 5	**exit**
	
	Example:
	
	`Switch(config) exit`
	`Switch#`
		Returns to privileged EXEC mode.

Step 6	**show table-map**
	
	Example:
	
	`Switch# show table-map`
	`Table Map table01`
	`    from 0 to 2`
	`    from 1 to 4`
	`    from 24 to 3`
	`    from 40 to 6`
	`    default 0`
		Displays the table map configuration.

Step 7	**configure terminal**
	
	Example:
	
	`Switch# configure terminal`
	`Switch(config)#`
		Enters global configuration mode.

Step 8	**policy-map**
	
	Example:
	
	`Switch(config)# policy-map table-policy`
	`Switch(config-pmap)#`
		Configures the policy map for the table map.

Step 9	**class class-default**
	
	Example:
	
	`Switch(config-pmap)# class  class-default`
	`Switch(config-pmap-c)#`
		Matches the class to the system default.

Step 10	**set cos dscp table** *table map name* **Example:** `Switch(config-pmap-c)# set cos dscp table table01`	If this policy is applied on input port, that port will have trust DSCP enabled on that port and marking will take place depending upon the specified table map.

	Command or Action	Purpose
	`Switch(config-pmap-c)#`	

Step 11	**end**
	
	Example:
	`Switch(config-pmap-c)# end`
	`Switch#`
		Returns to privileged EXEC mode.

What to Do Next

Configure any additional policy maps for QoS for your network. After creating your policy maps, attach the traffic policy or polices to an interface using the **service-policy** command.

Related Topics

Table Map Marking, on page 253

Examples: Table Map Marking Configuration, on page 321

Configuring Trust

Configuring Trust Behavior for the Device Type

This procedure explains how to configure trust for one or more device classes within your network configuration.

Before You Begin

There are two types of trust behavior supported on the switch:

- Trust QoS at the policy level—You can configure trust for individual packets by creating specific policy maps and applying them on an interface. If you do not configure a specific policy map, then the default is to trust DSCP.
- Trust devices at the interface level—You can configure trust for the device using the **trust device** interface configuration command.

Note The default mode on an interface is trusted and changes to untrusted only when an untrusted device is detected. In the untrusted mode, the DSCP, IP precedence, or CoS value is reset to 0.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type*
3. **trust device** { **cisco-phone** | **cts** | **ip-camera** | **media-player** }
4. **end**
5. **show interface status**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**interface** *type*
	
	Example:
	`Switch(config)# interface GigabitEthernet1/0/1`
	`Switch(config-if)#`
		Enters interface configuration mode and configures an interface.
		Command parameters for the interface configuration include:
		• **Auto Template—** Auto-Template interface
		• **Capwap—**Capwap tunnel interface
		• **GigabitEthernet—**Gigabit Ethernet IEEE 802
		• **GroupVI—**Group virtual interface
		• **Internal Interface—**Internal interface
		• **Loopback—**Loopback interface
		• **Null—**Null interface
		• **Port-channel—**Ethernet Channel of interface
		• **TenGigabitEthernet—**10-Gigabit Ethernet
		• **Tunnel—**Tunnel interface
		• **Vlan—**Catalyst VLANs
		• **range—**interface range

Step 3	**trust device** { **cisco-phone** | **cts** | **ip-camera** | **media-player** }
	
	Example:
	`Switch(config-if)# trust device cisco-phone`
	`Switch(config-if)#`
		Configures the trust value for the interface. You can configure trust for the following supported devices:
		• **cisco-phone—**Cisco IP Phone
		• **cts—**Cisco TelePresence system
		• **ip-camera—**IPVSC
		• **media-player—**DMP

	Command or Action	Purpose

Step 4	**end**
	
	Example:
	
	`Switch(config-if)# end`
	`Switch#`
		Saves configuration changes.

Step 5	**show interface status**
	
	Example:
	
	`Switch# show interface status`
	`Switch#`
		(Optional) Displays the configured interface's status.

What to Do Next

Connect the trusted device to the appropriately configured trusted port on the switch.

Related Topics

Port Security on a Trusted Boundary for Cisco IP Phones, on page 263

Configuring QoS Features and Functionality

Configuring Call Admission Control

This task explains how to configure class-based, unconditional packet marking features on your switch for Call Admission Control (CAC).

SUMMARY STEPS

1. **configure terminal**
2. **class-map** *class name*
3. **match dscp** *dscp value*
4. **exit**
5. **class-map** *class name*
6. **match dscp** *dscp value*
7. **exit**
8. **table-map** *name*
9. **default copy**
10. **exit**
11. **table-map** *name*
12. **default copy**
13. **exit**
14. **policy-map** *policy name*
15. **class** *class-map-name*
16. **priority level** *level_value*
17. **police** [*target_bit_rate* | **cir** | **rate**]
18. **admit cac wmm-tspec**
19. **rate** *value*
20. **wlan-up** *value*
21. **exit**
22. **exit**
23. **class** *class name*
24. **priority level** *level_value*
25. **police** [*target_bit_rate* | **cir** | **rate**]
26. **admit cac wmm-tspec**
27. **rate** *value*
28. **wlan-up** *value*
29. **exit**
30. **exit**
31. **policy-map** *policy name*
32. **class** *class-map-name*
33. **set dscp dscp table** *table_map_name*
34. **set wlan user-priority dscp table** *table_map_name*
35. **shape average** {*target bit rate* | **percent** *percentage*}
36. **queue-buffers** {**ratio** *ratio value*}
37. **service-policy** *policy_map_name*
38. **end**
39. **show policy-map**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**class-map** *class name*
	
	Example:
	
	`Switch(config)# class-map voice`
	`Switch(config-cmap)#`
		Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change. Command options for policy class map configuration mode include the following:
		
		• *word*—Class map name.
		
		• **class-default**—System default class matching any otherwise unclassified packets.

Step 3	**match dscp** *dscp value* **Example:** `Switch(config-cmap)#  match dscp 46`	(Optional) Matches the DSCP values in IPv4 and IPv6 packets.

Step 4	**exit**
	
	Example:
	
	`Switch(config-cmap)# exit`
	`Switch(config)#`
		Returns to global configuration mode.

Step 5	**class-map** *class name*
	
	Example:
	
	`Switch(config)# class-map video`
	`Switch(config-cmap)#`
		Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change. Command options for policy class map configuration mode include the following:
		
		• *word*—Class map name.
		
		• **class-default**—System default class matching any otherwise unclassified packets.

Step 6	**match dscp** *dscp value* **Example:** `Switch(config-cmap)#  match dscp 34`	(Optional) Matches the DSCP values in IPv4 and IPv6 packets.
Step 7	**exit**	Returns to global configuration mode.

	Command or Action	Purpose

	Example:
	`Switch(config-cmap)# exit`
	`Switch(config)#`	

Step 8	**table-map** *name*
	Example:
	`Switch(config)# table-map  dscp2dscp`
	`Switch(config-tablemap)#`
		Creates a table map and enters the table map configuration mode.

Step 9	**default copy** **Example:** `Switch(config-tablemap)# default copy`	Sets the default behavior for value not found in the table map to copy. **Note** This is the default option. You can also do a mapping of values for DSCP to DSCP.

Step 10	**exit**
	Example:
	`Switch(config-tablemap)# exit`
	`Switch(config)#`
		Returns to global configuration mode.

Step 11	**table-map** *name*
	Example:
	`Switch(config)# table-map dscp2up`
	`Switch(config-tablemap)#`
		Creates a new table map and enters the table map configuration mode.

Step 12	**default copy** **Example:** `Switch(config-tablemap)# default copy`	Sets the default behavior for value not found in the table map to copy. **Note** This is the default option. You can also do a mapping of values for DSCP to UP.

Step 13	**exit**
	Example:
	`Switch(config-tablemap)# exit`
	`Switch(config)#`
		Returns to global configuration mode.

Step 14	**policy-map** *policy name* **Example:** `Switch(config)#  policy-map ssid_child_cac`	Enters policy map configuration mode. Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.

	Command or Action	Purpose
	`Switch(config-pmap)#`	
Step 15	**class** *class-map-name* **Example:** `Switch(config-pmap)# class voice`	Defines an interface-level traffic classification, and enters policy-map configuration mode.
Step 16	**priority level** *level_value* **Example:** `Switch(config-pmap-c)# priority level 1`	The **priority** command assigns a strict scheduling priority for the class. **Note** Priority level 1 is more important than priority level 2. Priority level 1 reserves bandwidth that is processed first for QoS, so its latency is very low. Both priority level 1 and 2 reserve bandwidth.
Step 17	**police** [*target_bit_rate* | **cir** | **rate**] **Example:** `Switch(config-pmap-c)#  police cir 10m`	(Optional) Configures the policer: • *target_bit_rate*—Specifies the bit rate per second. Enter a value between 8000 and 10000000000. • **cir**—Committed Information Rate. • **rate**—Specifies the police rate, PCR for hierarchical policies, or SCR for single-level ATM 4.0 policer policies.

Step 18	**admit cac wmm-tspec**
	
	Example:
	
	`Switch(config-pmap-c)# admit cac wmm-tspec`
	`Switch(config-pmap-cac-wmm)#`
		Configures call admission control for the policy map.
		
		Note This command only configures CAC for wireless QoS.

Step 19	**rate** *value* **Example:** `Switch(config-pmap-admit-cac-wmm)# rate 5000`	Configures the target bit rate (Kilo Bits per second). Enter a value from 8 to 10000000.
Step 20	**wlan-up** *value* **Example:** `Switch(config-pmap-admit-cac-wmm)# wlan-up 6 7`	Configures the WLAN UP value. Enter a value from 0 to 7.
Step 21	**exit** **Example:** `Switch(config-pmap-admit-cac-wmm)# exit`	Returns to policy map class configuration mode.

	Command or Action	Purpose
	`Switch(config-pmap-c)#`	

Step 22	**exit**
	
	Example:
	`Switch(config-pmap-c)# exit`
	`Switch(config-pmap)#`
		Returns to policy map configuration mode.

Step 23	**class** *class name*
	
	Example:
	`Switch(config-pmap)# class video`
	`Switch(config-pmap-c)#`
		Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change. Command options for policy class map configuration mode include the following:
		• *word*—Class map name.
		• **class-default**—System default class matching any otherwise unclassified packets.

Step 24	**priority level** *level_value* **Example:** `Switch(config-pmap-c)# priority level 2`	The **priority** command assigns a strict scheduling priority for the class. **Note** Priority level 1 is more important than priority level 2. Priority level 1 reserves bandwidth that is processed first for QoS, so its latency is very low. Both priority level 1 and 2 reserve bandwidth.
Step 25	**police** [*target_bit_rate* | **cir** | **rate**] **Example:** `Switch(config-pmap-c)#  police cir 20m`	(Optional) Configures the policer: • *target_bit_rate*—Specifies the bit rate per second. Enter a value between 8000 and 10000000000. • **cir**—Committed Information Rate. • **rate**—Specifies the police rate, PCR for hierarchical policies, or SCR for single-level ATM 4.0 policer policies.

Step 26	**admit cac wmm-tspec**
	
	Example:
	`Switch(config-pmap-c)# admit cac wmm-tspec`
	`Switch(config-pmap-admit-cac-wmm)#`
		Configures call admission control for the policy map.
		Note This command only configures CAC for wireless QoS.

Step 27	**rate** *value* **Example:** `Switch(config-pmap-admit-cac-wmm)# rate 5000`	Configures the target bit rate (Kilo Bits per second). Enter a value from 8 to 10000000.
Step 28	**wlan-up** *value*	Configures the WLAN UP value. Enter a value from 0 to 7.

	Command or Action	Purpose
	Example: `Switch(config-pmap-admit-cac-wmm)# wlan-up 4 5`	

Step 29	**exit**
	
	Example:
	`Switch(config-pmap-cac-wmm)# exit`
	`Switch(config-pmap)#`
		Returns to policy map configuration mode.

Step 30	**exit**
	
	Example:
	`Switch(config-pmap)# exit`
	`Switch(config)#`
		Returns to global configuration mode.

Step 31	**policy-map** *policy name*
	
	Example:
	`Switch(config)# policy-map ssid_cac`
	`Switch(config-pmap)#`
		Enters policy map configuration mode.
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.

Step 32	**class** *class-map-name* **Example:** `Switch(config-pmap)# class default`	Defines an interface-level traffic classification, and enters policy-map configuration mode. In this example, the class map is set to default.
Step 33	**set dscp dscp table** *table_map_name* **Example:** `Switch(config-pmap-c)# set dscp dscp table dscp2dscp`	(Optional) Sets the QoS values. In this example, the **set dscp dscp table** command creates a table map and sets its values.
Step 34	**set wlan user-priority dscp table** *table_map_name* **Example:** `Switch(config-pmap-c)# set wlan user-priority dscp table dscp2up`	(Optional) Sets the QoS values. In this example, the **set wlan user-priority dscp table** command sets the WLAN user priority.

	Command or Action	Purpose
Step 35	**shape average** {*target bit rate* | **percent** *percentage*} **Example:** `Switch(config-pmap-c)# shape average 100000000`	Configures the average shape rate. You can configure the average shape rate by target bit rates (bits per second) or by percentage of interface bandwidth for the Committed Information Rate (CIR).
Step 36	**queue-buffers** {**ratio** *ratio value*} **Example:** `Switch(config-pmap-c)# queue-buffers ratio 0`	Configures the relative buffer size for the queue. **Note** The sum of all configured buffers in a policy must be less than or equal to 100 percent. Unallocated buffers are evenly distributed to all the remaining queues.
Step 37	**service-policy** *policy_map_name* **Example:** `Switch(config-pmap-c)#  service-policy ssid_child_cac`	Specifies the policy map for the service policy.

Step 38	**end**
	
	Example:
	
	`Switch(config-pmap)# end`
	`Switch#`
		Saves configuration changes.

Step 39	**show policy-map** **Example:** `Switch# show policy-map`	(Optional) Displays policy configuration information for all classes configured for all service policies.

What to Do Next

Configure any additional policy maps for QoS for your network. After creating your policy maps, attach the traffic policy or polices to an interface using the **service-policy** command.

For additional information about CAC, refer to the *System Management Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches).*

Configuring Bandwidth

This procedure explains how to configure bandwidth on your switch.

Before You Begin

You should have created a class map for bandwidth before beginning this procedure.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** *policy name*
3. **class** *class name*
4. **bandwidth** {*Kb/s* | **percent** *percentage* | **remaining** { **ratio** *ratio* }}
5. **end**
6. **show policy-map**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**policy-map** *policy name*
	
	Example:
	`Switch(config)# policy-map policy_bandwidth01`
	`Switch(config-pmap)#`
		Enters policy map configuration mode.
		
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.

Step 3	**class** *class name*
	
	Example:
	`Switch(config-pmap)# class class_bandwidth01`
	`Switch(config-pmap-c)#`
		Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change. Command options for policy class map configuration mode include the following:
		• *word*—Class map name.
		• **class-default**—System default class matching any otherwise unclassified packets.

Step 4	**bandwidth** {*Kb/s* | **percent** *percentage* | **remaining** { **ratio** *ratio* }}
	
	Example:
	`Switch(config-pmap-c)# bandwidth 200000`
	`Switch(config-pmap-c)#`
		Configures the bandwidth for the policy map. The parameters include:
		• *Kb/s*—Configures a specific value in kilobits per second (from 20000 to 10000000).
		• **percent-**—Allocates minimum bandwidth to a particular class based on a percentage. The queue can oversubscribe bandwidth in case other queues do not utilize the entire port bandwidth. The total sum cannot exceed 100 percent, and in case it is less than 100 percent, the rest of the bandwidth is equally divided along all bandwidth queues.

	Command or Action	Purpose
		• **remaining—** Allocates minimum bandwidth to a particular class. The queue can oversubscribe bandwidth in case other queues do not utilize entire port bandwidth. The total sum cannot exceed 100 percent. It is preferred to use this command when the **priority** command is used for certain queues in the policy. You can also assign ratios rather than percentages to each queue; the queues will be assigned certain weights which are inline with these ratios. Ratios can range from 0 to 100. Total bandwidth ratio allocation for the policy in this case can exceed 100. **Note** You cannot mix bandwidth types on a policy map. For example, you cannot configure bandwidth in a single policy map using both a bandwidth percent and in kilobits per second.

Step 5	**end**
	
	Example:
	
	`Switch(config-pmap-c)# end`
	`Switch#`
		Saves configuration changes.

Step 6	**show policy-map** **Example:** `Switch# show policy-map`	(Optional) Displays policy configuration information for all classes configured for all service policies.

What to Do Next

Configure any additional policy maps for QoS for your network. After creating the policy maps, attach the traffic policy or polices to an interface using the **service-policy** command.

Related Topics

Bandwidth, on page 258

Configuring Police

This procedure explains how to configure policing on your switch.

Before You Begin

You should have created a class map for policing before beginning this procedure.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** *policy name*
3. **class** *class name*
4. **police** {*target_bit_rate* [*burst bytes* | **bc** | **conform-action** | **pir**] | **cir** {*target_bit_rate* | **percent** *percentage*} | **rate** {*target_bit_rate* | **percent** *percentage*} **conform-action transmit exceed-action** {**drop** [**violate action**] | **set-cos-transmit** | **set-dscp-transmit** | **set-prec-transmit** | **transmit** [**violate action**] }}
5. **end**
6. **show policy-map**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**policy-map** *policy name*
	
	Example:
	
	`Switch(config)# policy-map policy_police01`
	`Switch(config-pmap)#`
		Enters policy map configuration mode.
		
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.

Step 3	**class** *class name*
	
	Example:
	
	`Switch(config-pmap)# class class_police01`
	`Switch(config-pmap-c)#`
		Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change. Command options for policy class map configuration mode include the following:
		
		• *word*—Class map name.
		
		• **class-default**—System default class matching any otherwise unclassified packets.

Step 4	**police** {*target_bit_rate* [*burst bytes* | **bc** | **conform-action** | **pir**] | **cir** {*target_bit_rate* | **percent** *percentage*} | **rate** {*target_bit_rate* | **percent** *percentage*} **conform-action transmit exceed-action** {**drop** [**violate action**] | **set-cos-transmit** | **set-dscp-transmit** | **set-prec-transmit** | **transmit** [**violate action**] }}
	
	Example:
	
	`Switch(config-pmap-c)# police 8000 conform-action transmit exceed-action`
		The following **police** subcommand options are available:
		
		• *target_bit_rate*—Bits per second (from 8000 to 10000000000).
		
		  ◦ *burst bytes*—Enter a value from 1000 to 512000000.
		
		  ◦ **bc**—Conform burst.
		
		  ◦ **conform-action**—Action taken when rate is less than conform burst.
		
		  ◦ **pir**—Peak Information Rate.
		
		• **cir**—Committed Information Rate.

	Command or Action	Purpose
	`drop` `Switch(config-pmap-c)#`	◦ *target_bit_rate*—Target bit rate (8000 to10000000000). ◦ **percent**—Percentage of interface bandwidth for CIR. • **rate**—Specifies the police rate, PCR for hierarchical policies, or SCR for single-level ATM 4.0 policer policies. ◦ *target_bit_rate*—Target Bit Rate (8000 to 10000000000). ◦ **percent**—Percentage of interface bandwidth for rate. The following **police conform-action transmit exceed-action** subcommand options are available: • **drop**—Drops the packet. • **set-cos-transmit**—Sets the CoS value and sends it. • **set-dscp-transmit**—Sets the DSCP value and sends it. • **set-prec-transmit**—Rewrites the packet precedence and sends it. • **transmit**—Transmits the packet. **Note** Policer-based markdown actions are only supported using table maps. Only one markdown table map is allowed for each marking field in the switch.

Step 5	**end**
	
	Example:
	`Switch(config-pmap-c)# end`
	`Switch#`
		Saves configuration changes.

Step 6	**show policy-map** **Example:** `Switch# show policy-map`	(Optional) Displays policy configuration information for all classes configured for all service policies.

What to Do Next

Configure any additional policy maps for QoS for your network. After creating your policy maps, attach the traffic policy or polices to an interface using the **service-policy** command.

Related Topics

Single-Rate Two-Color Policing, on page 256

Examples: Single-Rate Two-Color Policing Configuration, on page 320

Dual-Rate Three-Color Policing, on page 256
Examples: Dual-Rate Three-Color Policing Configuration, on page 321
Policing, on page 252
Examples: Policing Action Configuration, on page 318
Token-Bucket Algorithm, on page 252
Examples: Policing Units, on page 320

Configuring Priority

This procedure explains how to configure priority on your switch.

The switch supports giving priority to specified queues. There are two priority levels available (1 and 2).

Note Queues supporting voice and video should be assigned a priority level of 1.

Before You Begin

You should have created a class map for priority before beginning this procedure.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** *policy name*
3. **class** *class name*
4. **priority** [*Kb/s* [*burst_in_bytes*] | **level** *level_value* [*Kb/s* [*burst_in_bytes*] | **percent** *percentage* [*burst_in_bytes*]] | **percent** *percentage* [*burst_in_bytes*]]
5. **end**
6. **show policy-map**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**policy-map** *policy name*
	
	Example:
	`Switch(config)# policy-map policy_priority01`
	`Switch(config-pmap)#`
		Enters policy map configuration mode.
		
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.

	Command or Action	Purpose

Step 3	**class** *class name*
	
	Example:
	`Switch(config-pmap)# class class_priority01`
	`Switch(config-pmap-c)#`
		Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change. Command options for policy class map configuration mode include the following:
		• *word*—Class map name.
		• **class-default**—System default class matching any otherwise unclassified packets.

Step 4	**priority** [*Kb/s* [*burst_in_bytes*] | **level** *level_value* [*Kb/s* [*burst_in_bytes*] | **percent** *percentage* [*burst_in_bytes*]] | **percent** *percentage* [*burst_in_bytes*]]
	
	Example:
	`Switch(config-pmap-c)# priority level 1`
	`Switch(config-pmap-c)#`
		The **priority** command assigns a strict scheduling priority for the class.
		The command options include:
		• *Kb/s*—Specifies the kilobits per second (from 1 to 2000000).
		  ◦ *burst_in_bytes*—Specifies the burst in bytes (from 32 to 2000000).
		• **level** *level_value*—Specifies the multilevel (1-2) priority queue.
		  ◦ *Kb/s*—Specifies the kilobits per second (from 1 to 2000000).
		    ◦ *burst_in_bytes*—Specifies the burst in bytes (from 32 to 2000000).
		  ◦ **percent**—Percentage of the total bandwidth.
		    ◦ *burst_in_bytes*—Specifies the burst in bytes (from 32 to 2000000).
		• **percent**—Percentage of the total bandwidth.
		  ◦ *burst_in_bytes*—Specifies the burst in bytes (32 to 2000000).
		
		Note Priority level 1 is more important than priority level 2. Priority level 1 reserves bandwidth that is processed first for QoS, so its latency is very low. Both priority level 1 and 2 reserve bandwidth.

Step 5	**end**
	
	Example:
	`Switch(config-pmap-c)# end`
	`Switch#`
		Saves configuration changes.

Step 6	**show policy-map** **Example:** `Switch# show policy-map`	(Optional) Displays policy configuration information for all classes configured for all service policies.

What to Do Next

Configure any additional policy maps for QoS for your network. After creating your policy maps, attach the traffic policy or polices to an interface using the **service-policy** command.

Related Topics

Priority Queues, on page 260

Configuring Queues and Shaping

Configuring Egress Queue Characteristics

Depending on the complexity of your network and your QoS solution, you may need to perform all of the procedures in this section. You need to make decisions about these characteristics:

- Which packets are mapped by DSCP, CoS, or QoS group value to each queue and threshold ID?
- What drop percentage thresholds apply to the queues, and how much reserved and maximum memory is needed for the traffic type?
- How much of the fixed buffer space is allocated to the queues?
- Does the bandwidth of the port need to be rate limited?
- How often should the egress queues be serviced and which technique (shaped, shared, or both) should be used?

Note You can only configure the egress queues on the switch.

Configuring Queue Buffers

The switch allows you to allocate buffers to queues. If there is no allocation made to buffers, then they are divided equally for all queues. You can use the queue-buffer ratio to divide it in a particular ratio. Since by default DTS (Dynamic Threshold and Scaling) is active on all queues, these are soft buffers.

Note The queue-buffer ratio is supported on both wired and wireless ports, but the queue-buffer ratio cannot be configured with a queue-limit.

Before You Begin

The following are prerequisites for this procedure:

- You should have created a class map for the queue buffer before beginning this procedure.
- You must have configured either bandwidth, shape, or priority on the policy map prior to configuring the queue buffers.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** *policy name*
3. **class** *class name*
4. **bandwidth** {*Kb/s* | **percent** *percentage* | **remaining** { **ratio** *ratio value* }}
5. **queue-buffers** {**ratio** *ratio value*}
6. **end**
7. **show policy-map**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**policy-map** *policy name*
	
	Example:
	
	`Switch(config)# policy-map policy_queuebuffer01`
	`Switch(config-pmap)#`
		Enters policy map configuration mode.
		
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.

Step 3	**class** *class name*
	
	Example:
	
	`Switch(config-pmap)# class class_queuebuffer01`
	`Switch(config-pmap-c)#`
		Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change. Command options for policy class map configuration mode include the following:
		
		• *word*—Class map name.
		
		• **class-default**—System default class matching any otherwise unclassified packets.

Step 4	**bandwidth** {*Kb/s* | **percent** *percentage* | **remaining** { **ratio** *ratio value* }}
	
	Example:
	
	`Switch(config-pmap-c)# bandwidth percent 80`
	`Switch(config-pmap-c)#`
		Configures the bandwidth for the policy map. The command parameters include:
		
		• *Kb/s*—Use this command to configure a specific value. The range is 20000 to 10000000.
		
		• **percent**—Allocates a minimum bandwidth to a particular class using a percentage. The queue can oversubscribe bandwidth in case other queues do not utilize the entire port bandwidth. The total sum cannot exceed 100 percent, and in case it is less than 100 percent, the rest of the bandwidth is equally divided along all bandwidth queues.
		
		• **remaining**—Allocates a minimum bandwidth to a particular class. The queue can oversubscribe bandwidth in case other queues do not utilize

	Command or Action	Purpose
		entire port bandwidth. The total sum cannot exceed 100 percent. It is preferred to use this command when the **priority** command is used for certain queues in the policy. You can also assign ratios rather than a percentage to each queue; the queues will be assigned certain weights that are inline with these ratios. Ratios can range from 0 to 100. Total bandwidth ratio allocation for the policy in this case can exceed 100. **Note** You cannot mix bandwidth types on a policy map.
Step 5	**queue-buffers** {**ratio** *ratio value*} **Example:** `Switch(config-pmap-c)#` **`queue-buffers ratio 10`** `Switch(config-pmap-c)#`	Configures the relative buffer size for the queue. **Note** The sum of all configured buffers in a policy must be less than or equal to 100 percent. Unallocated buffers are are evenly distributed to all the remaining queues.
Step 6	**end** **Example:** `Switch(config-pmap-c)#` **`end`** `Switch#`	Saves configuration changes.
Step 7	**show policy-map** **Example:** `Switch#` **`show policy-map`**	(Optional) Displays policy configuration information for all classes configured for all service policies.

What to Do Next

Configure any additional policy maps for QoS for your network. After creating your policy maps, attach the traffic policy or polices to an interface using the **service-policy** command.

Related Topics

Queue Buffer Allocation, on page 261

Examples: Queue Buffers Configuration, on page 318

Configuring Queue Limits

You use queue limits to configure Weighted Tail Drop (WTD). WTD ensures the configuration of more than one threshold per queue. Each class of service is dropped at a different threshold value to provide for QoS differentiation. With the switch, each queue has 3 explicit programmable threshold classes—0, 1, 2. Therefore, the enqueue/drop decision of each packet per queue is determined by the packet's threshold class assignment, which is determined by the DSCP, CoS, or QoS group field of the frame header.

WTD also uses a soft limit, and therefore you are allowed to configure the queue limit to up to 400 percent (maximum four times the reserved buffer from common pool). This soft limit prevents overrunning the common pool without impacting other features.

Note You can only configure queue limits on the switch egress queues on wired ports.

Before You Begin

The following are prerequisites for this procedure:

- You should have created a class map for the queue limits before beginning this procedure.
- You must have configured either bandwidth, shape, or priority on the policy map prior to configuring the queue limits.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** *policy name*
3. **class** *class name*
4. **bandwidth** {*Kb/s* | **percent** *percentage* | **remaining** { **ratio** *ratio value* }}
5. **queue-limit** {*packets* **packets** | **cos** {*cos value* { *maximum threshold value* | **percent** *percentage* } | **values** {*cos value* | **percent** *percentage* } } | **dscp** {*dscp value* {*maximum threshold value* | **percent** *percentage*} | *match packet* {*maximum threshold value* | **percent** *percentage*} | **default** {*maximum threshold value* | **percent** *percentage*} | **ef** {*maximum threshold value* | **percent** *percentage*} | **dscp values** *dscp value*} | **percent** *percentage* }}
6. **end**
7. **show policy-map**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**policy-map** *policy name*
	
	Example:
	`Switch(config)# policy-map policy_queuelimit01`
	`Switch(config-pmap)#`
		Enters policy map configuration mode.
		
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td>Step 3</td><td>class class name

Example:
<code>Switch(config-pmap)# class class_queuelimit01</code>
<code>Switch(config-pmap-c)#</code></td><td>Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change. Command options for policy class map configuration mode include the following:
• word—Class map name.
• class-default—System default class matching any otherwise unclassified packets.</td></tr>
<tr><td>Step 4</td><td>bandwidth {Kb/s | percent percentage | remaining { ratio ratio value }}

Example:
<code>Switch(config-pmap-c)# bandwidth 500000</code>
<code>Switch(config-pmap-c)#</code></td><td>Configures the bandwidth for the policy map. The parameters include:
• Kb/s—Use this command to configure a specific value. The range is 20000 to 10000000.
• percent—Allocates a minimum bandwidth to a particular class. The queue can oversubscribe bandwidth in case other queues do not utilize the entire port bandwidth. The total sum cannot exceed 100 percent, and in case it is less than 100 percent, the rest of the bandwidth is equally divided along all bandwidth queues.
• remaining—Allocates a minimum bandwidth to a particular class. The queue can oversubscribe bandwidth in case other queues do not utilize entire port bandwidth. The total sum cannot exceed 100 percent. It is preferred to use this command when the priority command is used for certain queues in the policy. You can also assign ratios rather than a percentage to each queue; the queues will be assigned certain weights that are inline with these ratios. Ratios can range from 0 to 100. Total bandwidth ratio allocation for the policy in this case can exceed 100.

Note You cannot mix bandwidth types on a policy map.</td></tr>
<tr><td>Step 5</td><td>queue-limit {packets packets | cos {cos value { maximum threshold value | percent percentage } | values {cos value | percent percentage } } | dscp {dscp value {maximum threshold value | percent percentage} | match packet {maximum threshold value | percent percentage} | default {maximum threshold value | percent percentage} | ef {maximum threshold value | percent percentage} | dscp values dscp value} | percent percentage }}

Example:
<code>Switch(config-pmap-c)# queue-limit dscp 3 percent 20</code>
<code>Switch(config-pmap-c)# queue-limit dscp 4 percent 30</code>
<code>Switch(config-pmap-c)# queue-limit dscp</code></td><td>Sets the queue limit threshold percentage values.

With every queue, there are three thresholds (0,1,2), and there are default values for each of these thresholds. Use this command to change the default or any other queue limit threshold setting. For example, if DSCP 3, 4, and 5 packets are being sent into a specific queue in a configuration, then you can use this command to set the threshold percentages for these three DSCP values. For additional information about queue limit threshold values, see Weighted Tail Drop, on page 259.

Note The switch does not support absolute queue-limit percentages. The switch only supports DSCP or CoS queue-limit percentages.</td></tr>
</table>

	Command or Action	Purpose
	`5 percent 40`	

Step 6	**end**
	
	Example:
	
	`Switch(config-pmap-c)# end`
	`Switch#`
		Saves configuration changes.

Step 7	**show policy-map** **Example:** `Switch# show policy-map`	(Optional) Displays policy configuration information for all classes configured for all service policies.

What to Do Next

Proceed to configure any additional policy maps for QoS for your network. After creating your policy maps, proceed to attach the traffic policy or polices to an interface using the **service-policy** command.

Related Topics

Weighted Tail Drop, on page 259

Examples: Queue-limit Configuration, on page 317

Configuring Shaping

You use the **shape** command to configure shaping (maximum bandwidth) for a particular class. The queue's bandwidth is restricted to this value even though the port has additional bandwidth left. You can configure shaping as an average percent, as well as a shape average value in bits per second.

Before You Begin

You should have created a class map for shaping before beginning this procedure.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** *policy name*
3. **class** *class name*
4. **shape average** {*target bit rate* | **percent** *percentage*}
5. **end**
6. **show policy-map**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**policy-map** *policy name*
	
	Example:
	
	`Switch(config)# policy-map policy_shaping01`
	`Switch(config-pmap)#`
		Enters policy map configuration mode.
		
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.

Step 3	**class** *class name*
	
	Example:
	
	`Switch(config-pmap)# class class_shaping01`
	`Switch(config-pmap-c)#`
		Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change. Command options for policy class map configuration mode include the following:
		
		• *word*—Class map name.
		
		• **class-default**—System default class matching any otherwise unclassified packets.

Step 4	**shape average** {*target bit rate* | **percent** *percentage*}
	
	Example:
	
	`Switch(config-pmap-c)# shape average percent 50`
	`Switch(config-pmap-c)#`
		Configures the average shape rate. You can configure the average shape rate by target bit rates (bits per second) or by percentage of interface bandwidth for the Committed Information Rate (CIR).

Step 5	**end**
	
	Example:
	
	`Switch(config-pmap-c)# end`
	`Switch#`
		Saves configuration changes.

Step 6	**show policy-map** **Example:** `Switch# show policy-map`	(Optional) Displays policy configuration information for all classes configured for all service policies.

What to Do Next

Configure any additional policy maps for QoS for your network. After creating your policy maps, attach the traffic policy or polices to an interface using the **service-policy** command.

Related Topics

Average Rate Shaping, on page 257

Examples: Average Rate Shaping Configuration, on page 317

Hierarchical Shaping, on page 258

Monitoring QoS

The following commands can be used to monitor QoS on the switch.

Note Classification counters and statistics are not supported for any wireless targets.

Table 44: Monitoring QoS

Command	Description
show class-map [*class_map_name*]	Displays a list of all class maps configured.
show policy-map [*policy_map_name*]	Displays a list of all policy maps configured. Command parameters include: • **policy map name** • **interface** • **session**

Command	Description
show policy-map interface { **Auto-template** | **Capwap** | **GigabitEthernet** | **GroupVI** | **InternalInterface** | **Loopback** | **Null** | **Port-channel** | **TenGigabitEthernet** | **Tunnel** | **Vlan** | **Brief** | **class** | **input** | **output** | **wireless** }	Shows the runtime representation and statistics of all the policies configured on the switch. Command parameters include: • **Auto-template—**Auto-Template interface • **Capwap—**CAPWAP tunnel interface • **GigabitEthernet—**Gigabit Ethernet IEEE.802.3z • **GroupVI—**Group virtual interface • **InternalInterface—**Internal interface • **Loopback—**Loopback interface • **Null—**Null interface • **Port-channel—**Ethernet channel of interfaces • **TenGigabitEthernet—**10-Gigabit Ethernet • **Tunnel—**Tunnel interface • **Vlan—**Catalyst VLANs • **Brief—**Brief description of policy maps • **Class—**Show statistics for individual class • **Input—**Input policy • **Output—**Output policy • **Wireless—**wireless
show policy-map interface wireless ap [*access point*]	Shows the runtime representation and statistics for all the wireless APs on the switch.
show policy-map interface wireless ssid [*ssid*]	Shows the runtime representation and statistics for all the SSID targets on the switch.

Command	Description
show policy-map interface wireless client [*client*]	Shows the runtime representation and statistics for all the client targets on the switch.
show policy-map session [**input** | **output** | **uid** *UUID*]	Shows the session QoS policy. Command parameters include: • **input—**Input policy • **output—**Output policy • **uid—**Policy based on SSS unique identification.
show table-map	Displays all the table maps and their configurations.
show policy-map interface wireless ssid name *ssid-name* **radio type** {**24ghz** | **5ghz**} **ap name** *ap-name*	Displays SSID policy configuration on an access point.

Configuration Examples for QoS

Examples: Classification by Access Control Lists

This example shows how to classify packets for QoS by using access control lists (ACLs):

```
Switch# configure terminal
Switch(config)# access-list 101 permit ip host 12.4.1.1 host 15.2.1.1
Switch(config)# class-map acl-101
Switch(config-cmap)# description match on access-list 101
Switch(config-cmap)# match access-group 101
Switch(config-cmap)#
```

After creating a class map by using an ACL, you then create a policy map for the class, and apply the policy map to an interface for QoS.

Related Topics

Creating a Traffic Class , on page 265

Class Maps, on page 250

Examples: Class of Service Layer 2 Classification

This example shows how to classify packets for QoS using a class of service Layer 2 classification:

```
Switch# configure terminal
Switch(config)# class-map cos
Switch(config-cmap)# match cos ?
  <0-7>  Enter up to 4 class-of-service values separated by white-spaces
```

```
Switch(config-cmap)# match cos 3 4 5
Switch(config-cmap)#
```

After creating a class map by using a CoS Layer 2 classification, you then create a policy map for the class, and apply the policy map to an interface for QoS.

Examples: Class of Service DSCP Classification

This example shows how to classify packets for QoS using a class of service DSCP classification:

```
Switch# configure terminal
Switch(config)# class-map dscp
Switch(config-cmap)# match dscp af21 af22 af23
Switch(config-cmap)#
```

After creating a class map by using a DSCP classification, you then create a policy map for the class, and apply the policy map to an interface for QoS.

Examples: VLAN ID Layer 2 Classification

This example shows how to classify for QoS using a VLAN ID Layer 2 classification:

```
Switch# configure terminal
Switch(config)# class-map vlan-120
Switch(config-cmap)# match vlan ?
  <1-4095>  VLAN id
Switch(config-cmap)# match vlan 120
Switch(config-cmap)#
```

After creating a class map by using a VLAN Layer 2 classification, you then create a policy map for the class, and apply the policy map to an interface for QoS.

Examples: Classification by DSCP or Precedence Values

This example shows how to classify packets by using DSCP or precedence values:

```
Switch# configure terminal
Switch(config)# class-map prec2
Switch(config-cmap)# description matching precedence 2 packets
Switch(config-cmap)# match ip precedence 2
Switch(config-cmap)# exit
Switch(config)# class-map ef
Switch(config-cmap)# description EF traffic
Switch(config-cmap)# match ip dscp ef
Switch(config-cmap)#
```

After creating a class map by using a DSCP or precedence values, you then create a policy map for the class, and apply the policy map to an interface for QoS.

Examples: Hierarchical Classification

The following is an example of a hierarchical classification, where a class named parent is created, which matches another class named child. The class named child matches based on the IP precedence being set to 2.

```
Switch# configure terminal
```

```
Switch(config)# class-map child
Switch(config-cmap)# match ip precedence 2
Switch(config-cmap)# exit
Switch(config)# class-map parent
Switch(config-cmap)# match class child
Switch(config-cmap)#
```

After creating the parent class map, you then create a policy map for the class, and apply the policy map to an interface for QoS.

Related Topics

Hierarchical QoS, on page 244

Examples: Hierarchical Policy Configuration

The following is an example of a configuration using hierarchical polices:

```
Switch# configure terminal
Switch(config)# class-map c1
Switch(config-cmap)# match dscp 30
Switch(config-cmap)# exit

Switch(config)# class-map c2
Switch(config-cmap)# match precedence 4
Switch(config-cmap)# exit

Switch(config)# class-map c3
Switch(config-cmap)# exit

Switch(config)# policy-map child
Switch(config-pmap)# class c1
Switch(config-pmap-c)# priority level 1
Switch(config-pmap-c)# police rate percent 20 conform-action transmit exceed action drop
Switch(config-pmap-c-police)# exit
Switch(config-pmap-c)# exit

Switch(config-pmap)# class c2
Switch(config-pmap-c)# bandwidth 20000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# bandwidth 20000
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit

Switch(config)# policy-map parent
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# shape average 1000000
Switch(config-pmap-c)# service-policy child
Switch(config-pmap-c)# end
```

Examples: Classification for Voice and Video

This example describes how to classify packet streams for voice and video using switch specific information.

In this example, voice and video are coming in from end-point A into GigabitEthernet1/0/1 on the switch and have precedence values of 5 and 6, respectively. Additionally, voice and video are also coming from end-point B into GigabitEthernet1/0/2 on the switch with DSCP values of EF and AF11, respectively.

Assume that all the packets from the both the interfaces are sent on the uplink interface, and there is a requirement to police voice to 100 Mbps and video to 150 Mbps.

To classify per the above requirements, a class to match voice packets coming in on GigabitEthernet1/0/1 is created, named voice-interface-1, which matches precedence 5. Similarly another class for voice is created, named voice-interface-2, which will match voice packets in GigabitEthernet1/0/2. These classes are associated to two separate policies named input-interface-1, which is attached to GigabitEthernet1/0/1, and input-interface-2, which is attached to GigabitEthernet1/0/2. The action for this class is to mark the qos-group to 10. To match packets with QoS-group 10 on the output interface, a class named voice is created which matches on QoS-group 10. This is then associated to another policy named output-interface, which is associated to the uplink interface. Video is handled in the same way, but matches on QoS-group 20.

The following example shows how classify using the above switch specific information:

```
Switch(config)#
Switch(config)# class-map voice-interface-1
Switch(config-cmap)# match ip precedence 5
Switch(config-cmap)# exit

Switch(config)# class-map video-interface-1
Switch(config-cmap)# match ip precedence 6
Switch(config-cmap)# exit

Switch(config)# class-map voice-interface-2
Switch(config-cmap)# match ip dscp ef
Switch(config-cmap)# exit

Switch(config)# class-map video-interface-2
Switch(config-cmap)# match ip dscp af11
Switch(config-cmap)# exit

Switch(config)# policy-map input-interface-1
Switch(config-pmap)# class voice-interface-1
Switch(config-pmap-c)# set qos-group 10
Switch(config-pmap-c)# exit

Switch(config-pmap)# class video-interface-1
Switch(config-pmap-c)# set qos-group 20

Switch(config-pmap-c)# policy-map input-interface-2
Switch(config-pmap)# class voice-interface-2
Switch(config-pmap-c)# set qos-group 10
Switch(config-pmap-c)# class video-interface-2
Switch(config-pmap-c)# set qos-group 20
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit

Switch(config)# class-map voice
Switch(config-cmap)# match qos-group 10
Switch(config-cmap)# exit

Switch(config)# class-map video
Switch(config-cmap)# match qos-group 20

Switch(config)# policy-map output-interface
Switch(config-pmap)# class voice
Switch(config-pmap-c)# police 256000 conform-action transmit exceed-action drop
Switch(config-pmap-c-police)# exit
Switch(config-pmap-c)# exit

Switch(config-pmap)# class video
Switch(config-pmap-c)# police 1024000 conform-action transmit exceed-action drop
Switch(config-pmap-c-police)# exit
Switch(config-pmap-c)# exit
```

Examples: Average Rate Shaping Configuration

The following example shows how to configure average rate shaping:

```
Switch# configure terminal
Switch(config)# class-map prec1
Switch(config-cmap)# description matching precedence 1 packets
Switch(config-cmap)# match ip precedence 1
Switch(config-cmap)# end

Switch# configure terminal
Switch(config)# class-map prec2
Switch(config-cmap)# description matching precedence 2 packets
Switch(config-cmap)# match ip precedence 2
Switch(config-cmap)# exit

Switch(config)# policy-map shaper
Switch(config-pmap)# class prec1
Switch(config-pmap-c)# shape average 512000
Switch(config-pmap-c)# exit

Switch(config-pmap)# policy-map shaper
Switch(config-pmap)# class prec2
Switch(config-pmap-c)# shape average 512000
Switch(config-pmap-c)# exit

Switch(config-pmap)# class class-default
Switch(config-pmap-c)# shape average 1024000
```

After configuring the class maps, policy map, and shape averages for your configuration, proceed to then apply the policy map to the interface for QoS.

Related Topics

Configuring Shaping , on page 309

Average Rate Shaping, on page 257

Examples: Queue-limit Configuration

The following example shows how to configure a queue-limit policy based upon DSCP values and percentages:

```
Switch# configure terminal
Switch#(config)# policy-map port-queue
Switch#(config-pmap)# class dscp-1-2-3
Switch#(config-pmap-c)# bandwidth percent 20
Switch#(config-pmap-c)# queue-limit dscp 1 percent 80
Switch#(config-pmap-c)# queue-limit dscp 2 percent 90
Switch#(config-pmap-c)# queue-limit dscp 3 percent 100
Switch#(config-pmap-c)# exit

Switch#(config-pmap)# class dscp-4-5-6
Switch#(config-pmap-c)# bandwidth percent 20
Switch#(config-pmap-c)# queue-limit dscp 4 percent 20
Switch#(config-pmap-c)# queue-limit dscp 5 percent 30
Switch#(config-pmap-c)# queue-limit dscp 6 percent 20
Switch#(config-pmap-c)# exit

Switch#(config-pmap)# class dscp-7-8-9
Switch#(config-pmap-c)# bandwidth percent 20
Switch#(config-pmap-c)# queue-limit dscp 7 percent 20
Switch#(config-pmap-c)# queue-limit dscp 8 percent 30
Switch#(config-pmap-c)# queue-limit dscp 9 percent 20
Switch#(config-pmap-c)# exit
```

```
Switch#(config-pmap)# class dscp-10-11-12
Switch#(config-pmap-c)# bandwidth percent 20
Switch#(config-pmap-c)# queue-limit dscp 10 percent 20
Switch#(config-pmap-c)# queue-limit dscp 11 percent 30
Switch#(config-pmap-c)# queue-limit dscp 12 percent 20
Switch#(config-pmap-c)# exit

Switch#(config-pmap)# class dscp-13-14-15
Switch#(config-pmap-c)# bandwidth percent 10
Switch#(config-pmap-c)# queue-limit dscp 13 percent 20
Switch#(config-pmap-c)# queue-limit dscp 14 percent 30
Switch#(config-pmap-c)# queue-limit dscp 15 percent 20
Switch#(config-pmap-c)# end
Switch#
```

After finishing with the above policy map queue-limit configuration, you can then proceed to apply the policy map to an interface for QoS.

Related Topics

Configuring Queue Limits , on page 306

Weighted Tail Drop, on page 259

Examples: Queue Buffers Configuration

The following example shows how configure a queue buffer policy and then apply it to an interface for QoS:

```
Switch# configure terminal
Switch(config)# policy-map policy1001
Switch(config-pmap)# class class1001
Switch(config-pmap-c)# bandwidth remaining ratio 10
Switch(config-pmap-c)# queue-buffer ratio ?
  <0-100>  Queue-buffers ratio limit
Switch(config-pmap-c)# queue-buffer ratio 20
Switch(config-pmap-c)# end

Switch# configure terminal
Switch(config)# interface gigabitEthernet2/0/3
Switch(config-if)# service-policy output policy1001
Switch(config-if)# end
```

Related Topics

Configuring Queue Buffers , on page 304

Queue Buffer Allocation, on page 261

Examples: Policing Action Configuration

The following example displays the various policing actions that can be associated to the policer. These actions are accomplished using the conforming, exceeding, or violating packet configurations. You have the flexibility to drop, mark and transmit, or transmit packets that have exceeded or violated a traffic profile.

For example, a common deployment scenario is one where the enterprise customer polices traffic exiting the network towards the service provider and marks the conforming, exceeding and violating packets with different DSCP values. The service provider could then choose to drop the packets marked with the exceeded and violated DSCP values under cases of congestion, but may choose to transmit them when bandwidth is available.

Note The Layer 2 fields can be marked to include the CoS fields, and the Layer 3 fields can be marked to include the precedence and the DSCP fields.

One useful feature is the ability to associate multiple actions with an event. For example, you could set the precedence bit and the CoS for all conforming packets. A submode for an action configuration could then be provided by the policing feature.

This is an example of a policing action configuration:

```
Switch# configure terminal
Switch(config)# policy-map police
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# police cir 1000000 pir 2000000
Switch(config-pmap-c-police)# conform-action transmit
Switch(config-pmap-c-police)# exceed-action set-dscp-transmit dscp table exceed-markdown-table
Switch(config-pmap-c-police)# violate-action set-dscp-transmit dscp table
violate-markdown-table
Switch(config-pmap-c-police)# end
```

In this example, the exceed-markdown-table and violate-mark-down-table are table maps.

Note Policer-based markdown actions are only supported using table maps. Only one markdown table map is allowed for each marking field in the switch.

Related Topics

Configuring Police , on page 299

Policing, on page 252

Examples: Policer VLAN Configuration

The following example displays a VLAN policer configuration. At the end of this configuration, the VLAN policy map is applied to an interface for QoS.

```
Switch# configure terminal
Switch(config)# class-map vlan100
Switch(config-cmap)# match vlan 100
Switch(config-cmap)# exit
Switch(config)# policy-map vlan100
Switch(config-pmap)# policy-map class vlan100
Switch(config-pmap-c)# police 100000 bc conform-action transmit exceed-action drop
Switch(config-pmap-c-police)# end
Switch# configure terminal
Switch(config)# interface gigabitEthernet1/0/5
Switch(config-if)#  service-policy input vlan100
```

Related Topics

Classifying, Policing, and Marking Traffic on SVIs by Using Policy Maps , on page 282

Policy Map on VLANs, on page 251

Examples: Policing Units

The following examples display the various units of policing that are supported for QoS. The policing unit is the basis on which the token bucket works .

The following units of policing are supported:

- CIR and PIR are specified in bits per second. The burst parameters are specified in bytes. This is the default mode; it is the unit that is assumed when no units are specified. The CIR and PIR can also be configured in percent, in which case the burst parameters have to be configured in milliseconds.
- CIR and PIR are specified in packets per second. In this case, the burst parameters are configured in packets as well.

The following is an example of a policer configuration in bits per second:

```
Switch(config)# policy-map bps-policer
Switch(config-pmap)# class class-default
Switch(config-pmap-c) # police rate 256000 bps burst 1000 bytes
conform-action transmit exceed-action drop
```

The following is an example of a policer configuration in packets per second. In this configuration, a dual-rate three-color policer is configured where the units of measurement is packet. The burst and peak burst are all specified in packets.

```
Switch(config)# policy-map pps-policer
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# police rate 5000 pps burst 100 packets
peak-rate 10000 pps peak-burst 200 packets conform-action transmit
exceed-action drop violate-action drop
```

Related Topics

Configuring Police , on page 299

Token-Bucket Algorithm, on page 252

Examples: Single-Rate Two-Color Policing Configuration

The following example shows how to configure a single-rate two-color policer:

```
Switch(config)# class-map match-any prec1
Switch(config-cmap)# match ip precedence 1
Switch(config-cmap)# exit
Switch(config)# policy-map policer
Switch(config-pmap)# class prec1
Switch(config-pmap-c)# police cir 256000 conform-action transmit exceed-action drop
Switch(config-pmap-c-police)# exit
Switch(config-pmap-c)#
```

Related Topics

Configuring Police , on page 299

Single-Rate Two-Color Policing, on page 256

Examples: Dual-Rate Three-Color Policing Configuration

The following example shows how to configure a dual-rate three-color policer:

```
Switch# configure terminal
Switch(config)# policy-Map dual-rate-3color-policer
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# police cir 64000 bc 2000 pir 128000 be 2000
Switch(config-pmap-c-police)# conform-action transmit
Switch(config-pmap-c-police)# exceed-action set-dscp-transmit dscp table exceed-markdown-table
Switch(config-pmap-c-police)# violate-action set-dscp-transmit dscp table
violate-markdown-table
Switch(config-pmap-c-police)# exit
Switch(config-pmap-c)#
```

In this example, the exceed-markdown-table and violate-mark-down-table are table maps.

Note Policer based markdown actions are only supported using table maps. Only one markdown table map is allowed for each marking field in the switch.

Related Topics

Configuring Police , on page 299
Dual-Rate Three-Color Policing, on page 256

Examples: Table Map Marking Configuration

The following steps and examples show how to use table map marking for your QoS configuration:

1 Define the table map.

Define the table-map using the **table-map** command and indicate the mapping of the values. This table does not know of the policies or classes within which it will be used. The default command in the table map indicates the value to be copied into the 'to' field when there is no matching 'from' field. In the example, a table map named table-map1 is created. The mapping defined is to convert the value from 0 to 1 and from 2 to 3, while setting the default value to 4.

```
Switch(config)# table-map table-map1
Switch(config-tablemap)# map from 0 to 1
Switch(config-tablemap)# map from 2 to 3
Switch(config-tablemap)# default 4
Switch(config-tablemap)# exit
```

2 Define the policy map where the table map will be used.

In the example, the incoming CoS is mapped to the DSCP based on the mapping specified in the table table-map1. For this example, if the incoming packet has a DSCP of 0, the CoS in the packet is set 1. If no table map name is specified the command assumes a default behavior where the value is copied as is from the 'from' field (DSCP in this case) to the 'to' field (CoS in this case). Note however, that while the CoS is a 3-bit field, the DSCP is a 6-bit field, which implies that the CoS is copied to the first three bits in the DSCP.

```
Switch(config)# policy map policy1
```

```
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# set cos dscp table table-map1
Switch(config-pmap-c)# exit
```

3 Associate the policy to an interface.

```
Switch(config)# interface GigabitEthernet1/0/1
Switch(config-if)# service-policy output policy1
Switch(config-if)# exit
```

Related Topics

Configuring Table Maps , on page 285

Table Map Marking, on page 253

Example: Table Map Configuration to Retain CoS Markings

The following example shows how to use table maps to retain CoS markings on an interface for your QoS configuration.

The cos-trust-policy policy (configured in the example) is enabled in the ingress direction to retain the CoS marking coming into the interface. If the policy is not enabled, only the DSCP is trusted by default. If a pure Layer 2 packet arrives at the interface, then the CoS value will be rewritten to 0 when there is no such policy in the ingress port for CoS.

```
Switch# configure terminal
Switch(config)# table-map cos2cos
Switch(config-tablemap)# default copy
Switch(config-tablemap)# exit


Switch(config)# policy map cos-trust-policy
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# set cos cos table cos2cos
Switch(config-pmap-c)# exit

Switch(config)# interface GigabitEthernet1/0/2
Switch(config-if)# service-policy input cos-trust-policy
Switch(config-if)# exit
```

Where to Go Next

Review the auto-QoS documentation to see if you can use these automated capabilities for your QoS configuration.

Additional References for QoS

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	*QoS Command Reference (Catalyst 3850 Switches)* *Cisco IOS Quality of Service Solutions Command Reference*
Call Admission Control (CAC)	*System Management Configuration Guide (Catalyst 3850 Switches)* *System Management Command Reference (Catalyst 3850 Switches)*
Multicast Shaping and Policing Rate	*IP Multicast Routing Configuration Guide (Catalyst 3850 Switches)*
Precious Metal Policies	*Cisco Wireless LAN Controller Configuration Guide.*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
—	

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for QoS

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.
Cisco IOS XE 3.3SE	Consistent system default trust behavior for both wired and wireless ports. The Cisco IOS XE 3.2 Release supported different trust defaults for wired and wireless ports. The trust default for wired ports was the same as for this software release. For wireless ports, the default system behavior was non-trust, which meant that when the switch came up, all markings for the wireless ports were defaulted to zero and no traffic received priority treatment. For compatibility with an existing wired switch, all traffic went to the best-effort queue by default. The access point performed priority queuing by default. The default trust behavior in the case of wireless ports could be changed by using the **no qos wireless default untrust** command.

Release	Modification
Cisco IOS XE 3.3SE	Support for 3 radios and 11ac.
Cisco IOS XE 3.3SE	New classification counters available in the **show policy-map** command. **Note** This feature is only available on wired targets.

CHAPTER 17

Configuring Auto-QoS

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for Auto-QoS

The prerequisites for auto-QoS are the same as the prerequisites for standard QoS.

Restrictions for Auto-QoS

The following are restrictions for auto-QoS:

- Auto-qos is not supported on SVI interfaces.
- Auto-qos is not supported on interfaces which are bundled in an EtherChannel.
- The **trust device** *device_type* interface configuration command is only supported in an auto-QoS configuration, and not as a stand-alone command on the switch. When using the **trust device** *device_type* interface configuration command in an auto-QoS configuration, if the connected peer device is not a corresponding device (defined as a device matching your trust policy), both CoS and DSCP values are set to "0" and any input policy will not take effect.
- When upgrading your software release from a pre- 3.2.2 software version to a 3.2.2 or later software version, you must follow the auto-QoS upgrade procedure described in this chapter.

Information About Configuring Auto-QoS

Auto-QoS Overview

You can use the auto-QoS feature to simplify the deployment of QoS features. Auto-QoS determines the network design and enables QoS configurations so that the switch can prioritize different traffic flows.

The switch employs the MQC model. This means that instead of using certain global configurations, auto-QoS applied to any interface on a switch configures several global class maps and policy maps.

Auto-QoS matches traffic and assigns each matched packet to qos-groups. This allows the output policy map to put specific qos-groups into specific queues, including into the priority queue.

QoS is needed in both directions, both on inbound and outbound. When inbound, the switch port needs to trust the DSCP in the packet (done by default). When outbound, the switch port needs to give voice packets "front of line" priority. If voice is delayed too long by waiting behind other packets in the outbound queue, the end host drops the packet because it arrives outside of the receive window for that packet.

Auto-QoS Global Configuration Templates

In general, an auto-QoS command generates a series of class maps that either match on ACLs or on DSCP and/or CoS values to differentiate traffic into application classes. An input policy is also generated, which matches the generated classes and in some cases, polices the classes to a set bandwidth. Eight egress-queue class maps are generated. The actual egress output policy assigns a queue to each one of these eight egress-queue class maps.

The auto-QoS commands only generate templates as needed. For example, the first time any new auto-QoS command is used, global configurations that define the eight queue egress service-policy are generated. From this point on, auto-QoS commands applied to other interfaces do not generate templates for egress queuing because all auto-QoS commands rely on the same eight queue models, which have already been generated from the first time a new auto-QoS command was used.

Auto-QoS Policy and Class Maps

After entering the appropriate auto-QoS command, the following actions occur:

- Specific class maps are created.
- Specific policy maps (input and output) are created.
- Policy maps are attached to the specified interface.
- Trust level for the interface is configured.

Related Topics

Configuring Auto-QoS , on page 329
Example: auto qos trust cos, on page 332
Example: auto qos trust dscp, on page 334
Example: auto qos video cts, on page 336
Example: auto qos video ip-camera, on page 337
Example: auto qos video media-player, on page 339
Example: auto qos voip trust, on page 340
Example: auto qos voip cisco-phone, on page 342
Example: auto qos voip cisco-softphone, on page 344
Example: auto qos classify police, on page 347

Effects of Auto-QoS on Running Configuration

When auto-QoS is enabled, the **auto qos** interface configuration commands and the generated global configuration are added to the running configuration.

The switch applies the auto-QoS-generated commands as if the commands were entered from the CLI. An existing user configuration can cause the application of the generated commands to fail or to be overridden by the generated commands. These actions may occur without warning. If all the generated commands are successfully applied, any user-entered configuration that was not overridden remains in the running configuration. Any user-entered configuration that was overridden can be retrieved by reloading the switch without saving the current configuration to memory. If the generated commands are not applied, the previous running configuration is restored.

How to Configure Auto-QoS

Configuring Auto-QoS

For optimum QoS performance, configure auto-QoS on all the devices in your network.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. Depending on your auto-Qos configuration, use one of the following commands:
   - **auto qos voip** {**cisco-phone** | **cisco-softphone** | **trust**}
   - **auto qos video** {**cts** | **ip-camera** | **media-player**}
   - **auto qos classify** [**police**]
   - **auto qos trust** {**cos** | **dscp**}
4. **end**
5. **show auto qos interface** *interface-id*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 3/0/1`	Specifies the port that is connected to a VoIP port, video device, or the uplink port that is connected to another trusted switch or router in the network interior, and enters the interface configuration mode.
Step 3	Depending on your auto-Qos configuration, use one of the following commands: • **auto qos voip** {**cisco-phone** | **cisco-softphone** | **trust**} • **auto qos video** {**cts** | **ip-camera** | **media-player**} • **auto qos classify** [**police**] • **auto qos trust** {**cos** | **dscp**} **Example:** `Switch(config-if)# auto qos trust dscp`	The following commands enable auto-QoS for VoIP: • **auto qos voip cisco-phone—**If the port is connected to a Cisco IP Phone, the QoS labels of incoming packets are only trusted (conditional trust through CDP) when the telephone is detected. • **auto qos voip cisco-softphone—**The port is connected to device running the Cisco SoftPhone feature. This command generates a QoS configuration for interfaces connected to PCs running the Cisco IP SoftPhone application and mark, as well as police traffic coming from such interfaces. Ports configured with this command are considered untrusted. • **auto qos voip trust—**The uplink port is connected to a trusted switch or router, and the VoIP traffic classification in the ingress packet is trusted. The following commands enable auto-QoS for the specified video device (system, camera, or media player):

	Command or Action	Purpose
		• **auto qos video cts—**A port connected to a Cisco Telepresence system. QoS labels of incoming packets are only trusted (conditional trust through CDP) when a Cisco TelePresence is detected. • **auto qos video ip-camera—**A port connected to a Cisco video surveillance camera. QoS labels of incoming packets are only trusted (conditional trust through CDP) when a Cisco camera is detected. • **auto qos video media-player—**A port connected to a CDP-capable Cisco digital media player. QoS labels of incoming packets are only trusted (conditional trust through CDP) when a digital media player is detected. The following command enables auto-QoS for classification: • **auto qos classify police—** This command generates a QoS configuration for untrusted interfaces. The configuration places a service-policy on the interface to classify traffic coming from untrusted desktops/devices and mark them accordingly. The service-policies generated do police. The following commands enable auto-QoS for trusted interfaces: • **auto qos trust cos—**Class of service. • **auto qos trust dscp—**Differentiated Services Code Point. • <cr>—Trust interface.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show auto qos interface** *interface-id* **Example:** `Switch# show auto qos interface gigabitethernet 3/0/1`	(Optional) Verifies your entries. This command displays the auto-QoS command on the interface on which auto-QoS was enabled. You can use the **show running-config** privileged EXEC command to display the auto-QoS configuration and the user modifications.

Related Topics

Auto-QoS Policy and Class Maps, on page 329
Example: auto qos trust cos, on page 332
Example: auto qos trust dscp, on page 334
Example: auto qos video cts, on page 336
Example: auto qos video ip-camera, on page 337

Monitoring Auto-QoS

Table 45: Commands for Monitoring Auto-QoS

Command	Description
show auto qos [**interface** [*interface-id*]]	Displays the initial auto-QoS configuration. You can compare the **show auto qos** and the **show running-config** command output to identify the user-defined QoS settings.
show running-config	Displays information about the QoS configuration that might be affected by auto-QoS. You can compare the **show auto qos** and the **show running-config** command output to identify the user-defined QoS settings.

Troubleshooting Auto-QoS

To troubleshoot auto-QoS, use the **debug auto qos** privileged EXEC command. For more information, see the **debug auto qos** command in the command reference for this release.

To disable auto-QoS on a port, use the **no** form of the **auto qos** command interface configuration command, such as **no auto qos voip**. Only the auto-QoS-generated interface configuration commands for this port are removed. If this is the last port on which auto-QoS is enabled and you enter the **no auto qos voip** command, auto-QoS is considered disabled even though the auto-QoS-generated global configuration commands remain (to avoid disrupting traffic on other ports affected by the global configuration).

Configuration Examples for Auto-QoS

Example: auto qos trust cos

The following is an example of the **auto qos trust cos** command and the applied policies and class maps.

```
Switch(config)# interface gigabitEthernet1/0/30
Switch(config-if)# auto qos trust cos
Switch(config-if)# end
Switch# show policy-map interface GigabitEthernet1/0/30
```

```
GigabitEthernet1/0/30

 Service-policy input: AutoQos-4.0-Trust-Cos-Input-Policy

   Class-map: class-default (match-any)
     Match: any
     QoS Set
       cos cos table AutoQos-4.0-Trust-Cos-Table

 Service-policy output: AutoQos-4.0-Output-Policy

   queue stats for all priority classes:
     Queueing
     priority level 1

     (total drops) 0
     (bytes output) 0

   Class-map: AutoQos-4.0-Output-Priority-Queue (match-any)
     Match:  dscp cs4 (32) cs5 (40) ef (46)
     Match: cos  5
     Priority: 30% (300000 kbps), burst bytes 7500000,

     Priority Level: 1

   Class-map: AutoQos-4.0-Output-Control-Mgmt-Queue (match-any)
     Match:  dscp cs3 (24) cs6 (48) cs7 (56)
     Match: cos  3
     Queueing
     queue-limit dscp 16 percent 80
     queue-limit dscp 24 percent 90
     queue-limit dscp 48 percent 100

     (total drops) 0
     (bytes output) 0
     bandwidth remaining 10%

     queue-buffers ratio 10

   Class-map: AutoQos-4.0-Output-Multimedia-Conf-Queue (match-any)
     Match:  dscp af41 (34) af42 (36) af43 (38)
     Match: cos  4
     Queueing

     (total drops) 0
     (bytes output) 0
     bandwidth remaining 10%
     queue-buffers ratio 10

   Class-map: AutoQos-4.0-Output-Trans-Data-Queue (match-any)
     Match:  dscp af21 (18) af22 (20) af23 (22)
     Match: cos  2
     Queueing

     (total drops) 0
     (bytes output) 0
     bandwidth remaining 10%
     queue-buffers ratio 10

   Class-map: AutoQos-4.0-Output-Bulk-Data-Queue (match-any)
     Match:  dscp af11 (10) af12 (12) af13 (14)
     Match: cos  1
     Queueing

     (total drops) 0
     (bytes output) 0
     bandwidth remaining 4%
     queue-buffers ratio 10

   Class-map: AutoQos-4.0-Output-Scavenger-Queue (match-any)
     Match:  dscp cs1 (8)
     Queueing
```

```
      (total drops) 0
      (bytes output) 0
      bandwidth remaining 1%
      queue-buffers ratio 10

    Class-map: AutoQos-4.0-Output-Multimedia-Strm-Queue (match-any)
      Match:  dscp af31 (26) af32 (28) af33 (30)
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 10%
      queue-buffers ratio 10

    Class-map: class-default (match-any)
      Match: any
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 25%
      queue-buffers ratio 25
```

Related Topics

Configuring Auto-QoS , on page 329

Auto-QoS Policy and Class Maps, on page 329

Example: auto qos trust dscp

The following is an example of the **auto qos trust dscp** command and the applied policies and class maps.

```
Switch(config)# interface GigabitEthernet1/0/32
Switch(config-if)# auto qos trust dscp
Switch(config-if)# end
Switch#show policy-map interface GigabitEthernet1/0/32

 GigabitEthernet1/0/32

  Service-policy input: AutoQos-4.0-Trust-Dscp-Input-Policy

    Class-map: class-default (match-any)
      Match: any
      QoS Set
        dscp dscp table AutoQos-4.0-Trust-Dscp-Table

  Service-policy output: AutoQos-4.0-Output-Policy

    queue stats for all priority classes:
      Queueing
      priority level 1

      (total drops) 0
      (bytes output) 0

    Class-map: AutoQos-4.0-Output-Priority-Queue (match-any)
      Match:  dscp cs4 (32) cs5 (40) ef (46)
      Match: cos  5
      Priority: 30% (300000 kbps), burst bytes 7500000,

      Priority Level: 1

    Class-map: AutoQos-4.0-Output-Control-Mgmt-Queue (match-any)
      Match:  dscp cs3 (24) cs6 (48) cs7 (56)
      Match: cos  3
      Queueing
```

```
    queue-limit dscp 16 percent 80
    queue-limit dscp 24 percent 90
    queue-limit dscp 48 percent 100

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 10%

    queue-buffers ratio 10

  Class-map: AutoQos-4.0-Output-Multimedia-Conf-Queue (match-any)
    Match:  dscp af41 (34) af42 (36) af43 (38)
    Match: cos  4
    Queueing

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 10%
    queue-buffers ratio 10

  Class-map: AutoQos-4.0-Output-Trans-Data-Queue (match-any)
    Match:  dscp af21 (18) af22 (20) af23 (22)
    Match: cos  2
    Queueing

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 10%
    queue-buffers ratio 10

  Class-map: AutoQos-4.0-Output-Bulk-Data-Queue (match-any)
    Match:  dscp af11 (10) af12 (12) af13 (14)
    Match: cos  1
    Queueing

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 4%
    queue-buffers ratio 10

  Class-map: AutoQos-4.0-Output-Scavenger-Queue (match-any)
    Match:  dscp cs1 (8)
    Queueing

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 1%
    queue-buffers ratio 10

  Class-map: AutoQos-4.0-Output-Multimedia-Strm-Queue (match-any)
    Match:  dscp af31 (26) af32 (28) af33 (30)
    Queueing

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 10%
    queue-buffers ratio 10

  Class-map: class-default (match-any)
    Match: any
    Queueing

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 25%
    queue-buffers ratio 25
```

Related Topics

Configuring Auto-QoS , on page 329

Auto-QoS Policy and Class Maps, on page 329

Example: auto qos video cts

The following is an example of the **auto qos video cts** command and the applied policies and class maps.

```
Switch(config)# interface gigabitEthernet1/0/33
Switch(config-if)# auto qos video cts
Switch(config-if)# end
Switch# show policy-map interface gigabitEthernet1/0/33


GigabitEthernet1/0/33

  Service-policy input: AutoQos-4.0-Trust-Cos-Input-Policy

    Class-map: class-default (match-any)
      Match: any
      QoS Set
        cos cos table AutoQos-4.0-Trust-Cos-Table

  Service-policy output: AutoQos-4.0-Output-Policy

    queue stats for all priority classes:
      Queueing
      priority level 1

      (total drops) 0
      (bytes output) 0

    Class-map: AutoQos-4.0-Output-Priority-Queue (match-any)
      Match:  dscp cs4 (32) cs5 (40) ef (46)
      Match: cos  5
      Priority: 30% (300000 kbps), burst bytes 7500000,

      Priority Level: 1

    Class-map: AutoQos-4.0-Output-Control-Mgmt-Queue (match-any)
      Match:  dscp cs3 (24) cs6 (48) cs7 (56)
      Match: cos  3
      Queueing
      queue-limit dscp 16 percent 80
      queue-limit dscp 24 percent 90
      queue-limit dscp 48 percent 100

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 10%

      queue-buffers ratio 10

    Class-map: AutoQos-4.0-Output-Multimedia-Conf-Queue (match-any)
      Match:  dscp af41 (34) af42 (36) af43 (38)
      Match: cos  4
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 10%
      queue-buffers ratio 10

    Class-map: AutoQos-4.0-Output-Trans-Data-Queue (match-any)
      Match:  dscp af21 (18) af22 (20) af23 (22)
      Match: cos  2
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 10%
      queue-buffers ratio 10

    Class-map: AutoQos-4.0-Output-Bulk-Data-Queue (match-any)
```

```
      Match:  dscp af11 (10) af12 (12) af13 (14)
      Match: cos  1
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 4%
      queue-buffers ratio 10

    Class-map: AutoQos-4.0-Output-Scavenger-Queue (match-any)
      Match:  dscp cs1 (8)
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 1%
      queue-buffers ratio 10

    Class-map: AutoQos-4.0-Output-Multimedia-Strm-Queue (match-any)
      Match:  dscp af31 (26) af32 (28) af33 (30)
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 10%
      queue-buffers ratio 10

    Class-map: class-default (match-any)
      Match: any
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 25%
      queue-buffers ratio 25
```

Related Topics

Configuring Auto-QoS , on page 329

Auto-QoS Policy and Class Maps, on page 329

Example: auto qos video ip-camera

The following is an example of the **auto qos video ip-camera** command and the applied policies and class maps.

```
Switch(config)# interface GigabitEthernet1/0/34
Switch(config-if)# auto qos video ip-camera
Switch(config-if)# end
Switch# show policy-map interface GigabitEthernet1/0/34

 GigabitEthernet1/0/34

  Service-policy input: AutoQos-4.0-Trust-Dscp-Input-Policy

    Class-map: class-default (match-any)
      Match: any
      QoS Set
        dscp dscp table AutoQos-4.0-Trust-Dscp-Table

  Service-policy output: AutoQos-4.0-Output-Policy

    queue stats for all priority classes:
      Queueing
      priority level 1
```

```
    (total drops) 0
    (bytes output) 0

  Class-map: AutoQos-4.0-Output-Priority-Queue (match-any)
    Match:  dscp cs4 (32) cs5 (40) ef (46)
    Match: cos  5
    Priority: 30% (300000 kbps), burst bytes 7500000,

    Priority Level: 1

  Class-map: AutoQos-4.0-Output-Control-Mgmt-Queue (match-any)
    Match:  dscp cs3 (24) cs6 (48) cs7 (56)
    Match: cos  3
    Queueing
    queue-limit dscp 16 percent 80
    queue-limit dscp 24 percent 90
    queue-limit dscp 48 percent 100

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 10%

    queue-buffers ratio 10

  Class-map: AutoQos-4.0-Output-Multimedia-Conf-Queue (match-any)
    Match:  dscp af41 (34) af42 (36) af43 (38)
    Match: cos  4
    Queueing

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 10%
    queue-buffers ratio 10

  Class-map: AutoQos-4.0-Output-Trans-Data-Queue (match-any)
    Match:  dscp af21 (18) af22 (20) af23 (22)
    Match: cos  2
    Queueing

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 10%
    queue-buffers ratio 10

  Class-map: AutoQos-4.0-Output-Bulk-Data-Queue (match-any)
    Match:  dscp af11 (10) af12 (12) af13 (14)
    Match: cos  1
    Queueing

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 4%
    queue-buffers ratio 10

  Class-map: AutoQos-4.0-Output-Scavenger-Queue (match-any)
    Match:  dscp cs1 (8)
    Queueing

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 1%
    queue-buffers ratio 10

  Class-map: AutoQos-4.0-Output-Multimedia-Strm-Queue (match-any)
    Match:  dscp af31 (26) af32 (28) af33 (30)
    Queueing

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 10%
    queue-buffers ratio 10

  Class-map: class-default (match-any)
```

```
      Match: any
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 25%
      queue-buffers ratio 25
```

Related Topics

Configuring Auto-QoS , on page 329

Auto-QoS Policy and Class Maps, on page 329

Example: auto qos video media-player

The following is an example of the **auto qos video media-player** command and the applied policies and class maps.

```
Switch(config)# interface GigabitEthernet1/0/35
Switch(config-if)# auto qos video media-player
Switch(config-if)# end
Switch# show policy-map interface GigabitEthernet1/0/35

GigabitEthernet1/0/35

  Service-policy input: AutoQos-4.0-Trust-Dscp-Input-Policy

    Class-map: class-default (match-any)
      Match: any
      QoS Set
        dscp dscp table AutoQos-4.0-Trust-Dscp-Table

  Service-policy output: AutoQos-4.0-Output-Policy

    queue stats for all priority classes:
      Queueing
      priority level 1

      (total drops) 0
      (bytes output) 0

    Class-map: AutoQos-4.0-Output-Priority-Queue (match-any)
      Match:  dscp cs4 (32) cs5 (40) ef (46)
      Match: cos  5
      Priority: 30% (300000 kbps), burst bytes 7500000,

      Priority Level: 1

    Class-map: AutoQos-4.0-Output-Control-Mgmt-Queue (match-any)
      Match:  dscp cs3 (24) cs6 (48) cs7 (56)
      Match: cos  3
      Queueing
      queue-limit dscp 16 percent 80
      queue-limit dscp 24 percent 90
      queue-limit dscp 48 percent 100

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 10%

      queue-buffers ratio 10

    Class-map: AutoQos-4.0-Output-Multimedia-Conf-Queue (match-any)
      Match:  dscp af41 (34) af42 (36) af43 (38)
      Match: cos  4
      Queueing
```

```
      (total drops) 0
      (bytes output) 0
      bandwidth remaining 10%
      queue-buffers ratio 10

    Class-map: AutoQos-4.0-Output-Trans-Data-Queue (match-any)
      Match:  dscp af21 (18) af22 (20) af23 (22)
      Match: cos  2
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 10%
      queue-buffers ratio 10

    Class-map: AutoQos-4.0-Output-Bulk-Data-Queue (match-any)
      Match:  dscp af11 (10) af12 (12) af13 (14)
      Match: cos  1
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 4%
      queue-buffers ratio 10

    Class-map: AutoQos-4.0-Output-Scavenger-Queue (match-any)
      Match:  dscp cs1 (8)
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 1%
      queue-buffers ratio 10

    Class-map: AutoQos-4.0-Output-Multimedia-Strm-Queue (match-any)
      Match:  dscp af31 (26) af32 (28) af33 (30)
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 10%
      queue-buffers ratio 10

    Class-map: class-default (match-any)
      Match: any
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 25%
      queue-buffers ratio 25
```

Related Topics

Configuring Auto-QoS , on page 329

Auto-QoS Policy and Class Maps, on page 329

Example: auto qos voip trust

The following is an example of the **auto qos voip trust** command and the applied policies and class maps.

```
Switch(config)# interface gigabitEthernet1/0/36
Switch(config-if)# auto qos voip trust
Switch(config-if)# end
Switch# show policy-map interface GigabitEthernet1/0/36

GigabitEthernet1/0/36
```

```
Service-policy input: AutoQos-4.0-Trust-Cos-Input-Policy

  Class-map: class-default (match-any)
    Match: any
    QoS Set
      cos cos table AutoQos-4.0-Trust-Cos-Table

Service-policy output: AutoQos-4.0-Output-Policy

  queue stats for all priority classes:
    Queueing
    priority level 1

    (total drops) 0
    (bytes output) 0

  Class-map: AutoQos-4.0-Output-Priority-Queue (match-any)
    Match:  dscp cs4 (32) cs5 (40) ef (46)
    Match: cos  5
    Priority: 30% (300000 kbps), burst bytes 7500000,

    Priority Level: 1

  Class-map: AutoQos-4.0-Output-Control-Mgmt-Queue (match-any)
    Match:  dscp cs3 (24) cs6 (48) cs7 (56)
    Match: cos  3
    Queueing
    queue-limit dscp 16 percent 80
    queue-limit dscp 24 percent 90
    queue-limit dscp 48 percent 100

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 10%

    queue-buffers ratio 10

  Class-map: AutoQos-4.0-Output-Multimedia-Conf-Queue (match-any)
    Match:  dscp af41 (34) af42 (36) af43 (38)
    Match: cos  4
    Queueing

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 10%
    queue-buffers ratio 10

  Class-map: AutoQos-4.0-Output-Trans-Data-Queue (match-any)
    Match:  dscp af21 (18) af22 (20) af23 (22)
    Match: cos  2
    Queueing

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 10%
    queue-buffers ratio 10

  Class-map: AutoQos-4.0-Output-Bulk-Data-Queue (match-any)
    Match:  dscp af11 (10) af12 (12) af13 (14)
    Match: cos  1
    Queueing

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 4%
    queue-buffers ratio 10

  Class-map: AutoQos-4.0-Output-Scavenger-Queue (match-any)
    Match:  dscp cs1 (8)
    Queueing

    (total drops) 0
```

```
      (bytes output) 0
      bandwidth remaining 1%
      queue-buffers ratio 10

    Class-map: AutoQos-4.0-Output-Multimedia-Strm-Queue (match-any)
      Match:  dscp af31 (26) af32 (28) af33 (30)
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 10%
      queue-buffers ratio 10

    Class-map: class-default (match-any)
      Match: any
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 25%
      queue-buffers ratio 25
```

Related Topics

Configuring Auto-QoS , on page 329

Auto-QoS Policy and Class Maps, on page 329

Example: auto qos voip cisco-phone

The following is an example of the **auto qos voip cisco-phone** command and the applied policies and class maps.

```
Switch(config)# interface gigabitEthernet1/0/37
Switch(config-if)# auto qos voip cisco-phone
Switch(config-if)# end
Switch# show policy-map interface gigabitEthernet1/0/37

GigabitEthernet1/0/37

  Service-policy input: AutoQos-4.0-CiscoPhone-Input-Policy

    Class-map: AutoQos-4.0-Voip-Data-Class (match-any)
      Match: ip dscp ef (46)
      QoS Set
        ip dscp ef
      police:
          cir 128000 bps, bc 8000 bytes, be 8000 bytes
        conformed 0 bytes; actions:
          transmit
        exceeded 0 bytes; actions:
          set-dscp-transmit dscp table policed-dscp
        violated 0 bytes; actions:
          drop
        conformed 0000 bps, exceed 0000 bps, violate 0000 bps

    Class-map: AutoQos-4.0-Voip-Signal-Class (match-any)
      Match: ip dscp cs3 (24)
      QoS Set
        ip dscp cs3
      police:
          cir 32000 bps, bc 8000 bytes, be 8000 bytes
        conformed 0 bytes; actions:
          transmit
        exceeded 0 bytes; actions:
          set-dscp-transmit dscp table policed-dscp
        violated 0 bytes; actions:
          drop
```

```
        conformed 0000 bps, exceed 0000 bps, violate 0000 bps

    Class-map: AutoQos-4.0-Default-Class (match-any)
      Match: access-group name AutoQos-4.0-Acl-Default
      QoS Set
        dscp default
      police:
          cir 10000000 bps, bc 8000 bytes, be 8000 bytes
        conformed 0 bytes; actions:
          transmit
        exceeded 0 bytes; actions:
          set-dscp-transmit dscp table policed-dscp
        violated 0 bytes; actions:
          drop
        conformed 0000 bps, exceed 0000 bps, violate 0000 bps

    Class-map: class-default (match-any)
      Match: any

  Service-policy output: AutoQos-4.0-Output-Policy

    queue stats for all priority classes:
      Queueing
      priority level 1

      (total drops) 0
      (bytes output) 0

    Class-map: AutoQos-4.0-Output-Priority-Queue (match-any)
      Match:  dscp cs4 (32) cs5 (40) ef (46)
      Match: cos  5
      Priority: 30% (300000 kbps), burst bytes 7500000,

      Priority Level: 1

    Class-map: AutoQos-4.0-Output-Control-Mgmt-Queue (match-any)
      Match:  dscp cs3 (24) cs6 (48) cs7 (56)
      Match: cos  3
      Queueing
      queue-limit dscp 16 percent 80
      queue-limit dscp 24 percent 90
      queue-limit dscp 48 percent 100

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 10%

      queue-buffers ratio 10

    Class-map: AutoQos-4.0-Output-Multimedia-Conf-Queue (match-any)
      Match:  dscp af41 (34) af42 (36) af43 (38)
      Match: cos  4
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 10%
      queue-buffers ratio 10

    Class-map: AutoQos-4.0-Output-Trans-Data-Queue (match-any)
      Match:  dscp af21 (18) af22 (20) af23 (22)
      Match: cos  2
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 10%
      queue-buffers ratio 10

    Class-map: AutoQos-4.0-Output-Bulk-Data-Queue (match-any)
      Match:  dscp af11 (10) af12 (12) af13 (14)
      Match: cos  1
      Queueing
```

```
        (total drops) 0
        (bytes output) 0
        bandwidth remaining 4%
        queue-buffers ratio 10

      Class-map: AutoQos-4.0-Output-Scavenger-Queue (match-any)
        Match:  dscp cs1 (8)
        Queueing

        (total drops) 0
        (bytes output) 0
        bandwidth remaining 1%
        queue-buffers ratio 10

      Class-map: AutoQos-4.0-Output-Multimedia-Strm-Queue (match-any)
        Match:  dscp af31 (26) af32 (28) af33 (30)
        Queueing

        (total drops) 0
        (bytes output) 0
        bandwidth remaining 10%
        queue-buffers ratio 10

      Class-map: class-default (match-any)
        Match: any
        Queueing

        (total drops) 0
        (bytes output) 0
        bandwidth remaining 25%
        queue-buffers ratio 25
```

Related Topics

Configuring Auto-QoS , on page 329

Auto-QoS Policy and Class Maps, on page 329

Example: auto qos voip cisco-softphone

The following is an example of the **auto qos voip cisco-softphone** command and the applied policies and class maps.

```
Switch(config)# interface gigabitEthernet1/0/38
Switch(config-if)# auto qos voip cisco-softphone
Switch(config-if)# end
Switch# show policy-map interface gigabitEthernet1/0/38

GigabitEthernet1/0/38

  Service-policy input: AutoQos-4.0-CiscoSoftPhone-Input-Policy

    Class-map: AutoQos-4.0-Voip-Data-Class (match-any)
      Match: ip dscp ef (46)
      QoS Set
        ip dscp ef
      police:
          cir 128000 bps, bc 8000 bytes, be 8000 bytes
        conformed 0 bytes; actions:
          transmit
        exceeded 0 bytes; actions:
          set-dscp-transmit dscp table policed-dscp
        violated 0 bytes; actions:
          drop
        conformed 0000 bps, exceed 0000 bps, violate 0000 bps

    Class-map: AutoQos-4.0-Voip-Signal-Class (match-any)
```

```
  Match: ip dscp cs3 (24)
  QoS Set
    ip dscp cs3
  police:
      cir 32000 bps, bc 8000 bytes, be 8000 bytes
    conformed 0 bytes; actions:
      transmit
    exceeded 0 bytes; actions:
      set-dscp-transmit dscp table policed-dscp
    violated 0 bytes; actions:
      drop
    conformed 0000 bps, exceed 0000 bps, violate 0000 bps

Class-map: AutoQos-4.0-Multimedia-Conf-Class (match-any)
  Match: access-group name AutoQos-4.0-Acl-MultiEnhanced-Conf
  QoS Set
    dscp af41
  police:
      cir 5000000 bps, bc 8000 bytes, be 8000 bytes
    conformed 0 bytes; actions:
      transmit
    exceeded 0 bytes; actions:
      set-dscp-transmit dscp table policed-dscp
    violated 0 bytes; actions:
      drop
    conformed 0000 bps, exceed 0000 bps, violate 0000 bps

Class-map: AutoQos-4.0-Bulk-Data-Class (match-any)
  Match: access-group name AutoQos-4.0-Acl-Bulk-Data
  QoS Set
    dscp af11
  police:
      cir 10000000 bps, bc 8000 bytes, be 8000 bytes
    conformed 0 bytes; actions:
      transmit
    exceeded 0 bytes; actions:
      set-dscp-transmit dscp table policed-dscp
    violated 0 bytes; actions:
      drop
    conformed 0000 bps, exceed 0000 bps, violate 0000 bps

Class-map: AutoQos-4.0-Transaction-Class (match-any)
  Match: access-group name AutoQos-4.0-Acl-Transactional-Data
  QoS Set
    dscp af21
  police:
      cir 10000000 bps, bc 8000 bytes, be 8000 bytes
    conformed 0 bytes; actions:
      transmit
    exceeded 0 bytes; actions:
      set-dscp-transmit dscp table policed-dscp
    violated 0 bytes; actions:
      drop
    conformed 0000 bps, exceed 0000 bps, violate 0000 bps

Class-map: AutoQos-4.0-Scavanger-Class (match-any)
  Match: access-group name AutoQos-4.0-Acl-Scavanger
  QoS Set
    dscp cs1
  police:
      cir 10000000 bps, bc 8000 bytes, be 8000 bytes
    conformed 0 bytes; actions:
      transmit
    exceeded 0 bytes; actions:
      set-dscp-transmit dscp table policed-dscp
    violated 0 bytes; actions:
      drop
    conformed 0000 bps, exceed 0000 bps, violate 0000 bps

Class-map: AutoQos-4.0-Signaling-Class (match-any)
  Match: access-group name AutoQos-4.0-Acl-Signaling
  QoS Set
    dscp cs3
```

```
    police:
        cir 32000 bps, bc 8000 bytes, be 8000 bytes
      conformed 0 bytes; actions:
        transmit
      exceeded 0 bytes; actions:
        set-dscp-transmit dscp table policed-dscp
      violated 0 bytes; actions:
        drop
      conformed 0000 bps, exceed 0000 bps, violate 0000 bps

  Class-map: AutoQos-4.0-Default-Class (match-any)
    Match: access-group name AutoQos-4.0-Acl-Default
    QoS Set
      dscp default
    police:
        cir 10000000 bps, bc 8000 bytes, be 8000 bytes
      conformed 0 bytes; actions:
        transmit
      exceeded 0 bytes; actions:
        set-dscp-transmit dscp table policed-dscp
      violated 0 bytes; actions:
        drop
      conformed 0000 bps, exceed 0000 bps, violate 0000 bps

  Class-map: class-default (match-any)
    Match: any

Service-policy output: AutoQos-4.0-Output-Policy

  queue stats for all priority classes:
    Queueing
    priority level 1

    (total drops) 0
    (bytes output) 0

  Class-map: AutoQos-4.0-Output-Priority-Queue (match-any)
    Match:  dscp cs4 (32) cs5 (40) ef (46)
    Match: cos  5
    Priority: 30% (300000 kbps), burst bytes 7500000,

    Priority Level: 1

  Class-map: AutoQos-4.0-Output-Control-Mgmt-Queue (match-any)
    Match:  dscp cs3 (24) cs6 (48) cs7 (56)
    Match: cos  3
    Queueing
    queue-limit dscp 16 percent 80
    queue-limit dscp 24 percent 90
    queue-limit dscp 48 percent 100

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 10%

    queue-buffers ratio 10

  Class-map: AutoQos-4.0-Output-Multimedia-Conf-Queue (match-any)
    Match:  dscp af41 (34) af42 (36) af43 (38)
    Match: cos  4
    Queueing

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 10%
    queue-buffers ratio 10

  Class-map: AutoQos-4.0-Output-Trans-Data-Queue (match-any)
    Match:  dscp af21 (18) af22 (20) af23 (22)
    Match: cos  2
    Queueing

    (total drops) 0
```

```
      (bytes output) 0
      bandwidth remaining 10%
      queue-buffers ratio 10

    Class-map: AutoQos-4.0-Output-Bulk-Data-Queue (match-any)
      Match:  dscp af11 (10) af12 (12) af13 (14)
      Match: cos  1
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 4%
      queue-buffers ratio 10

    Class-map: AutoQos-4.0-Output-Scavenger-Queue (match-any)
      Match:  dscp cs1  (8)
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 1%
      queue-buffers ratio 10

    Class-map: AutoQos-4.0-Output-Multimedia-Strm-Queue (match-any)
      Match:  dscp af31 (26) af32 (28) af33 (30)
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 10%
      queue-buffers ratio 10

    Class-map: class-default (match-any)
      Match: any
      Queueing

      (total drops) 0
      (bytes output) 0
      bandwidth remaining 25%
      queue-buffers ratio 25
```

Related Topics

Configuring Auto-QoS , on page 329

Auto-QoS Policy and Class Maps, on page 329

Example: auto qos classify police

The following is an example of the **auto qos classify police** command and the applied policies and class maps.

```
Switch(config)# interface gigabitEthernet1/0/39
Switch(config-if)# auto qos classify police
Switch(config-if)# end
Switch# show policy-map interface gigabitEthernet1/0/39

GigabitEthernet1/0/39

  Service-policy input: AutoQos-4.0-Classify-Police-Input-Policy

    Class-map: AutoQos-4.0-Multimedia-Conf-Class (match-any)
      Match: access-group name AutoQos-4.0-Acl-MultiEnhanced-Conf
      QoS Set
        dscp af41
      police:
          cir 5000000 bps, bc 8000 bytes, be 8000 bytes
        conformed 0 bytes; actions:
          transmit
```

```
      exceeded 0 bytes; actions:
        set-dscp-transmit dscp table policed-dscp
      violated 0 bytes; actions:
        drop
      conformed 0000 bps, exceed 0000 bps, violate 0000 bps

  Class-map: AutoQos-4.0-Bulk-Data-Class (match-any)
    Match: access-group name AutoQos-4.0-Acl-Bulk-Data
    QoS Set
      dscp af11
    police:
        cir 10000000 bps, bc 8000 bytes, be 8000 bytes
      conformed 0 bytes; actions:
        transmit
      exceeded 0 bytes; actions:
        set-dscp-transmit dscp table policed-dscp
      violated 0 bytes; actions:
        drop
      conformed 0000 bps, exceed 0000 bps, violate 0000 bps

  Class-map: AutoQos-4.0-Transaction-Class (match-any)
    Match: access-group name AutoQos-4.0-Acl-Transactional-Data
    QoS Set
      dscp af21
    police:
        cir 10000000 bps, bc 8000 bytes, be 8000 bytes
      conformed 0 bytes; actions:
        transmit
      exceeded 0 bytes; actions:
        set-dscp-transmit dscp table policed-dscp
      violated 0 bytes; actions:
        drop
      conformed 0000 bps, exceed 0000 bps, violate 0000 bps

  Class-map: AutoQos-4.0-Scavanger-Class (match-any)
    Match: access-group name AutoQos-4.0-Acl-Scavanger
    QoS Set
      dscp cs1
    police:
        cir 10000000 bps, bc 8000 bytes, be 8000 bytes
      conformed 0 bytes; actions:
        transmit
      exceeded 0 bytes; actions:
        set-dscp-transmit dscp table policed-dscp
      violated 0 bytes; actions:
        drop
      conformed 0000 bps, exceed 0000 bps, violate 0000 bps

  Class-map: AutoQos-4.0-Signaling-Class (match-any)
    Match: access-group name AutoQos-4.0-Acl-Signaling
    QoS Set
      dscp cs3
    police:
        cir 32000 bps, bc 8000 bytes, be 8000 bytes
      conformed 0 bytes; actions:
        transmit
      exceeded 0 bytes; actions:
        set-dscp-transmit dscp table policed-dscp
      violated 0 bytes; actions:
        drop
      conformed 0000 bps, exceed 0000 bps, violate 0000 bps

  Class-map: AutoQos-4.0-Default-Class (match-any)
    Match: access-group name AutoQos-4.0-Acl-Default
    QoS Set
      dscp default
    police:
        cir 10000000 bps, bc 8000 bytes, be 8000 bytes
      conformed 0 bytes; actions:
        transmit
      exceeded 0 bytes; actions:
        set-dscp-transmit dscp table policed-dscp
      violated 0 bytes; actions:
```

```
        drop
      conformed 0000 bps, exceed 0000 bps, violate 0000 bps

  Class-map: class-default (match-any)
    Match: any

Service-policy output: AutoQos-4.0-Output-Policy

  queue stats for all priority classes:
    Queueing
    priority level 1

    (total drops) 0
    (bytes output) 0

  Class-map: AutoQos-4.0-Output-Priority-Queue (match-any)
    Match:  dscp cs4 (32) cs5 (40) ef (46)
    Match: cos  5
    Priority: 30% (300000 kbps), burst bytes 7500000,

    Priority Level: 1

  Class-map: AutoQos-4.0-Output-Control-Mgmt-Queue (match-any)
    Match:  dscp cs3 (24) cs6 (48) cs7 (56)
    Match: cos  3
    Queueing
    queue-limit dscp 16 percent 80
    queue-limit dscp 24 percent 90
    queue-limit dscp 48 percent 100

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 10%

    queue-buffers ratio 10

  Class-map: AutoQos-4.0-Output-Multimedia-Conf-Queue (match-any)
    Match:  dscp af41 (34) af42 (36) af43 (38)
    Match: cos  4
    Queueing

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 10%
    queue-buffers ratio 10

  Class-map: AutoQos-4.0-Output-Trans-Data-Queue (match-any)
    Match:  dscp af21 (18) af22 (20) af23 (22)
    Match: cos  2
    Queueing

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 10%
    queue-buffers ratio 10

  Class-map: AutoQos-4.0-Output-Bulk-Data-Queue (match-any)
    Match:  dscp af11 (10) af12 (12) af13 (14)
    Match: cos  1
    Queueing

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 4%
    queue-buffers ratio 10

  Class-map: AutoQos-4.0-Output-Scavenger-Queue (match-any)
    Match:  dscp cs1 (8)
    Queueing

    (total drops) 0
    (bytes output) 0
    bandwidth remaining 1%
```

```
  queue-buffers ratio 10

Class-map: AutoQos-4.0-Output-Multimedia-Strm-Queue (match-any)
  Match:  dscp af31 (26) af32 (28) af33 (30)
  Queueing

  (total drops) 0
  (bytes output) 0
  bandwidth remaining 10%
  queue-buffers ratio 10

Class-map: class-default (match-any)
  Match: any
  Queueing

  (total drops) 0
  (bytes output) 0
  bandwidth remaining 25%
  queue-buffers ratio 25
```

Related Topics

Configuring Auto-QoS , on page 329

Auto-QoS Policy and Class Maps, on page 329

Where to Go Next for Auto-QoS

Review the QoS documentation if you require any specific QoS changes to your auto-QoS configuration.

Additional References for Auto-QoS

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	*QoS Command Reference (Catalyst 3850 Switches)* *Cisco IOS Quality of Service Solutions Command Reference*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
—	

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Auto-QoS

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 18

Configuring Wireless QoS

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for Wireless QoS

Before configuring wireless QoS, you must have a thorough understanding of these items:

- QoS concepts.
- Wireless concepts and network topologies.
- Classic Cisco IOS QoS.
- Modular QoS CLI (MQC).
- Understanding of QoS implementation.

- The types of applications used and the traffic patterns on your network.
- Traffic characteristics and needs of your network. For example, is the traffic on your network bursty? Do you need to reserve bandwidth for voice and video streams?
- Bandwidth requirements and speed of the network.
- Location of congestion points in the network.

Restrictions for QoS on Wireless Targets

General Restrictions

A target is an entity where a policy is applied. You can apply a policy to either a wired or wireless target. A wired target can be either a port or VLAN. A wireless target can be either a port, radio, SSID, or client. Only port, SSID, and client policies are user configurable. Radio polices are not user configurable. Wireless QoS policies for port, radio, SSID, and client are applied in the downstream direction, and for upstream only SSID and client targets are supported. Downstream indicates that traffic is flowing from the switch to the wireless client. Upstream indicates that traffic is flowing from wireless client to the switch.

- Only port, SSID, and client (using AAA and Cisco IOS command-line interface) policies are user-configurable. Radio policies are set by the wireless control module and are not user-configurable.
- Port and radio policies are applicable only in the downstream direction.
- SSID and client support non-queuing policies in the upstream direction.

  SSID and client targets can be configured with marking and policing policies.
- One policy per target per direction is supported.

Wireless QoS Restrictions on Ports

The following are restrictions for applying QoS features on a wireless port target:

- All wireless ports have similar parent policy with one class-default and one action shape under class-default. Shape rates are dependent on the 802.11a/b/g/ac bands.
- You can create a maximum of four classes in a child policy by modifying the `port_chlid_policy`.
- If there are four classes in the `port_child_policy` at the port level, one must be a non-client-nrt class and one must be class-default.
- No two classes can have the same priority level. Only priority level 1 (for voice traffic and control traffic) and 2 (for video) are supported.
- Priority is not supported in the multicast NRT class (non-client-nrt class) and `class-default`.
- If four classes are configured, two of them have to be priority classes. If only three classes are configured, at least one of them should be a priority class. If three classes are configured and there is no non-client-nrt class, both priority levels must be present.
- Only match DSCP is supported.
- The port policy applied by the wireless control module cannot be removed using the CLI.
- Both priority rate and police CIR (using MQC) in the same class is unsupported.

- Queue limit (which is used to configure Weighted Tail Drop) is unsupported.

Wireless QoS Restrictions on SSID

The following are restrictions for applying QoS features on SSID:

- One table map is supported at the ingress policy.
- Table maps are supported for the parent class-default only. Up to two table maps are supported in the egress direction and three table-maps can be configured when a QoS group is involved.

Note Table-maps are not supported at the client targets.

- If a wireless port has a default policy with only two queues (one for multicast-NRT, one for class-default), the policy at SSID level cannot have voice and video class in the egress direction.
- Policing without priority is not supported in the egress direction.
- Priority configuration at the SSID level is used only to configure the RT1 and RT2 policers (AFD for policer). Priority configuration does not include the shape rate. Therefore, priority is restricted for SSID policies without police.
- The mapping in the DSCP2DSCP and COS2COS table should be based on the classification function for the voice and video classes in the port level policy.
- No action is allowed under the class-default of a child policy.
- For a flat policy (non hierarchical), in the ingress direction, the policy configuration must be a set (table map) or policing or both.

Wireless QoS Restrictions on Clients

The following are restrictions for applying QoS policies on client targets:

- Queuing is not supported.
- Attaching, removing, or modifying client policies on a WLAN in the enabled state is not supported. You must shut down the WLAN to apply, remove, or modify a policy.
- Table-map configuration is not supported for client targets.
- Policing and set configured together in class-default is blocked in both the upstream and downstream direction:

```
policy-map foo
class class-default
police X
set dscp Y
```

- Child policy is not supported under class-default if the parent policy contains other user-defined class maps in it.
- Hierarchical client polies are only supported in the egress direction.
- For flat egress client policy, policing in class-default and marking action in other classes are not supported.
- Restrictions for ACLs:

- ◦ All the filters in classes in a policy map for client policy must have the same attributes. Filters matching on protocol-specific attributes such as IPv4 or IPv6 addresses are considered as different attribute sets.
- ◦ For filters matching on ACLs, all ACEs (Access Control Entry) in the access list should have the same type and number of attributes. For example, the following is an invalid access list as they match on different attributes:

```
policy map foo
  class acl-101 (match on 3 tuple)
  police X
  class acl-102 (match on 5 tuple)
  police Y
```

- ◦ For filters matching on marking attributes, all filters in the policy-map must match on the same marking attribute. For example, If filter matches on DSCP, then all filters in the policy must match on DSCP.
- ◦ ACL matching on port ranges and subnet are only supported in ingress direction.

- If an ingress SSID policy is configured along with an ingress client policy matching ACLs with port ranges, the SSID policy takes precedence over the client policy. As a result, the client policy will not take effect.

Related Topics

Queuing in Wireless, on page 364
Port Policy Format, on page 362
Port Policies, on page 359
Radio Policies, on page 359
Restrictions for QoS on Wired Targets, on page 240

Information about Wireless QoS

Wireless QoS Overview

The flow of traffic from a wired source to a wireless target is known as downstream traffic. The flow of traffic from a wireless source to a wired target is known as upstream traffic.

- Wireless ports, including all physical ports to which an access point can be associated.
- Radio
- SSID (applicable on a per-radio, per-AP, and per-SSID)
- Client

Port, SSID, and client policies are user configurable. Radio policies are controlled by the wireless control module.

A target is the entity where the policy is applied. Wireless QoS policies for port, SSID, client, and radio are applied in the downstream direction. That is, when traffic is flowing from the switch to wireless client.

Note Only SSID and client policies are supported in both egress and ingress direction.

The following are some of the specific features provided by wireless QoS:

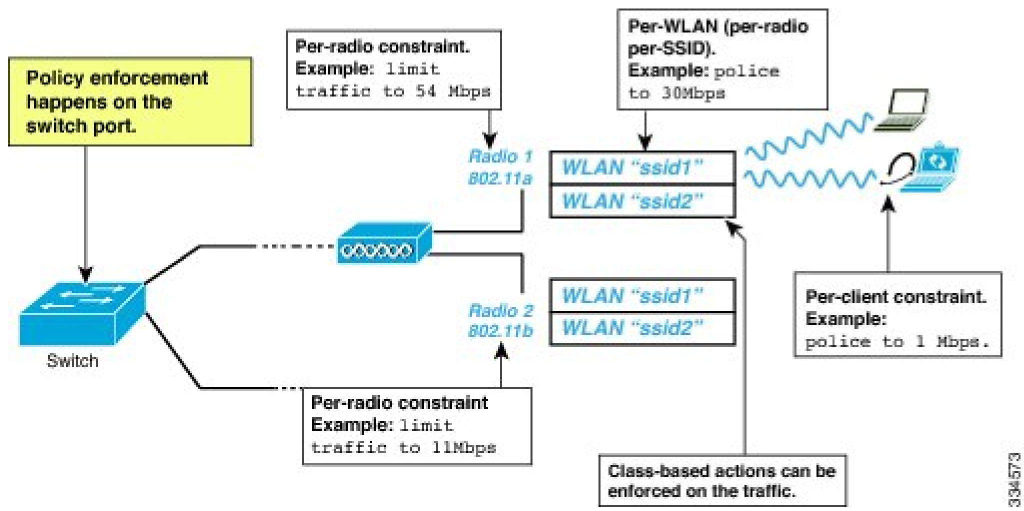
- Policies on wireless QoS targets:
    - Port
    - Radio
    - SSID
    - Client
- Queuing support
- Policing of wireless traffic
- Shaping of wireless traffic
- Rate limiting in both downstream and upstream direction
- Approximate Fair Drop (AFD). AFD is configured using shaping in SSID policies and policing in client policies. Queue limits are not defined on AFT policiers in clients.
- Mobility support for QoS
- Compatibility with precious metal QoS policies available on Cisco Unified Wireless Controllers.

Hierarchical Wireless QoS

The switch supports hierarchical QoS for wireless targets. Hierarchical QoS policies are applicable on port, radio, SSID, and client. QoS policies configured on the device (including marking, shaping, policing) can be applied across the targets. If the network contains non-realtime traffic, the non-realtime traffic is subject to approximate fair drop. Hierarchy refers to the process of application of the various QoS policies on the packets arriving to the device.

This figure shows the various targets available on a wireless network, as well as a hierarchal wireless configuration. Wireless QoS is applied per-radio constraint, per-WLAN, and per-client constraint.

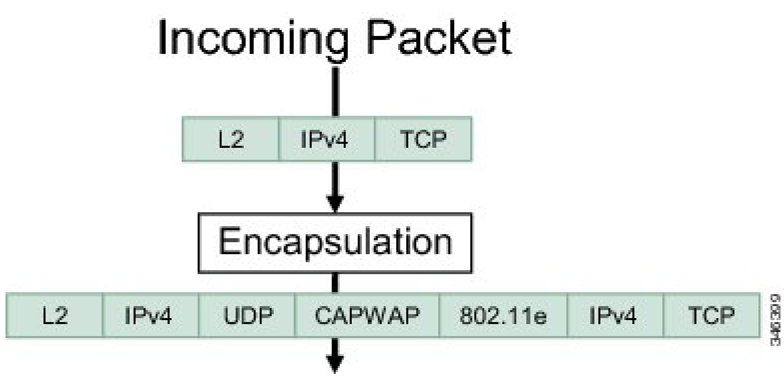
Figure 5: Hierarchical QoS



Wireless Packet Format

This figure displays the wireless packet flow and encapsulation used in hierarchical wireless QoS. The incoming packet enters the switch. The switch encapsulates this incoming packet and adds the 802.11e and CAPWAP headers.

Figure 6: Wireless Packet Path in the Egress Direction during First Pass



Hierarchical AFD

Approximate Fair Dropping (AFD) is a feature provided by the QoS infrastructure in Cisco IOS. For wireless targets, AFD can be configured on SSID (via shaping) and clients (via policing). AFD shaping rate is only applicable for downstream direction. Unicast real-time traffic is not subjected to AFD drops.

Wireless QoS Targets

This section describes the various wireless QoS targets available on a switch.

Port Policies

The switch supports port-based policies. The port policies includes port shaper and a child policy (port_child_policy).

Note Port child policies only apply to wireless ports and not to wired ports on the switch. A wireless port is defined as a port to which APs join. A default port child policy is applied on the switch to the wireless ports at start up.The port shaper rate is limited to 1G

Port shaper specifies the traffic policy applicable between the device and the AP. This is the sum of the radio rates supported on the access point.

The child policy determines the mapping between packets and queues defined by the port-child policy. The child policy can be configured to include voice, video, class-default, and non-client-nrt classes where voice and video are based on DSCP value (which is the outer CAPWAP header DSCP value). The definition of class-default is known to the system as any value other than voice and video DSCP.

The DSCP value is assigned when the packet reaches the port. Before the packet arrives at the port, the SSID policies are applied on the packet. Port child policy also includes multicast percentage for a given port traffic. By default, the port child policy allocates up to 10 percent of the available rate.

Related Topics

Queuing in Wireless, on page 364

Restrictions for QoS on Wireless Targets, on page 354

Supported QoS Features on Wireless Targets, on page 361

Examples: Wireless QoS Policy Classified by Voice, Video, and Multicast Traffic, on page 372

Radio Policies

The radio policies are system defined and are not user configurable. Radio wireless targets are only applicable in the downstream direction.

Radio policies are applicable on a per-radio, per-access point basis. The rate limit on the radios is the practical limit of the AP radio rate. This value is equivalent to the sum of the radios supported by the access point.

The following radios are supported:

- 802.11 a/n
- 802.11 b/n

Related Topics

Restrictions for QoS on Wireless Targets, on page 354

Supported QoS Features on Wireless Targets, on page 361

SSID Policies

You can create QoS policies on SSID BSSID (Basic Service Set Identification) in both the upstream and downstream directions. By default, there is no SSID policy. You can configure an SSID policy based on the SSID name. The policy is applicable on a per BSSID.

The types of policies you can create on SSID include marking by using table maps (table-maps), shape rate, and RT1 (Real Time 1) and RT2 (Real Time 2) policiers. If traffic is upstream, you usually configure a marking policy on the SSID. If traffic is downstream, you can configure marking and queuing.

There should be a one-to-one mapping between the policies configured on a port and an SSID. For example, if you configure class voice and class video on the port, you can have a similar policy on the SSID.

SSID priorities can be specified by configuring bandwidth remaining ratio. Queuing SSID policies are applied in the downstream direction.

Related Topics

Supported QoS Features on Wireless Targets, on page 361

Examples: SSID Policy, on page 373

Examples: Configuring Downstream SSID Policy, on page 373

Client Policies

Client policies are applicable in the upstream and downstream direction. The wireless control module of the switch applies the default client policies when admission control is enabled for WMM clients. When admission control is disabled, there is no default client policy. You can configure policing and marking policies on clients.

You can configure client policies in the following ways:

- Using AAA—You can use a combination of AAA and TCLAS, and AAA and SIP snooping when configuring with AAA.
- Using the Cisco IOS MQC CLI—You can use a combination of CLI and TCLAS and CLI and SIP snooping.
- Using the default configuration

Note When applying client policies on a WLAN, you must disable the WLAN before modifying the client policy. SSID policies can be modified even if the WLAN is enabled.

Note If you configured AAA by configuring the unified wireless controller procedure, and using the MQC QoS commands, the policy configuration performed through the MQC QoS commands takes precedence.

For client policies, the following filters are supported:

- ACL
- DSCP
- COS

- WLAN UP

Related Topics

Supported QoS Features on Wireless Targets, on page 361
Examples: Client Policies, on page 374

Supported QoS Features on Wireless Targets

This table describes the various features available on wireless targets.

Table 46: QoS Features Available on Wireless Targets

Target	Features	Traffic	Direction Where Policies Are Applicable	Comments
Port	• Port shaper • Priority queuing • Multicast policing	Non-Real Time (NRT), Real Time (RT)	Downstream	
Radio	• Shaping	Non-Real Time	Downstream	Radio policies are not user configurable.
SSID	• Shaping • Police • Table map • BRR	Non-Real Time, Real Time	Upstream and downstream	Queuing actions such as shaping and BRR are allowed only in the downstream direction.
Client	• Set • Police	Non-Real Time, Real time	Upstream and downstream	

Related Topics

Queuing in Wireless, on page 364
Port Policy Format, on page 362
Port Policies, on page 359
Radio Policies, on page 359
SSID Policies, on page 360
Client Policies, on page 360

Port Policy Format

This section describes the behavior of the port policies on a switch. The ports on the switch do not distinguish between wired or wireless physical ports. Depending on the kind of device associated to the switch, the policies are applied. For example, when an access point is connected to a switch port, the switch detects it as a wireless device and applies the default hierarchical policy which is in the format of a parent-child policy. This policy is an hierarchical policy. The parent policy cannot be modified but the child policy (port-child policy) can be modified to suit the QoS configuration. The switch is pre configured with a default class map and a policy map.

Default class map:

```
Class Map match-any non-client-nrt-class
   Match non-client-nrt
```

The above port policy processes all network traffic to the Q3 queue. You can view the class map by executing the **show class-map** command.

Default policy map:

```
Policy Map port_child_policy
    Class non-client-nrt-class
      bandwidth remaining ratio 10
```

Note The class map and policy map listed are system-defined policies and cannot be changed.

The following is the system-defined policy map available on the ports on which wireless devices are associated. The format consists of a parent policy and a service child policy (**port_child_policy**). To customize the policies to suite your network needs, you must configure the port child policy.

```
Policy-map policy_map_name
    Class class-default
         Shape average average_rate
         Service-policy port_child_policy
```

Note The parent policy is system generated and cannot be changed. You must configure the *port_child_policy* policy to suit the QoS requirements on your network.

Depending on the type of traffic in your network, you can configure the port child policy. For example, in a typical wireless network deployment, you can assign specific priorities to voice and video traffic. Here is an example:

```
Policy-map port_child_policy
     Class  voice-policy-name (match dscp ef)
          Priority level 1
         Police (multicast-policer-name-voice) Multicast Policer
    Class video-policy-name (match dscp af41)
        Priority level 2
        Police (multicast-policer-name-video) Multicast Policer
Class non-client-nrt-class traffic(match non-client-nrt)
        Bandwidth remaining ratio (brr-value-nrt-q2)
    Class class-default  (NRT Data)
        Bandwidth remaining ratio (brr-value-q3)
```

In the above port child policy:

- *voice-policy-name*— Refers to the name of the class that specifies rules for the traffic for voice packets. Here the DSCP value is mapped to a value of 46 (represented by the keyword **ef**). The voice traffic is assigned the highest priority of 1.
- *video-policy-name*— Refers to the name of the class that specifies rules for the traffic for video packets. The DSCP value is mapped to a value of 34 (represented by the keyword **af41**).
- *multicast-policer-name-voice*— If you need to configure multicast voice traffic, you can configure policing for the voice class map.
- *multicast-policer-name-video*— If you need to configure multicast video traffic, you can configure policing for the video class map.

In the above sample configuration, all voice and video traffic is directed to the Q0 and Q1 queues, respectively. These queues maintain a strict priority. The packets in Q0 and Q1 are processed in that order. The bandwidth remaining ratios *brr-value-nrt-q2* and *brr-value-q3* are directed to the Q2 and Q3 respectively specified by the class maps and *class-default* and *non-client-nrt*. The processing of packets on Q2 and Q3 are based on a weighted round-robin approach. For example, if the *brr-value-nrtq2* has a value of 90 and *brr-value-nrtq3* is 10, the packets in queue 2 and queue 3 are processed in the ratio of 9:1.

Related Topics

Queuing in Wireless, on page 364
Restrictions for QoS on Wireless Targets, on page 354
Supported QoS Features on Wireless Targets, on page 361
Examples: Wireless QoS Policy Classified by Voice, Video, and Multicast Traffic, on page 372

Wireless QoS Rate Limiting

QoS per Client Rate Limit—Wireless

QoS policies can be configured to rate-limit client traffic using policiers. Ths includes both real-time and non real time traffic. The non real-time traffic is policed using AFD policiers. These policiers can only be one rate two color.

Note For client policy, the voice and video rate limits are applied at the same time.

QoS Downstream Rate Limit—Wireless

Downstream rate limiting is done using policing at the SSID level. AFD cannot drop real-time traffic, it can only be policed in the traffic queues. Real-time policing and AFD shaping is performed at the SSID level.

The radio has a default shaping policy. This shaping limit is the physical limit of the radio itself. You can check the policy maps on the radio by using the **show policy-map interface wireless radio** command.

Wireless QoS Multicast

You can configure multicast policing rate at the port level.

Related Topics

Configuring QoS Policies for Multicast Traffic

Queuing in Wireless

Queuing in the wireless component is performed based on the port policy and is applicable only in the downstream direction. The wireless module supports the following four queues:

- Voice—This is a strict priority queue. Represented by Q0, this queue processes control traffic and multicast or unicast voice traffic. All control traffic (such as CAPWAP packets) is processed through the voice queue. The QoS module uses a different threshold within the voice queue to process control and voice packets to ensure that control packets get higher priority over other non-control packets.
- Video—This is a strict priority queue. Represented by Q1, this queue processes multicast or unicast video traffic.
- Data NRT—Represented by Q2, this queue processes all non-real-time unicast traffic.
- Multicast NRT—Represented by Q3, this queue processes Multicast NRT traffic. Any traffic that does not match the traffic in Q0, Q1, or Q2 is processed through Q3.

Note By default, the queues Q0 and Q1 are not enabled.

Note A weighted round-robin policy is applied for traffic in the queues Q2 and Q3.

For upstream direction only one queue is available. Port and radio policies are applicable only in the downstream direction.

Note The wired ports support eight queues.

Related Topics

Port Policy Format, on page 362

Port Policies, on page 359

Restrictions for QoS on Wireless Targets, on page 354

Supported QoS Features on Wireless Targets, on page 361

Examples: Wireless QoS Policy Classified by Voice, Video, and Multicast Traffic, on page 372

Wireless QoS Mobility

Wireless QoS mobility enables you to configure QoS policies so that the network provides the same service anywhere in the network. A wireless client can roam from one location to another and as a result the client can get associated to different access points associated with a different switch. Wireless client roaming can be classified into two types:

- Intra-switch roaming
- Inter-switch roaming

Note The client policies must be available on all of the switches in the mobility group. The same SSID and port policy must be applied to all switches in the mobility group so that the clients get consistent treatment.

Inter-Switch Roaming

When a client roams from one location to another, the client can get associated to access points either associated to the same switch (anchor switch) or a different switch (foreign switch). Inter-switch roaming refers to the scenario where the client gets associated to an access point that is not associated to the same device before the client roamed. The host device is now foreign to the device to which the client was initially anchored.

In the case of inter-switch roaming, the client QoS policy is always executed on the foreign controller. When a client roams from anchor switch to foreign switch, the QoS policy is uninstalled on the anchor switch and installed on the foreign switch. In the mobility handoff message, the anchor device passes the name of the policy to the foreign switch. The foreign switch should have a policy with the same name configured for the QoS policy to be applied correctly.

In the case of inter-switch roaming, all of the QoS policies are moved from the anchor device to the foreign device. While the QoS policies are in transition from the anchor device to the foreign device, the traffic on the foreign device is provided the default treatment. This is comparable to a new policy installation on the client target.

Note If the foreign device is not configured with the user-defined physical port policy, the default port policy is applicable to all traffic is routed through the NRT queue, except the control traffic which goes through RT1 queue. The network administrator must configure the same physical port policy on both the anchor and foreign devices symmetrically.

Intra-Switch Roaming

With intra-switch roaming, the client gets associated to an access point that is associated to the same switch before the client roamed, but this association to the device occurs through a different access point.

Note QoS policies remain intact in the case of intra-switch roaming.

Precious Metal Policies for Wireless QoS

Wireless QoS is backward compatible with the precious metal policies offered by the unified wireless controller platforms. The precious metal policies are system-defined policies that are available on the controller.

The following policies are available:

- Platinum—Used for VoIP clients.

- Gold—Used for video clients.
- Silver— Used for traffic that can be considered best-effort.
- Bronze—Used for NRT traffic.

These policies (also known as profiles) can be applied to a WLAN based on the traffic. We recommend the configuration using the Cisco IOS MQC configuration. The policies are available in the system based on the precious metal policy required.

Based on the policies applied, the 802.1p, 802.11e (WMM), and DSCP fields in the packets are affected. These values are preconfigured and installed when the switch is booted.

Note Unlike the precious metal policies that were applicable in the Cisco Unified Wireless controllers, the attributes rt-average-rate, nrt-average-rate, and peak rates are not applicable for the precious metal policies configured on this switch platform.

Related Topics

Configuring Precious Metal Policies , on page 366

How to Configure Wireless QoS

Configuring Precious Metal Policies

You can configure precious metal QoS policies on a per-WLAN basis.

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name*
3. **service-policy output** *policy-name*
4. **end**
5. **show wlan** {*wlan-id* | *wlan-name*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.
Step 2	**wlan** *wlan-name* **Example:** `Switchwlan test4`	Enters the WLAN configuration submode.

	Command or Action	Purpose

Step 3	**service-policy output** *policy-name*
	
	Example:
	`Switch(config-wlan)# service-policy output platinum`
	
	Example:
	`Switch(config-wlan)# service-policy input platinum-up`
		Configures the WLAN with the QoS policy. To configure the WLAN with precious metal policies, you must enter one of the following keywords: **platinum**, **gold**, **silver**, or **bronze**. The upstream policy is specified with the keyword **platinum-up** as shown in the example.
		
		Note Upstream policies differ from downstream policies. The upstream policies have a suffix of -up.

Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit the global configuration mode.

Step 5	**show wlan** {*wlan-id* | *wlan-name*}
	
	Example:
	`Switch# show wlan name qos-wlan`
		Verifies the configured QoS policy on the WLAN.
		
		`Switch# show wlan name qos-wlan`
		`. . .`
		`. . .`
		`. . .`
		
		`QoS Service Policy - Input`
		`  Policy Name                          : platinum-up`
		
		`  Policy State                         : Validated`
		`QoS Service Policy - Output`
		`  Policy Name                          : platinum`
		`  Policy State                         : Validated`
		`. . .`
		
		`. . .`

Related Topics

Precious Metal Policies for Wireless QoS, on page 365

Configuring Class Maps for Voice and Video

To configure class maps for voice and video traffic, follow these steps:

SUMMARY STEPS

1. **configure terminal**
2. **class-map** *class-map-name*
3. **match dscp** *dscp-value-for-voice*
4. **end**
5. **configure terminal**
6. **class-map** *class-map-name*
7. **match dscp** *dscp-value-for-video*
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**class-map** *class-map-name* **Example:** `Switch(config)# class-map voice`	Creates a class map.
Step 3	**match dscp** *dscp-value-for-voice* **Example:** `Switch(config-cmap)# match dscp 46`	Matches the DSCP value in the IPv4 and IPv6 packets. Set this value to 46.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.
Step 5	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 6	**class-map** *class-map-name* **Example:** `Switch(config)# class-map video`	Configures a class map.
Step 7	**match dscp** *dscp-value-for-video* **Example:** `Switch(config-cmap)# match dscp 34`	Matches the DSCP value in the IPv4 and IPv6 packets. Set this value to 34.

	Command or Action	Purpose
Step 8	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Client Policies

Before You Begin

You must have the following features configured before configuring client policies:

- Access lists
- Access group name

SUMMARY STEPS

1. **configure terminal**
2. **ip access-list extended** *ext-name*
3. **permit ip host** *host-ip-address*
4. **end**
5. **configure terminal**
6. **class map** *acl-name*
7. **match access-group name** *access-list-name*
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip access-list extended** *ext-name* **Example:** `Switch(config)# ip access-list extended`	Configures a named access list.
Step 3	**permit ip host** *host-ip-address* **Example:** `Switch(config-ext-nacl)# permit ip host 203.0.113.3 host 203.0.113.5`	Configures IP protocol traffic from a source address to a destination address.

	Command or Action	Purpose
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.
Step 5	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 6	**class map** *acl-name* **Example:** `Switch(config)# class-map acl-a1`	Configures the class map name.
Step 7	**match access-group name** *access-list-name* **Example:** `Switch(config-cmap)# match access-group name a1`	Assigns the class map to an access group name.
Step 8	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Table Maps

SUMMARY STEPS

1. **configure terminal**
2. table-map *table-map-name*
3. **map from** *from-value* **to** *to-value*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	table-map *table-map-name*	Create the table map.

	Command or Action	Purpose
	Example: `Switch(config)# table-map mutate-dscp`	

Step 3	**map from** *from-value* **to** *to-value*
	
	Example:
	`Switch(config-tablemap)# map from 10 to 34`
	`Switch(config-tablemap)# map from 34 to 40`
	`Switch(config-tablemap)# map from 46 to 48`
		Map a to value to a from value.

Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Applying an SSID or Client Policy on a WLAN

Before You Begin

You must have a service-policy map configured before applying it on an SSID.

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *profile-name*
3. **service-policy** [**input** | **output**] *policy-name*
4. **service-policy client** [**input** | **output**] *policy-name*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *profile-name* **Example:** `Switch# wlan test4`	Enters the WLAN configuration submode. The *profile-name* is the profile name of the configured WLAN.

	Command or Action	Purpose
Step 3	**service-policy** [**input** | **output**] *policy-name* **Example:** `Switch(config-wlan)# service-policy input policy-map-ssid`	Applies the policy. The following options are available: • **input**— Assigns the policy map to WLAN ingress traffic. • **output**— Assigns the policy map to WLAN egress traffic.
Step 4	**service-policy client** [**input** | **output**] *policy-name* **Example:** `Switch(config-wlan)# service-policy client input policy-map-client`	Applies the policy. The following options are available: • **input**— Assigns the client policy for ingress direction on the WLAN. • **output**— Assigns the client policy for egress direction on the WLAN.
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuration Examples

Examples: Wireless QoS Policy Classified by Voice, Video, and Multicast Traffic

The following example provides a template for creating a port child policy for managing quality of service for voice and video traffic.

```
Policy-map port_child_policy
    Class voice (match dscp ef)
            Priority level 1
            Police  Multicast Policer
    Class video (match dscp af41)
            Priority level 2
            Police  Multicast Policer
    Class mcast-data  (match non-client-nrt)
            Bandwidth remaining ratio <>
    Class class-default  (NRT Data)
            Bandwidth remaining ratio <>
```

Note Multicast Policer in the example above is not a keyword. It refers to the policing policy configured.

Two class maps with name voice and video are configured with DSCP assignments of 46 and 34. The voice traffic is assigned the priority of 1 and the video traffic is assigned the priority level 2 and is processed using Q0 and Q1. If your network receives multicast voice and video traffic, you can configure multicast policers. The non-client NRT data and NRT data are processed using the Q2 and Q3 queues.

Related Topics

Queuing in Wireless, on page 364

Port Policy Format, on page 362

Port Policies, on page 359

Examples: SSID Policy

SSID Policy 1

The following is an example of an SSID policy for voice and video:

```
Policy-map enterprise-ssid-1
     Class  voice (match dscp ef)
          Priority level 1
         Police Unicast Policer
Class video (match dscp af41)
        Priority level 2
        Police Unicast Policer
Policy-map ssid-shaper
Class class-default  (NRT Data)
        queue-buffer ratio 0
        shape average 100000000
        set wlan-user-priority dscp table dscp2up
        set dscp dscp table dscp2dscp
        service-policy  enterprise-ssid-1
```

SSID Policy 2

The following is an example of SSID policy configured with an average SSID shaping rate:

```
Policy-map enterprise-ssid-2
     Class  voice (match dscp af11)
          Priority level 1
         Police Unicast Policer
Class video (match dscp  ef)
        Priority level 2
        Police Unicast Policer
Policy-map ssid-shaper
Class class-default  (NRT Data)
       shape average 1000000000
       service-policy enterprise-ssid-2
       set wlan-user-priority dscp table dscp2up
       set dscp dscp table dscp2dscp
```

Related Topics

SSID Policies, on page 360

Examples: Configuring Downstream SSID Policy

To configure a downstream BSSID policy, you must first configure a port child policy with priority level queuing.

Configuring a User-Defined Port Child Policy

The following is an example of configuring a user-defined port child policy:

```
policy-map port_child_policy
   class voice
     priority level 1 20000

   class video
```

```
      priority level 2 10000

    class non-client-nrt-class
      bandwidth remaining ratio 10

    class class-default
      bandwidth remaining ratio 15
```

Configuring Downstream BSSID Policy

The following configuration example displays how to configure a downstream BSSID policy:

```
policy-map bssid-policer
   queue-buffer ratio 0
   class class-default
   shape average 30000000
set dscp dscp table dscp2dscp
set wlan user-priority dscp table dscp2up
service-policy ssid_child_qos
```

The SSID child QoS policy may be defined as below:

```
Policy Map ssid-child_qos
    Class voice
      priority level 1
      police cir 5m
      admit cac wmm-tspec
           UP 6,7  / tells WCM allow 'voice' TSPEC\SIP snoop for this ssid
            rate 4000 / must be police rate value is in kbps)
   Class video
      priority level 2
      police cir 60000
```

Related Topics

SSID Policies, on page 360

Examples: Client Policies

The following example shows a default client policy in the downstream direction. Any incoming traffic contains the user-priority as 0:

Note The default client policy is enabled only on WMM clients that are ACM-enabled.

```
Policy-map client-def-down
  class class-default
      set wlan user-priority 0
```

The following example shows the default client policy in the upstream direction. Any traffic that is sent to the wired network from wireless network will result in the DSCP value being set to 0.

Note The default client policy is enabled only on WMM clients that are ACM-enabled.

```
Policy-map client-def-up
   class class-default
    set dscp 0
```

The following examples shows client policies that are generated automatically and applied to the WMM client when the client authenticates to a profile in AAA with a QoS-level attribute configured.

```
Policy Map platinum-WMM
Class voice-plat
  set wlan user-priority 6
 Class video-plat
 set wlan user-priority 4
Class class-default
  set wlan user-priority 0

Policy Map gold-WMM
Class voice-gold
  set wlan user-priority 4
 Class video-gold
 set wlan user-priority 4
Class class-default
  set wlan user-priority 0
```

The following is an example of non-WMM client precious metal policies:

```
Policy Map platinum
  set wlan user-priority 6
```

Any traffic matching class voice1 the user priority is set to a pre-defined value. The class can be set to assign a DSCP or ACL.

```
Policy Map client1-down
Class voice1     //match dscp, cos
  set wlan user-priority <>
Class voice2    //match acl
   set wlan user-priority <>
Class voice3
   set wlan user-priority <>
Class class-default
  set wlan user-priority 0
```

The following is an example of a client policy based on AAA and TCLAS:

```
Policy Map client2-down[ AAA+ TCLAS pol example]
Class      voice\\match dscp
      police <>
      set <>
Class class-default
    set <>
Class voice1|| voice2 [match acls]
      police <>
      class voice1
        set <>
      class voice2
        set <>
```

The following is an example of a client policy for voice and video for traffic in the downstream direction:

```
Policy Map client3-down
    class voice \\match dscp, cos
          police X
     class video
         police Y
    class class-default
        police Z
```

The following is an example of a client policy for voice and video for traffic in the upstream direction using policing:

```
Policy Map client1-up
    class voice     \\match dscp, up, cos
```

```
      police X
    class video
     police Y
  class class-default
    police Z
```

The following is an example of a client policy for voice and video based on DSCP:

```
Policy Map client2-up
    class voice     \\match dscp, up, cos
set dscp <>
    class video
     set dscp <>
  class class-default
    set dscp <>
```

Related Topics

Client Policies, on page 360

Additional References

Related Documents

Related Topic	Document Title
QoS Command Reference	*QoS Command Reference (Catalyst 3850 Switches)*
Mobility Configuration Guide	*Mobility Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*
Quality of Service Solutions Configuration Guide (Cisco IOS Software)	*Quality of Service Solutions Configuration Guide Library, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

PART III

Interface

CHAPTER 19

Configuring Interface Characteristics

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information About Configuring Interface Characteristics

Interface Types

This section describes the different types of interfaces supported by the switch. The rest of the chapter describes configuration procedures for physical interface characteristics.

Note The stack ports on the rear of the stacking-capable switches are not Ethernet ports and cannot be configured.

Port-Based VLANs

A VLAN is a switched network that is logically segmented by function, team, or application, without regard to the physical location of the users. Packets received on a port are forwarded only to ports that belong to the same VLAN as the receiving port. Network devices in different VLANs cannot communicate with one another without a Layer 3 device to route traffic between the VLANs.

VLAN partitions provide hard firewalls for traffic in the VLAN, and each VLAN has its own MAC address table. A VLAN comes into existence when a local port is configured to be associated with the VLAN, when the VLAN Trunking Protocol (VTP) learns of its existence from a neighbor on a trunk, or when a user creates a VLAN. VLANs can be formed with ports across the stack.

To configure VLANs, use the **vlan** *vlan-id* global configuration command to enter VLAN configuration mode. The VLAN configurations for normal-range VLANs (VLAN IDs 1 to 1005) are saved in the VLAN database. If VTP is version 1 or 2, to configure extended-range VLANs (VLAN IDs 1006 to 4094), you must first set VTP mode to transparent. Extended-range VLANs created in transparent mode are not added to the VLAN database but are saved in the switch running configuration. With VTP version 3, you can create extended-range VLANs in client or server mode. These VLANs are saved in the VLAN database.

In a switch stack, the VLAN database is downloaded to all switches in a stack, and all switches in the stack build the same VLAN database. The running configuration and the saved configuration are the same for all switches in a stack.

Add ports to a VLAN by using the **switchport** interface configuration commands:

- Identify the interface.
- For a trunk port, set trunk characteristics, and, if desired, define the VLANs to which it can belong.
- For an access port, set and define the VLAN to which it belongs.
- For a tunnel port, set and define the VLAN ID for the customer-specific VLAN tag.

Switch Ports

Switch ports are Layer 2-only interfaces associated with a physical port. Switch ports belong to one or more VLANs. A switch port can be an access port or a trunk port. You can configure a port as an access port or trunk port or let the Dynamic Trunking Protocol (DTP) operate on a per-port basis to set the switchport mode by negotiating with the port on the other end of the link. Switch ports are used for managing the physical interface and associated Layer 2 protocols and do not handle routing or bridging.

Configure switch ports by using the **switchport** interface configuration commands.

Access Ports

An access port belongs to and carries the traffic of only one VLAN (unless it is configured as a voice VLAN port). Traffic is received and sent in native formats with no VLAN tagging. Traffic arriving on an access port is assumed to belong to the VLAN assigned to the port. If an access port receives a tagged packet (Inter-Switch Link [ISL] or IEEE 802.1Q tagged), the packet is dropped, and the source address is not learned.

Two types of access ports are supported:

- Static access ports are manually assigned to a VLAN (or through a RADIUS server for use with IEEE 802.1x.
- VLAN membership of dynamic access ports is learned through incoming packets. By default, a dynamic access port is not a member of any VLAN, and forwarding to and from the port is enabled only when the VLAN membership of the port is discovered. Dynamic access ports on the switch are assigned to a VLAN by a VLAN Membership Policy Server (VMPS). The VMPS can be a Catalyst 6500 series switch; the switch cannot be a VMPS server.

You can also configure an access port with an attached Cisco IP Phone to use one VLAN for voice traffic and another VLAN for data traffic from a device attached to the phone.

Trunk Ports

A trunk port carries the traffic of multiple VLANs and by default is a member of all VLANs in the VLAN database. These trunk port types are supported:

- In an ISL trunk port, all received packets are expected to be encapsulated with an ISL header, and all transmitted packets are sent with an ISL header. Native (non-tagged) frames received from an ISL trunk port are dropped.
- An IEEE 802.1Q trunk port supports simultaneous tagged and untagged traffic. An IEEE 802.1Q trunk port is assigned a default port VLAN ID (PVID), and all untagged traffic travels on the port default PVID. All untagged traffic and tagged traffic with a NULL VLAN ID are assumed to belong to the port default PVID. A packet with a VLAN ID equal to the outgoing port default PVID is sent untagged. All other traffic is sent with a VLAN tag.

Although by default, a trunk port is a member of every VLAN known to the VTP, you can limit VLAN membership by configuring an allowed list of VLANs for each trunk port. The list of allowed VLANs does not affect any other port but the associated trunk port. By default, all possible VLANs (VLAN ID 1 to 4094) are in the allowed list. A trunk port can become a member of a VLAN only if VTP knows of the VLAN and if the VLAN is in the enabled state. If VTP learns of a new, enabled VLAN and the VLAN is in the allowed list for a trunk port, the trunk port automatically becomes a member of that VLAN and traffic is forwarded to and from the trunk port for that VLAN. If VTP learns of a new, enabled VLAN that is not in the allowed list for a trunk port, the port does not become a member of the VLAN, and no traffic for the VLAN is forwarded to or from the port.

Tunnel Ports

Tunnel ports are used in IEEE 802.1Q tunneling to segregate the traffic of customers in a service-provider network from other customers who are using the same VLAN number. You configure an asymmetric link from a tunnel port on a service-provider edge switch to an IEEE 802.1Q trunk port on the customer switch. Packets entering the tunnel port on the edge switch, already IEEE 802.1Q-tagged with the customer VLANs, are encapsulated with another layer of an IEEE 802.1Q tag (called the metro tag), containing a VLAN ID unique in the service-provider network, for each customer. The double-tagged packets go through the service-provider network keeping the original customer VLANs separate from those of other customers. At the outbound interface, also a tunnel port, the metro tag is removed, and the original VLAN numbers from the customer network are retrieved.

Tunnel ports cannot be trunk ports or access ports and must belong to a VLAN unique to each customer.

Routed Ports

A routed port is a physical port that acts like a port on a router; it does not have to be connected to a router. A routed port is not associated with a particular VLAN, as is an access port. A routed port behaves like a regular router interface, except that it does not support VLAN subinterfaces. Routed ports can be configured with a Layer 3 routing protocol. A routed port is a Layer 3 interface only and does not support Layer 2 protocols, such as DTP and STP.

Configure routed ports by putting the interface into Layer 3 mode with the **no switchport** interface configuration command. Then assign an IP address to the port, enable routing, and assign routing protocol characteristics by using the **ip routing** and **router** *protocol* global configuration commands.

Note Entering a **no switchport** interface configuration command shuts down the interface and then re-enables it, which might generate messages on the device to which the interface is connected. When you put an interface that is in Layer 2 mode into Layer 3 mode, the previous configuration information related to the affected interface might be lost.

The number of routed ports that you can configure is not limited by software. However, the interrelationship between this number and the number of other features being configured might impact CPU performance because of hardware limitations.

Note The IP base feature set supports static routing and the Routing Information Protocol (RIP). For full Layer 3 routing or for fallback bridging, you must enable the IP services feature set on the standalone switch, or the active switch.

Switch Virtual Interfaces

A switch virtual interface (SVI) represents a VLAN of switch ports as one interface to the routing or bridging function in the system. You can associate only one SVI with a VLAN. You configure an SVI for a VLAN only to route between VLANs or to provide IP host connectivity to the switch. By default, an SVI is created for the default VLAN (VLAN 1) to permit remote switch administration. Additional SVIs must be explicitly configured.

Note You cannot delete interface VLAN 1.

SVIs provide IP host connectivity only to the system.

Although the switch stack or switch supports a total of 1005 VLANs and SVIs, the interrelationship between the number of SVIs and routed ports and the number of other features being configured might impact CPU performance because of hardware limitations.

SVIs are created the first time that you enter the **vlan** interface configuration command for a VLAN interface. The VLAN corresponds to the VLAN tag associated with data frames on an ISL or IEEE 802.1Q encapsulated trunk or the VLAN ID configured for an access port. Configure a VLAN interface for each VLAN for which you want to route traffic, and assign it an IP address.

Note When you create an SVI, it does not become active until it is associated with a physical port.

SVI Autostate Exclude

The line state of an SVI with multiple ports on a VLAN is in the *up* state when it meets these conditions:

- The VLAN exists and is active in the VLAN database on the switch
- The VLAN interface exists and is not administratively down.
- At least one Layer 2 (access or trunk) port exists, has a link in the *up* state on this VLAN, and is in the spanning-tree forwarding state on the VLAN.

Note The protocol link state for VLAN interfaces come up when the first switchport belonging to the corresponding VLAN link comes up and is in STP forwarding state.

The default action, when a VLAN has multiple ports, is that the SVI goes down when all ports in the VLAN go down. You can use the SVI autostate exclude feature to configure a port so that it is not included in the SVI line-state up-or-down calculation. For example, if the only active port on the VLAN is a monitoring port, you might configure autostate exclude on that port so that the VLAN goes down when all other ports go down. When enabled on a port, **autostate exclude** applies to all VLANs that are enabled on that port.

The VLAN interface is brought up when one Layer 2 port in the VLAN has had time to converge (transition from STP listening-learning state to forwarding state). This prevents features such as routing protocols from using the VLAN interface as if it were fully operational and minimizes other problems, such as routing black holes.

EtherChannel Port Groups

EtherChannel port groups treat multiple switch ports as one switch port. These port groups act as a single logical port for high-bandwidth connections between switches or between switches and servers. An EtherChannel balances the traffic load across the links in the channel. If a link within the EtherChannel fails, traffic previously carried over the failed link changes to the remaining links. You can group multiple trunk ports into one logical trunk port, group multiple access ports into one logical access port, group multiple tunnel ports into one logical tunnel port, or group multiple routed ports into one logical routed port. Most protocols operate over either single ports or aggregated switch ports and do not recognize the physical ports within the port group. Exceptions are the DTP, the Cisco Discovery Protocol (CDP), and the Port Aggregation Protocol (PAgP), which operate only on physical ports.

When you configure an EtherChannel, you create a port-channel logical interface and assign an interface to the EtherChannel. For Layer 3 interfaces, you manually create the logical interface by using the **interface port-channel** global configuration command. Then you manually assign an interface to the EtherChannel by using the **channel-group** interface configuration command. For Layer 2 interfaces, use the **channel-group** interface configuration command to dynamically create the port-channel logical interface. This command binds the physical and logical ports together.

10-Gigabit Ethernet Interfaces

A 10-Gigabit Ethernet interface operates only in full-duplex mode. The interface can be configured as a switched or routed port.

For more information about the Cisco TwinGig Converter Module, see the switch hardware installation guide and your transceiver module documentation.

Power over Ethernet Ports

A PoE-capable switch port automatically supplies power to one of these connected devices if the switch senses that there is no power on the circuit:

- a Cisco pre-standard powered device (such as a Cisco IP Phone or a Cisco Aironet Access Point)
- an IEEE 802.3af-compliant powered device
- an IEEE 802.3at-compliant powered device

A powered device can receive redundant power when it is connected to a PoE switch port and to an AC power source. The device does not receive redundant power when it is only connected to the PoE port.

After the switch detects a powered device, the switch determines the device power requirements and then grants or denies power to the device. The switch can also sense the real-time power consumption of the device by monitoring and policing the power usage.

Using the Switch USB Ports

The switch has two USB ports on the front panel — a USB mini-Type B console port and a USB Type A port.

USB Mini-Type B Console Port

The switch has the following console ports available on its front panel:

- USB mini-Type B console connection
- RJ-45 console port

Console output appears on devices connected to both ports, but console input is active on only one port at a time. By default, the USB connector takes precedence over the RJ-45 connector.

Note Windows PCs require a driver for the USB port. See the hardware installation guide for driver installation instructions.

Use the supplied USB Type A-to-USB mini-Type B cable to connect a PC or other device to the switch. The connected device must include a terminal emulation application. When the switch detects a valid USB connection to a powered-on device that supports host functionality (such as a PC), input from the RJ-45 console is immediately disabled, and input from the USB console is enabled. Removing the USB connection immediately reenables input from the RJ-45 console connection. An LED on the switch shows which console connection is in use.

Console Port Change Logs

At software startup, a log shows whether the USB or the RJ-45 console is active. Each switch in a stack issues this log. Every switch always first displays the RJ-45 media type.

In the sample output, switch 1 has a connected USB console cable. Because the bootloader did not change to the USB console, the first log from switch 1 shows the RJ-45 console. A short time later, the console changes and the USB console log appears. Switch 2 and switch 3 have connected RJ-45 console cables.

```
switch-stack-1
*Mar  1 00:01:00.171: %USB_CONSOLE-6-MEDIA_RJ45: Console media-type is RJ45.
*Mar  1 00:01:00.431: %USB_CONSOLE-6-MEDIA_USB: Console media-type is USB.

switch-stack-2
*Mar  1 00:01:09.835: %USB_CONSOLE-6-MEDIA_RJ45: Console media-type is RJ45.

switch-stack-3
*Mar  1 00:01:10.523: %USB_CONSOLE-6-MEDIA_RJ45: Console media-type is RJ45.
```

When the USB cable is removed or the PC de-activates the USB connection, the hardware automatically changes to the RJ-45 console interface:

```
switch-stack-1
Mar  1 00:20:48.635: %USB_CONSOLE-6-MEDIA_RJ45: Console media-type is RJ45.
```

You can configure the console type to always be RJ-45, and you can configure an inactivity timeout for the USB connector.

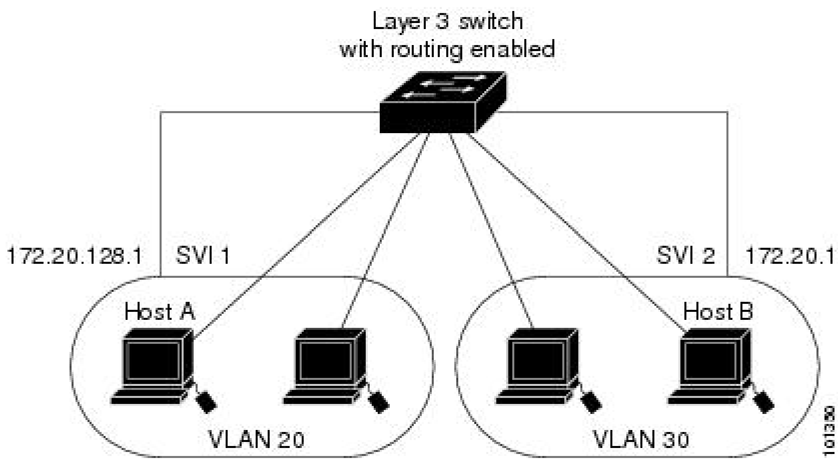
USB Type A Port

The USB Type A port provides access to external USB flash devices, also known as thumb drives or USB keys. The port supports Cisco USB flash drives with capacities from 128 MB to 8 GB (USB devices with port densities of 128 MB, 256 MB, 1 GB, 4 GB, 8 GB are supported). You can use standard Cisco IOS command- line interface (CLI) commands to read, write, erase, and copy to or from the flash device. You can also configure the switch to boot from the USB flash drive.

Interface Connections

Devices within a single VLAN can communicate directly through any switch. Ports in different VLANs cannot exchange data without going through a routing device. With a standard Layer 2 switch, ports in different VLANs have to exchange information through a router. By using the switch with routing enabled, when you configure both VLAN 20 and VLAN 30 with an SVI to which an IP address is assigned, packets can be sent from Host A to Host B directly through the switch with no need for an external router.

Figure 7: Connecting VLANs with the Switch



When the IP services feature set is running on the switch or the active switch, the switch uses two methods to forward traffic between interfaces: routing and fallback bridging. If the IP base feature set is on the switch or the active switch, only basic routing (static routing and RIP) is supported. Whenever possible, to maintain high performance, forwarding is done by the switch hardware. However, only IPv4 packets with Ethernet II encapsulation are routed in hardware. Non-IP traffic and traffic with other encapsulation methods are fallback-bridged by hardware.

- The routing function can be enabled on all SVIs and routed ports. The switch routes only IP traffic. When IP routing protocol parameters and address configuration are added to an SVI or routed port, any IP traffic received from these ports is routed.
- Fallback bridging forwards traffic that the switch does not route or traffic belonging to a nonroutable protocol, such as DECnet. Fallback bridging connects multiple VLANs into one bridge domain by bridging between two or more SVIs or routed ports. When configuring fallback bridging, you assign SVIs or routed ports to bridge groups with each SVI or routed port assigned to only one bridge group. All interfaces in the same group belong to the same bridge domain.

Interface Configuration Mode

The switch supports these interface types:

- Physical ports—switch ports and routed ports
- VLANs—switch virtual interfaces
- Port channels—EtherChannel interfaces

You can also configure a range of interfaces.

To configure a physical interface (port), specify the interface type, stack member number (only stacking-capable switches), module number, and switch port number, and enter interface configuration mode.

- Type—Gigabit Ethernet (gigabitethernet or gi) for 10/100/1000 Mb/s Ethernet ports, 10-Gigabit Ethernet (tengigabitethernet or te) for 10,000 Mb/s, or small form-factor pluggable (SFP) module Gigabit Ethernet interfaces (gigabitethernet or gi).
- Stack member number—The number that identifies the switch within the stack. The switch number range is 1 to 9 and is assigned the first time the switch initializes. The default switch number, before it is integrated into a switch stack, is 1. When a switch has been assigned a stack member number, it keeps that number until another is assigned to it.

  You can use the switch port LEDs in Stack mode to identify the stack member number of a switch.
- Module number—The module or slot number on the switch: switch (downlink) ports are 0, and uplink ports are 1.
- Port number—The interface number on the switch. The 10/100/1000 port numbers always begin at 1, starting with the far left port when facing the front of the switch, for example, gigabitethernet1/0/1 or gigabitethernet1/0/8.

  On a switch with SFP uplink ports, the module number is 1 and the port numbers restart. For example, if the switch has 24 10/100/1000 ports, the SFP module ports are gigabitethernet1/1/1 through gigabitethernet1/1/4 or tengigabitethernet1/1/1 through tengigabitethernet1/1/4.

You can identify physical interfaces by physically checking the interface location on the switch. You can also use the **show** privileged EXEC commands to display information about a specific interface or all the interfaces on the switch. The remainder of this chapter primarily provides physical interface configuration procedures.

These are examples of how to identify interfaces on a stacking-capable switch:

- To configure 10/100/1000 port 4 on a standalone switch, enter this command:

  ```
  Switch(config)# interface gigabitethernet1/0/4
  ```

- To configure 10-Gigabit Ethernet port 1 on a standalone switch, enter this command:

  ```
  Switch(config)# interface tengigabitethernet1/0/1
  ```

- To configure 10-Gigabit Ethernet port on stack member 3, enter this command:

  ```
  Switch(config)# interface tengigabitethernet3/0/1
  ```

- To configure the first SFP module (uplink) port on a standalone switch, enter this command:

  ```
  Switch(config)# interface gigabitethernet1/1/1
  ```

Default Ethernet Interface Configuration

To configure Layer 2 parameters, if the interface is in Layer 3 mode, you must enter the **switchport** interface configuration command without any parameters to put the interface into Layer 2 mode. This shuts down the interface and then re-enables it, which might generate messages on the device to which the interface is connected. When you put an interface that is in Layer 3 mode into Layer 2 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration.

This table shows the Ethernet interface default configuration, including some features that apply only to Layer 2 interfaces.

Table 47: Default Layer 2 Ethernet Interface Configuration

Feature	Default Setting
Operating mode	Layer 2 or switching mode (**switchport** command).
Allowed VLAN range	VLANs 1– 4094.
Default VLAN (for access ports)	VLAN 1 (Layer 2 interfaces only).
Native VLAN (for IEEE 802.1Q trunks)	VLAN 1 (Layer 2 interfaces only).
VLAN trunking	Switchport mode dynamic auto (supports DTP) (Layer 2 interfaces only).
Port enable state	All ports are enabled.
Port description	None defined.
Speed	Autonegotiate. (Not supported on the 10-Gigabit interfaces.)
Duplex mode	Autonegotiate. (Not supported on the 10-Gigabit interfaces.)
Flow control	Flow control is set to **receive: off**. It is always off for sent packets.
EtherChannel (PAgP)	Disabled on all Ethernet ports.
Port blocking (unknown multicast and unknown unicast traffic)	Disabled (not blocked) (Layer 2 interfaces only).
Broadcast, multicast, and unicast storm control	Disabled.
Protected port	Disabled (Layer 2 interfaces only).
Port security	Disabled (Layer 2 interfaces only).
Port Fast	Disabled.

Feature	Default Setting
Auto-MDIX	Enabled. **Note** The switch might not support a pre-standard powered device—such as Cisco IP phones and access points that do not fully support IEEE 802.3af—if that powered device is connected to the switch through a crossover cable. This is regardless of whether auto-MIDX is enabled on the switch port.
Power over Ethernet (PoE)	Enabled (auto).

Interface Speed and Duplex Mode

Ethernet interfaces on the switch operate at 10, 100, 1000, or 10,000 Mb/s and in either full- or half-duplex mode. In full-duplex mode, two stations can send and receive traffic at the same time. Normally, 10-Mb/s ports operate in half-duplex mode, which means that stations can either receive or send traffic.

Switch models include Gigabit Ethernet (10/100/1000-Mb/s) ports, 10-Gigabit Ethernet ports, and small form-factor pluggable (SFP) module slots supporting SFP modules.

Speed and Duplex Configuration Guidelines

When configuring an interface speed and duplex mode, note these guidelines:

- The 10-Gigabit Ethernet ports do not support the speed and duplex features. These ports operate only at 10,000 Mb/s and in full-duplex mode.
- Gigabit Ethernet (10/100/1000-Mb/s) ports support all speed options and all duplex options (auto, half, and full). However, Gigabit Ethernet ports operating at 1000 Mb/s do not support half-duplex mode.
- For SFP module ports, the speed and duplex CLI options change depending on the SFP module type:
    - The 1000BASE-*x* (where -*x* is -BX, -CWDM, -LX, -SX, and -ZX) SFP module ports support the **nonegotiate** keyword in the **speed** interface configuration command. Duplex options are not supported.
    - The 1000BASE-T SFP module ports support the same speed and duplex options as the 10/100/1000-Mb/s ports.

  For information about which SFP modules are supported on your switch, see the product release notes.
- If both ends of the line support autonegotiation, we highly recommend the default setting of **auto** negotiation.
- If one interface supports autonegotiation and the other end does not, configure duplex and speed on both interfaces; do not use the **auto** setting on the supported side.
- When STP is enabled and a port is reconfigured, the switch can take up to 30 seconds to check for loops. The port LED is amber while STP reconfigures.

Caution Changing the interface speed and duplex mode configuration might shut down and re-enable the interface during the reconfiguration.

IEEE 802.3x Flow Control

Flow control enables connected Ethernet ports to control traffic rates during congestion by allowing congested nodes to pause link operation at the other end. If one port experiences congestion and cannot receive any more traffic, it notifies the other port by sending a pause frame to stop sending until the condition clears. Upon receipt of a pause frame, the sending device stops sending any data packets, which prevents any loss of data packets during the congestion period.

Note The switch ports can receive, but not send, pause frames.

You use the **flowcontrol** interface configuration command to set the interface's ability to **receive** pause frames to **on**, **off**, or **desired**. The default state is **off**.

When set to **desired**, an interface can operate with an attached device that is required to send flow-control packets or with an attached device that is not required to but can send flow-control packets.

These rules apply to flow control settings on the device:

- **receive on** (or **desired**): The port cannot send pause frames but can operate with an attached device that is required to or can send pause frames; the port can receive pause frames.
- **receive off**: Flow control does not operate in either direction. In case of congestion, no indication is given to the link partner, and no pause frames are sent or received by either device.

Note For details on the command settings and the resulting flow control resolution on local and remote ports, see the **flowcontrol** interface configuration command in the command reference for this release.

Layer 3 Interfaces

The switch supports these types of Layer 3 interfaces:

- SVIs: You should configure SVIs for any VLANs for which you want to route traffic. SVIs are created when you enter a VLAN ID following the **interface vlan** global configuration command. To delete an SVI, use the **no interface vlan** global configuration command. You cannot delete interface VLAN 1.

  Note When you create an SVI, it does not become active until it is associated with a physical port.

  When configuring SVIs, you can also configure SVI autostate exclude on a port in the SVI to exclude that port from being included in determining SVI line-state status.

- Routed ports: Routed ports are physical ports configured to be in Layer 3 mode by using the **no switchport** interface configuration command.
- Layer 3 EtherChannel ports: EtherChannel interfaces made up of routed ports.

A Layer 3 switch can have an IP address assigned to each routed port and SVI.

There is no defined limit to the number of SVIs and routed ports that can be configured in a switch or in a switch stack. However, the interrelationship between the number of SVIs and routed ports and the number of other features being configured might have an impact on CPU usage because of hardware limitations. If the switch is using its maximum hardware resources, attempts to create a routed port or SVI have these results:

- If you try to create a new routed port, the switch generates a message that there are not enough resources to convert the interface to a routed port, and the interface remains as a switchport.
- If you try to create an extended-range VLAN, an error message is generated, and the extended-range VLAN is rejected.
- If the switch is notified by VLAN Trunking Protocol (VTP) of a new VLAN, it sends a message that there are not enough hardware resources available and shuts down the VLAN. The output of the **show vlan** user EXEC command shows the VLAN in a suspended state.
- If the switch attempts to boot up with a configuration that has more VLANs and routed ports than hardware can support, the VLANs are created, but the routed ports are shut down, and the switch sends a message that this was due to insufficient hardware resources.

All Layer 3 interfaces require an IP address to route traffic. This procedure shows how to configure an interface as a Layer 3 interface and how to assign an IP address to an interface.

Note If the physical port is in Layer 2 mode (the default), you must enter the **no switchport** interface configuration command to put the interface into Layer 3 mode. Entering a **no switchport** command disables and then re-enables the interface, which might generate messages on the device to which the interface is connected. Furthermore, when you put an interface that is in Layer 2 mode into Layer 3 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration

How to Configure Interface Characteristics

Configuring Interfaces Procedure

These general instructions apply to all interface configuration processes.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Enter the **configure terminal** command at the privileged EXEC prompt:	

	Command or Action	Purpose

	Example:
	`Switch# configure terminal`
	`Enter configuration commands, one per line. End with CNTL/Z.`
	`Switch(config)#`	

Step 2	Enter the **interface** global configuration command. Identify the interface type, the switch number (only on stacking-capable switches), and the number of the connector. In this example, Gigabit Ethernet port 1 on switch 1 is selected:
	Example:
	`Switch(config)# interface gigabitethernet1/0/1`
	`Switch(config-if)#`
		Note You do not need to add a space between the interface type and the interface number. For example, in the preceding line, you can specify either **gigabitethernet 1/0/1**, **gigabitethernet1/0/1**, **gi 1/0/1**, or **gi1/0/1**.

Step 3	Follow each **interface** command with the interface configuration commands that the interface requires. The commands that you enter define the protocols and applications that will run on the interface. The commands are collected and applied to the interface when you enter another interface command or enter **end** to return to privileged EXEC mode.	You can also configure a range of interfaces by using the **interface range** or **interface range macro** global configuration commands. Interfaces configured in a range must be the same type and must be configured with the same feature options.
Step 4	After you configure an interface, verify its status by using the **show** privileged EXEC commands. **Example:**	Enter the **show interfaces** privileged EXEC command to see a list of all interfaces on or configured for the switch. A report is provided for each interface that the device supports or for the specified interface.

Adding a Description for an Interface

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **description** *string*
4. **end**
5. **show interfaces** *interface-id* **description**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/2`	Specifies the interface for which you are adding a description, and enter interface configuration mode.
Step 3	**description** *string* **Example:** `Switch(config-if)# description Connects to Marketing`	Adds a description (up to 240 characters) for an interface.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show interfaces** *interface-id* **description**	Verifies your entry.

Configuring a Range of Interfaces

To configure multiple interfaces with the same configuration parameters, use the **interface range** global configuration command. When you enter the interface-range configuration mode, all command parameters that you enter are attributed to all interfaces within that range until you exit this mode.

SUMMARY STEPS

1. **configure terminal**
2. **interface range** {*port-range* | **macro** *macro_name*}
3. **end**
4. **show interfaces** [*interface-id*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface range** {*port-range* | **macro** *macro_name*} **Example:** `Switch(config)# interface range macro`	Specifies the range of interfaces (VLANs or physical ports) to be configured, and enter interface-range configuration mode. • You can use the **interface range** command to configure up to five port ranges or a previously defined macro. • The **macro** variable is explained in the Configuring and Using Interface Range Macros, on page 396. • In a comma-separated *port-range*, you must enter the interface type for each entry and enter spaces before and after the comma. • In a hyphen-separated *port-range*, you do not need to re-enter the interface type, but you must enter a space before the hyphen. **Note** Use the normal configuration commands to apply the configuration parameters to all interfaces in the range. Each command is executed as it is entered.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show interfaces** [*interface-id*] **Example:** `Switch# show interfaces`	Verifies the configuration of the interfaces in the range.

Configuring and Using Interface Range Macros

You can create an interface range macro to automatically select a range of interfaces for configuration. Before you can use the **macro** keyword in the **interface range macro** global configuration command string, you must use the **define interface-range** global configuration command to define the macro.

SUMMARY STEPS

1. **configure terminal**
2. **define interface-range** *macro_name interface-range*
3. **interface range macro** *macro_name*
4. **end**
5. **show running-config | include define**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**define interface-range** *macro_name interface-range* **Example:** `Switch(config)# define interface-range enet_list gigabitethernet1/0/1 - 2`	Defines the interface-range macro, and save it in NVRAM. • The *macro_name* is a 32-character maximum character string. • A macro can contain up to five comma-separated interface ranges. • Each *interface-range* must consist of the same port type. **Note** Before you can use the **macro** keyword in the **interface range macro** global configuration command string, you must use the **define interface-range** global configuration command to define the macro.
Step 3	**interface range macro** *macro_name* **Example:** `Switch(config)# interface range macro enet_list`	Selects the interface range to be configured using the values saved in the interface-range macro called *macro_name*. You can now use the normal configuration commands to apply the configuration to all interfaces in the defined macro.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config | include define** **Example:** `Switch# show running-config | include define`	Shows the defined interface range macro configuration.

Configuring Ethernet Interfaces

Setting the Interface Speed and Duplex Parameters

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **speed** {**10** | **100** | **1000** | **auto** [**10** | **100** | **1000**] | **nonegotiate**}
4. **duplex** {**auto** | **full** | **half**}
5. **end**
6. **show interfaces** *interface-id*
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/3`	Specifies the physical interface to be configured, and enter interface configuration mode.
Step 3	**speed** {**10** | **100** | **1000** | **auto** [**10** | **100** | **1000**] | **nonegotiate**} **Example:** `Switch(config-if)# speed 10`	This command is not available on a 10-Gigabit Ethernet interface. Enter the appropriate speed parameter for the interface: • Enter **10**, **100**, or **1000** to set a specific speed for the interface. The **1000** keyword is available only for 10/100/1000 Mb/s ports. • Enter **auto** to enable the interface to autonegotiate speed with the connected device. If you use the **10**, **100**, or the **1000** keywords with the **auto** keyword, the port autonegotiates only at the specified speeds. • The **nonegotiate** keyword is available only for SFP module ports. SFP module ports operate only at 1000 Mb/s but can be configured

	Command or Action	Purpose
		to not negotiate if connected to a device that does not support autonegotiation.
Step 4	**duplex** {**auto** | **full** | **half**} **Example:** `Switch(config-if)# duplex half`	This command is not available on a 10-Gigabit Ethernet interface. Enter the duplex parameter for the interface. Enable half-duplex mode (for interfaces operating only at 10 or 100 Mb/s). You cannot configure half-duplex mode for interfaces operating at 1000 Mb/s. You can configure the duplex setting when the speed is set to **auto**.
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**show interfaces** *interface-id* **Example:** `Switch# show interfaces gigabitethernet1/0/3`	Displays the interface speed and duplex mode configuration.
Step 7	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring IEEE 802.3x Flow Control

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **flowcontrol** {**receive**} {**on** | **off** | **desired**}
4. **end**
5. **show interfaces** *interface-id*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Specifies the physical interface to be configured, and enter interface configuration mode.
Step 3	**flowcontrol** {**receive**} {**on** | **off** | **desired**} **Example:** `Switch(config-if)# flowcontrol receive on`	Configures the flow control mode for the port.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show interfaces** *interface-id* **Example:** `Switch# show interfaces gigabitethernet1/0/1`	Verifies the interface flow control settings.

Configuring Layer 3 Interfaces

SUMMARY STEPS

1. **configure terminal**
2. **interface** {**gigabitethernet** *interface-id*} | {**vlan** *vlan-id*} | {**port-channel** *port-channel-number*}
3. **no switchport**
4. **ip address** *ip_address subnet_mask*
5. **no shutdown**
6. **end**
7. **show interfaces** [*interface-id*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** {**gigabitethernet** *interface-id*} | {**vlan** *vlan-id*} | {**port-channel** *port-channel-number*} **Example:** `Switch(config)# interface gigabitethernet1/0/2`	Specifies the interface to be configured as a Layer 3 interface, and enter interface configuration mode.
Step 3	**no switchport** **Example:** `Switch(config-if)# no switchport`	For physical ports only, enters Layer 3 mode.
Step 4	**ip address** *ip_address subnet_mask* **Example:** `Switch(config-if)# ip address 192.20.135.21 255.255.255.0`	Configures the IP address and IP subnet.
Step 5	**no shutdown** **Example:** `Switch(config-if)# no shutdown`	Enables the interface.
Step 6	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 7	**show interfaces** [*interface-id*]	Verifies the configuration.

Configuring SVI Autostate Exclude

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport autostate exclude**
4. **end**
5. **show running config interface** *interface-id*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/2`	Specifies a Layer 2 interface (physical port or port channel), and enter interface configuration mode.
Step 3	**switchport autostate exclude** **Example:** `Switch(config-if)# switchport autostate exclude`	Excludes the access or trunk port when defining the status of an SVI line state (up or down)
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show running config interface** *interface-id*	(Optional) Shows the running configuration. Verifies the configuration.

Shutting Down and Restarting the Interface

Shutting down an interface disables all functions on the specified interface and marks the interface as unavailable on all monitoring command displays. This information is communicated to other network servers through all dynamic routing protocols. The interface is not mentioned in any routing updates.

SUMMARY STEPS

1. **configure terminal**
2. **interface** {**vlan** *vlan-id*} | {**gigabitethernet** *interface-id*} | {**port-channel** *port-channel-number*}
3. **shutdown**
4. **no shutdown**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** {**vlan** *vlan-id*} | {**gigabitethernet** *interface-id*} | {**port-channel** *port-channel-number*} **Example:** `Switch(config)# interface gigabitethernet1/0/2`	Selects the interface to be configured.
Step 3	**shutdown** **Example:** `Switch(config-if)# shutdown`	Shuts down an interface.
Step 4	**no shutdown** **Example:** `Switch(config-if)# no shutdown`	Restarts an interface.
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Configuring the Console Media Type

Beginning in privileged EXEC mode, follow these steps to set the console media type to RJ-45. If you configure the console as RJ-45, USB console operation is disabled, and input comes only through the RJ-45 connector.

This configuration applies to all switches in a stack.

SUMMARY STEPS

1. **configure terminal**
2. **line console 0**
3. **media-type rj45**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**line console 0** **Example:** `Switch(config)# line console 0`	Configures the console and enters line configuration mode.
Step 3	**media-type rj45** **Example:** `Switch(config-line)# media-type rj45`	Configures the console media type to be only RJ-45 port. If you do not enter this command and both types are connected, the USB port is used by default.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring the USB Inactivity Timeout

The configurable inactivity timeout reactivates the RJ-45 console port if the USB console port is activated but no input activity occurs on it for a specified time period. When the USB console port is deactivated due to a timeout, you can restore its operation by disconnecting and reconnecting the USB cable.

Note The configured inactivity timeout applies to all switches in a stack. However, a timeout on one switch does not cause a timeout on other switches in the stack.

Beginning in privileged EXEC mode, follow these steps to configure an inactivity timeout.

SUMMARY STEPS

1. **configure terminal**
2. **line console 0**
3. **usb-inactivity-timeout** *timeout-minutes*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**line console 0** **Example:** `Switch(config)# line console 0`	Configures the console and enters line configuration mode.
Step 3	**usb-inactivity-timeout** *timeout-minutes* **Example:** `Switch(config-line)# usb-inactivity-timeout 30`	Specify an inactivity timeout for the console port. The range is 1 to 240 minutes. The default is to have no timeout configured.

Monitoring Interface Characteristics

Monitoring Interface Status

Commands entered at the privileged EXEC prompt display information about the interface, including the versions of the software and the hardware, the configuration, and statistics about the interfaces.

This table lists some of the available interface monitoring commands.

Table 48: Show Commands for Interfaces

Command	Purpose
show interfaces [*interface-id*]	Displays the status and configuration of all interfaces or a specific interface.
show interfaces *interface-id* **status** [**err-disabled**]	Displays interface status or a list of interfaces in the error-disabled state.
show interfaces [*interface-id*] **switchport**	Displays administrative and operational status of switching (nonrouting) ports. You can use this command to find out if a port is in routing or in switching mode.
show interfaces [*interface-id*] **description**	Displays the description configured on an interface or all interfaces and the interface status.
show ip interface [*interface-id*]	Displays the usability status of all interfaces configured for IP routing or the specified interface.
show interface [*interface-id*] **stats**	Displays the input and output packets by the switching path for the interface.
show interfaces *interface-id*	(Optional) Displays speed and duplex on the interface.
show interfaces transceiver dom-supported-list	(Optional) Displays Digital Optical Monitoring (DOM) status on the connect SFP modules.
show interfaces transceiver properties	(Optional) Displays temperature, voltage, or amount of current on the interface.
show interfaces [*interface-id*] [{**transceiver properties** | **detail**}] *module number*]	Displays physical and operational status about an SFP module.
show running-config interface [*interface-id*]	Displays the running configuration in RAM for the interface.

Command	Purpose
show version	Displays the hardware configuration, software version, the names and sources of configuration files, and the boot images.
show controllers ethernet-controller *interface-id* **phy**	Displays the operational state of the auto-MDIX feature on the interface.

Clearing and Resetting Interfaces and Counters

Table 49: Clear Commands for Interfaces

Command	Purpose
clear counters [*interface-id*]	Clears interface counters.
clear interface *interface-id*	Resets the hardware logic on an interface.
clear line [*number* | **console 0** | **vty** *number*]	Resets the hardware logic on an asynchronous serial line.

Note The **clear counters** privileged EXEC command does not clear counters retrieved by using Simple Network Management Protocol (SNMP), but only those seen with the **show interface** privileged EXEC command.

Configuration Examples for Interface Characteristics

Adding a Description to an Interface: Example

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTRL/Z.
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# description Connects to Marketing
Switch(config-if)# end
Switch# show interfaces gigabitethernet1/0/2 description
Interface Status         Protocol Description
Gi1/0/2   admin down     down     Connects to Marketing
```

Identifying Interfaces on a Stack-Capable Switch: Examples

To configure 10/100/1000 port 4 on a standalone switch, enter this command:

```
Switch(config)# interface gigabitethernet1/0/4
```

To configure 10-Gigabit Ethernet port 1 on a standalone switch, enter this command:

```
Switch(config)# interface tengigabitethernet1/0/1
```

To configure 10-Gigabit Ethernet port on stack member 3, enter this command:

```
Switch(config)# interface tengigabitethernet3/0/1
```

To configure the first SFP module uplink port on stack member 1, enter this command:

```
Switch(config)# interface gigabitethernet1/1/1
```

Configuring a Range of Interfaces: Examples

This example shows how to use the **interface range** global configuration command to set the speed to 100 Mb/s on ports 1 to 4 on switch 1:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet1/0/1 - 4
Switch(config-if-range)# speed 100
```

This example shows how to use a comma to add different interface type strings to the range to enable Gigabit Ethernet ports 1 to 3 and 10-Gigabit Ethernet ports 1 and 2 to receive flow-control pause frames:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet1/0/1 - 3 , tengigabitethernet1/0/1 - 2
Switch(config-if-range)# flowcontrol receive on
```

If you enter multiple configuration commands while you are in interface-range mode, each command is executed as it is entered. The commands are not batched and executed after you exit interface-range mode. If you exit interface-range configuration mode while the commands are being executed, some commands might not be executed on all interfaces in the range. Wait until the command prompt reappears before exiting interface-range configuration mode.

Configuring and Using Interface Range Macros: Examples

This example shows how to define an interface-range named *enet_list* to include ports 1 and 2 on switch 1 and to verify the macro configuration:

```
Switch# configure terminal
Switch(config)# define interface-range enet_list gigabitethernet1/0/1 - 2
Switch(config)# end
Switch# show running-config | include define
define interface-range enet_list GigabitEthernet1/0/1 - 2
```

This example shows how to create a multiple-interface macro named *macro1*:

```
Switch# configure terminal
```

```
Switch(config)# define interface-range macro1 gigabitethernet1/0/1 - 2, gigabitethernet1/0/5
 - 7, tengigabitethernet1/0/1 -2
Switch(config)# end
```

This example shows how to enter interface-range configuration mode for the interface-range macro *enet_list*:

```
Switch# configure terminal
Switch(config)# interface range macro enet_list
Switch(config-if-range)#
```

This example shows how to delete the interface-range macro *enet_list* and to verify that it was deleted.

```
Switch# configure terminal
Switch(config)# no define interface-range enet_list
Switch(config)# end
Switch# show run | include define
Switch#
```

Setting Interface Speed and Duplex Mode: Example

This example shows how to set the interface speed to 100 Mb/s and the duplex mode to half on a 10/100/1000 Mb/s port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/3
Switch(config-if)# speed 10
Switch(config-if)# duplex half
```

This example shows how to set the interface speed to 100 Mb/s on a 10/100/1000 Mb/s port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# speed 100
```

Configuring Layer 3 Interfaces: Example

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# no switchport
Switch(config-if)# ip address 192.20.135.21 255.255.255.0
Switch(config-if)# no shutdown
```

Configuring the Console Media Type: Example

This example disables the USB console media type and enables the RJ-45 console media type.

```
Switch# configure terminal
Switch(config)# line console 0
Switch(config-line)# media-type rj45
```

This configuration terminates any active USB console media type in the stack. A log shows that this termination has occurred. This example shows that the console on switch 1 reverted to RJ-45.

```
*Mar  1 00:25:36.860: %USB_CONSOLE-6-CONFIG_DISABLE: Console media-type USB disabled by
system configuration, media-type reverted to RJ45.
```

At this point no switches in the stack allow a USB console to have input. A log entry shows when a console cable is attached. If a USB console cable is connected to switch 2, it is prevented from providing input.

```
*Mar  1 00:34:27.498: %USB_CONSOLE-6-CONFIG_DISALLOW: Console media-type USB is disallowed
 by system configuration, media-type remains RJ45. (switch-stk-2)
```

This example reverses the previous configuration and immediately activates any USB console that is connected.

```
Switch# configure terminal
Switch(config)# line console 0
Switch(config-line)# no media-type rj45
```

Configuring the USB Inactivity Timeout: Example

This example configures the inactivity timeout to 30 minutes:

```
Switch# configure terminal
Switch(config)# line console 0
Switch(config-line)# usb-inactivity-timeout 30
```

To disable the configuration, use these commands:

```
Switch# configure terminal
Switch(config)# line console 0
Switch(config-line)# no usb-inactivity-timeout
```

If there is no (input) activity on a USB console port for the configured number of minutes, the inactivity timeout setting applies to the RJ-45 port, and a log shows this occurrence:

```
*Mar  1 00:47:25.625: %USB_CONSOLE-6-INACTIVITY_DISABLE: Console media-type USB disabled
due to inactivity, media-type reverted to RJ45.
```

At this point, the only way to reactivate the USB console port is to disconnect and reconnect the cable.

When the USB cable on the switch has been disconnected and reconnected, a log similar to this appears:

```
*Mar  1 00:48:28.640: %USB_CONSOLE-6-MEDIA_USB: Console media-type is USB.
```

Additional References for the Interface Characteristics Feature

Related Documents

Related Topic	Document Title
Platform-independent command reference	*Interface and Hardware Command Reference, Cisco IOS XE Release 3.2SE (Catalyst 3850 Switches)*
Platform-independent configuration information	*Interface and Hardware Component Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	--

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Configuring Interface Characteristics

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 20

Configuring Auto-MDIX

Prerequisites for Auto-MDIX

To configure Layer 2 parameters, if the interface is in Layer 3 mode, you must enter the **switchport** interface configuration command without any parameters to put the interface into Layer 2 mode. This shuts down the interface and then re-enables it, which might generate messages on the device to which the interface is connected. When you put an interface that is in Layer 3 mode into Layer 2 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration.

Automatic medium-dependent interface crossover (auto-MDIX) is enabled by default. When you enable auto-MDIX, you must also set the interface speed and duplex to **auto** so that the feature operates correctly.

Auto-MDIX is supported on all 10/100/1000-Mb/s and on 10/100/1000BASE-TX small form-factor pluggable (SFP)-module interfaces. It is not supported on 1000BASE-SX or -LX SFP module interfaces.

Restrictions for Auto-MDIX

The switch might not support a pre-standard powered device—such as Cisco IP phones and access points that do not fully support IEEE 802.3af—if that powered device is connected to the switch through a crossover cable. This is regardless of whether auto-MIDX is enabled on the switch port.

Information about Configuring Auto-MDIX

Auto-MDIX on an Interface

When automatic medium-dependent interface crossover (auto-MDIX) is enabled on an interface, the interface automatically detects the required cable connection type (straight through or crossover) and configures the connection appropriately. When connecting switches without the auto-MDIX feature, you must use straight-through cables to connect to devices such as servers, workstations, or routers and crossover cables to connect to other switches or repeaters. With auto-MDIX enabled, you can use either type of cable to connect to other devices, and the interface automatically corrects for any incorrect cabling. For more information about cabling requirements, see the hardware installation guide.

This table shows the link states that result from auto-MDIX settings and correct and incorrect cabling.

Table 50: Link Conditions and Auto-MDIX Settings

Local Side Auto-MDIX	Remote Side Auto-MDIX	With Correct Cabling	With Incorrect Cabling
On	On	Link up	Link up
On	Off	Link up	Link up
Off	On	Link up	Link up
Off	Off	Link up	Link down

How to Configure Auto-MDIX

Configuring Auto-MDIX on an Interface

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **speed auto**
4. **duplex auto**
5. **mdix auto**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch#` **configure terminal**	Enters global configuration mode
Step 2	**interface** *interface-id* **Example:** `Switch(config)#` **interface gigabitethernet1/0/1**	Specifies the physical interface to be configured, and enter interface configuration mode.
Step 3	**speed auto** **Example:** `Switch(config-if)#` **speed auto**	Configures the interface to autonegotiate speed with the connected device.
Step 4	**duplex auto** **Example:** `Switch(config-if)#` **duplex auto**	Configures the interface to autonegotiate duplex mode with the connected device.
Step 5	**mdix auto** **Example:** `Switch(config-if)#` **mdix auto**	Enables auto-MDIX on the interface.
Step 6	**end** **Example:** `Switch(config-if)#` **end**	Returns to privileged EXEC mode.

Monitoring Auto-MDIX

Command	Purpose

show controllers ethernet-controller*interface-id* **phy**	Verifies the operational state of the auto-MDIX feature on the interface.

Example for Configuring Auto-MDIX

This example shows how to enable auto-MDIX on a port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# speed auto
Switch(config-if)# duplex auto
Switch(config-if)# mdix auto
Switch(config-if)# end
```

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Auto-MDIX

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 21

Configuring Ethernet Management Port

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for Ethernet Management Ports

When connecting a PC to the Ethernet management port, you must first assign an IP address.

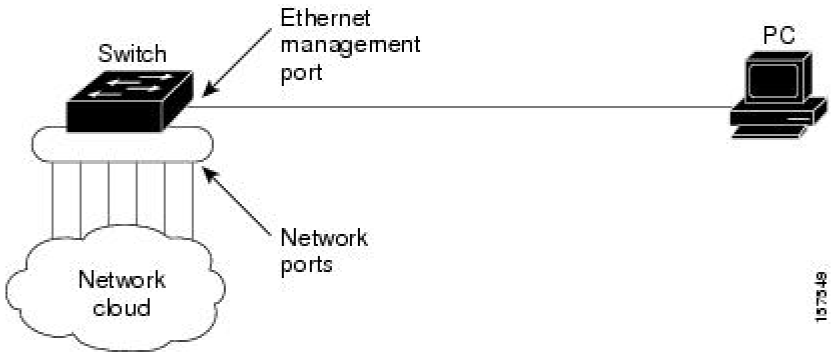
Information about the Ethernet Management Port

The Ethernet management port, also referred to as the *Gi0/0* or *GigabitEthernet0/0* port, is a VRF (VPN routing/forwarding) interface to which you can connect a PC. You can use the Ethernet management port instead of the switch console port for network management. When managing a switch stack, connect the PC to the Ethernet management port on a stack member.

Ethernet Management Port Direct Connection to a Switch

This figure displays how to connect the Ethernet management port to the PC for a switch or a standalone switch.

Figure 8: Connecting a Switch to a PC

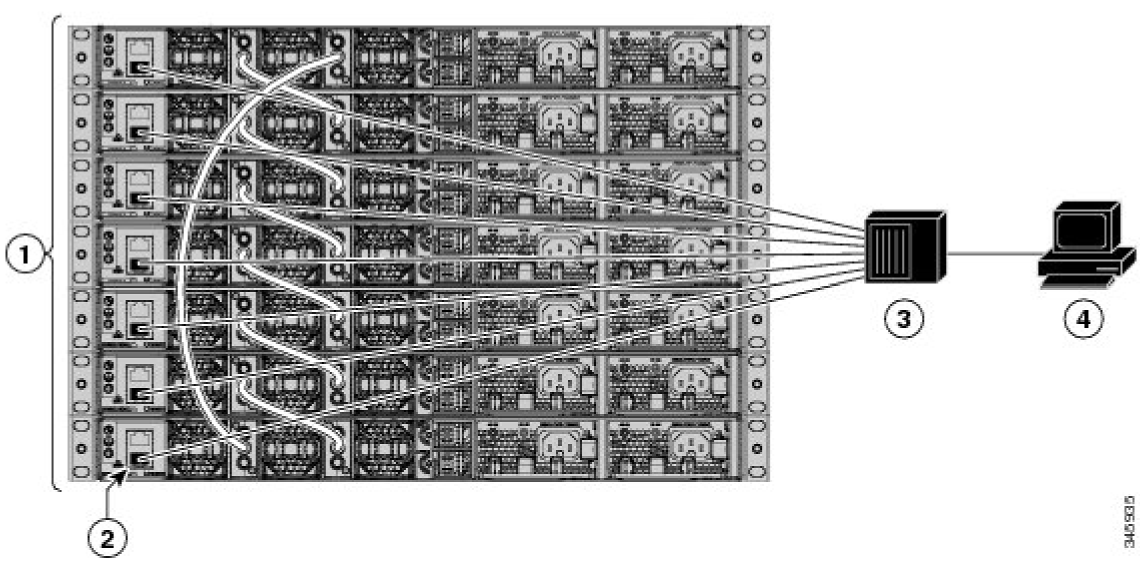


Ethernet Management Port Connection to Stack Switches using a Hub

In a stack with only stack switches, all the Ethernet management ports on the stack members are connected to a hub to which the PC is connected. The active link is from the Ethernet management port on the through the hub, to the PC. If the active switch fails and a new active switch is elected, the active link is now from the Ethernet management port on the new active switch to the PC.

This figure displays how a PC uses a hub to connect to a switch stack.

Figure 9: Connecting a Switch Stack to a PC



1	Switch stack	3	Hub

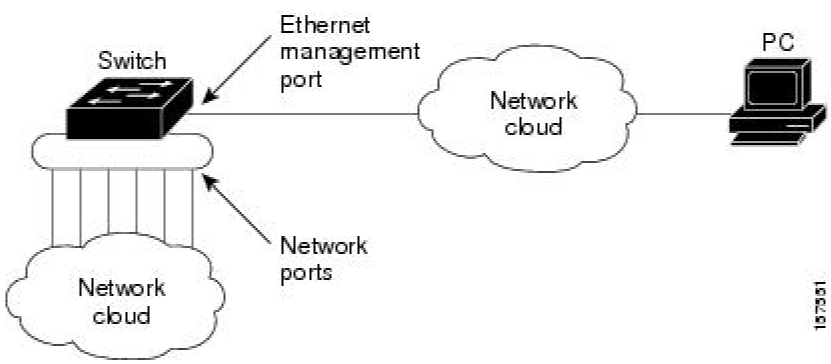
2	Management port	4	PC

Ethernet Management Port and Routing

By default, the Ethernet management port is enabled. The switch cannot route packets from the Ethernet management port to a network port, and the reverse. Even though the Ethernet management port does not support routing, you may need to enable routing protocols on the port.

In the following figure, you must enable routing protocols on the Ethernet management port when the PC is multiple hops away from the switch and the packets must pass through multiple Layer 3 devices to reach the PC.

Figure 10: Network Example with Routing Protocols Enabled



In the above figure , if the Ethernet management port and the network ports are associated with the same routing process, the routes are propagated as follows:

- The routes from the Ethernet management port are propagated through the network ports to the network.
- The routes from the network ports are propagated through the Ethernet management port to the network.

Because routing is not supported between the Ethernet management port and the network ports, traffic between these ports cannot be sent or received. If this happens, data packet loops occur between the ports, which disrupt the switch and network operation. To prevent the loops, configure route filters to avoid routes between the Ethernet management port and the network ports.

Supported Features on the Ethernet Management Port

The Ethernet management port supports these features:

- Express Setup (only in switch stacks)
- Network Assistant
- Telnet with passwords
- TFTP
- Secure Shell (SSH)
- DHCP-based autoconfiguration

- SMNP (only the ENTITY-MIB and the IF-MIB)
- IP ping
- Interface features
    - Speed—10 Mb/s, 100 Mb/s, 1000 Mb/s, and autonegotiation
    - Duplex mode—Full, half, and autonegotiation
    - Loopback detection
- Cisco Discovery Protocol (CDP)
- DHCP relay agent
- IPv4 and IPv6 access control lists (ACLs)
- Routing protocols

Caution Before enabling a feature on the Ethernet management port, make sure that the feature is supported. If you try to configure an unsupported feature on the Ethernet Management port, the feature might not work properly, and the switch might fail.

How to Configure the Ethernet Management Port

Disabling and Enabling the Ethernet Management Port

To disable or enable the Ethernet management port in the CLI, follow this procedure.

SUMMARY STEPS

1. **configure terminal**
2. **interface gigabitethernet0/0**
3. **shutdown**
4. **no shutdown**
5. **exit**
6. **show interfaces gigabitethernet0/0**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**interface gigabitethernet0/0** **Example:** `Switch(config)# interface gigabitethernet0/0`	Specifies the Ethernet management port in the CLI.
Step 3	**shutdown** **Example:** `Switch(config-if)# shutdown`	Disables the Ethernet management port.
Step 4	**no shutdown** **Example:** `Switch(config-if)# no shutdown`	Enables the Ethernet management port.
Step 5	**exit** **Example:** `Switch(config-if)# exit`	Exits interface configuration mode.
Step 6	**show interfaces gigabitethernet0/0** **Example:** `Switch# show interfaces gigabitethernet0/0`	Displays the link status. To find out the link status to the PC, you can monitor the LED for the Ethernet management port. The LED is green (on) when the link is active, and the LED is off when the link is down. The LED is amber when there is a POST failure.

What to Do Next

Proceed to manage or configure your switch using the Ethernet management port. Refer to the *Network Management Configuration Guide (Catalyst 3850 Switches)*.

Additional References

Related Documents

Related Topic	Document Title
Bootloader configuration	*System Management Configuration Guide (Catalyst 3850 Switches)*
Bootloader commands	*System Management Command Reference (Catalyst 3850 Switches)*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for Ethernet Management Ports

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 22

Configuring LLDP, LLDP-MED, and Wired Location Service

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

LLDP, LLDP-MED, and Wired Location Service Overview

LLDP

The Cisco Discovery Protocol (CDP) is a device discovery protocol that runs over Layer 2 (the data link layer) on all Cisco-manufactured devices (routers, bridges, access servers, and switches). CDP allows network management applications to automatically discover and learn about other Cisco devices connected to the network.

To support non-Cisco devices and to allow for interoperability between other devices, the switch supports the IEEE 802.1AB Link Layer Discovery Protocol (LLDP). LLDP is a neighbor discovery protocol that is used for network devices to advertise information about themselves to other devices on the network. This protocol runs over the data-link layer, which allows two systems running different network layer protocols to learn about each other.

LLDP Supported TLVs

LLDP supports a set of attributes that it uses to discover neighbor devices. These attributes contain type, length, and value descriptions and are referred to as TLVs. LLDP supported devices can use TLVs to receive and send information to their neighbors. This protocol can advertise details such as configuration information, device capabilities, and device identity.

The switch supports these basic management TLVs. These are mandatory LLDP TLVs.

- Port description TLV
- System name TLV
- System description TLV
- System capabilities TLV
- Management address TLV

These organizationally specific LLDP TLVs are also advertised to support LLDP-MED.

- Port VLAN ID TLV (IEEE 802.1 organizationally specific TLVs)
- MAC/PHY configuration/status TLV (IEEE 802.3 organizationally specific TLVs)

LLDP and Cisco Switch Stacks

A switch stack appears as a single switch in the network. Therefore, LLDP discovers the switch stack, not the individual stack members.

LLDP and Cisco Medianet

When you configure LLDP or CDP location information on a per-port basis, remote devices can send Cisco Medianet location information to the switch. For information, go to http://www.cisco.com/en/US/docs/ios/netmgmt/configuration/guide/nm_cdp_discover.html.

LLDP-MED

LLDP for Media Endpoint Devices (LLDP-MED) is an extension to LLDP that operates between endpoint devices such as IP phones and network devices such as switches. It specifically provides support for voice over IP (VoIP) applications and provides additional TLVs for capabilities discovery, network policy, Power over Ethernet, inventory management and location information. By default, all LLDP-MED TLVs are enabled.

LLDP-MED Supported TLVs

LLDP-MED supports these TLVs:

- LLDP-MED capabilities TLV

  Allows LLDP-MED endpoints to determine the capabilities that the connected device supports and has enabled.

- Network policy TLV

  Allows both network connectivity devices and endpoints to advertise VLAN configurations and associated Layer 2 and Layer 3 attributes for the specific application on that port. For example, the switch can notify a phone of the VLAN number that it should use. The phone can connect to any switch, obtain its VLAN number, and then start communicating with the call control.

  By defining a network-policy profile TLV, you can create a profile for voice and voice-signaling by specifying the values for VLAN, class of service (CoS), differentiated services code point (DSCP), and tagging mode. These profile attributes are then maintained centrally on the switch and propagated to the phone.

- Power management TLV

  Enables advanced power management between LLDP-MED endpoint and network connectivity devices. Allows switches and phones to convey power information, such as how the device is powered, power priority, and how much power the device needs.

  LLDP-MED also supports an extended power TLV to advertise fine-grained power requirements, end-point power priority, and end-point and network connectivity-device power status. LLDP is enabled and power is applied to a port, the power TLV determines the actual power requirement of the endpoint device so that the system power budget can be adjusted accordingly. The switch processes the requests and either grants or denies power based on the current power budget. If the request is granted, the switch updates the power budget. If the request is denied, the switch turns off power to the port, generates a syslog message, and updates the power budget. If LLDP-MED is disabled or if the endpoint does not support the LLDP-MED power TLV, the initial allocation value is used throughout the duration of the connection.

  You can change power settings by entering the **power inline** {**auto** [**max** *max-wattage*] | **never** | **static** [**max** *max-wattage*]} interface configuration command. By default the PoE interface is in **auto** mode; If no value is specified, the maximum is allowed (30 W).

- Inventory management TLV

  Allows an endpoint to send detailed inventory information about itself to the switch, including information hardware revision, firmware version, software version, serial number, manufacturer name, model name, and asset ID TLV.

- Location TLV

  Provides location information from the switch to the endpoint device. The location TLV can send this information:

  - Civic location information

    Provides the civic address information and postal information. Examples of civic location information are street address, road name, and postal community name information.

  - ELIN location information

Provides the location information of a caller. The location is determined by the Emergency location identifier number (ELIN), which is a phone number that routes an emergency call to the local public safety answering point (PSAP) and which the PSAP can use to call back the emergency caller.

◦ Geographic location information

Provides the geographical details of a switch location such as latitude, longitude, and altitude of a switch.

◦ custom location

Provides customized name and value of a switch location.

Wired Location Service

The switch uses the location service feature to send location and attachment tracking information for its connected devices to a Cisco Mobility Services Engine (MSE). The tracked device can be a wireless endpoint, a wired endpoint, or a wired switch or controller. The switch notifies the MSE of device link up and link down events through the Network Mobility Services Protocol (NMSP) location and attachment notifications.

The MSE starts the NMSP connection to the switch, which opens a server port. When the MSE connects to the switch there are a set of message exchanges to establish version compatibility and service exchange information followed by location information synchronization. After connection, the switch periodically sends location and attachment notifications to the MSE. Any link up or link down events detected during an interval are aggregated and sent at the end of the interval.

When the switch determines the presence or absence of a device on a link-up or link-down event, it obtains the client-specific information such as the MAC address, IP address, and username. If the client is LLDP-MED- or CDP-capable, the switch obtains the serial number and UDI through the LLDP-MED location TLV or CDP.

Depending on the device capabilities, the switch obtains this client information at link up:

- Slot and port specified in port connection
- MAC address specified in the client MAC address
- IP address specified in port connection
- 802.1X username if applicable
- Device category is specified as a *wired station*
- State is specified as *new*
- Serial number, UDI
- Model number
- Time in seconds since the switch detected the association

Depending on the device capabilities, the switch obtains this client information at link down:

- Slot and port that was disconnected
- MAC address
- IP address

- 802.1X username if applicable
- Device category is specified as a *wired station*
- State is specified as *delete*
- Serial number, UDI
- Time in seconds since the switch detected the disassociation

When the switch shuts down, it sends an attachment notification with the state *delete* and the IP address before closing the NMSP connection to the MSE. The MSE interprets this notification as disassociation for all the wired clients associated with the switch.

If you change a location address on the switch, the switch sends an NMSP location notification message that identifies the affected ports and the changed address information.

Default LLDP Configuration

Table 51: Default LLDP Configuration

Feature	Default Setting
LLDP global state	Disabled
LLDP holdtime (before discarding)	120 seconds
LLDP timer (packet update frequency)	30 seconds
LLDP reinitialization delay	2 seconds
LLDP tlv-select	Disabled to send and receive all TLVs
LLDP interface state	Disabled
LLDP receive	Disabled
LLDP transmit	Disabled
LLDP med-tlv-select	Disabled to send all LLDP-MED TLVs. When LLDP is globally enabled, LLDP-MED-TLV is also enabled.

Configuration Guidelines

- If the interface is configured as a tunnel port, LLDP is automatically disabled.
- If you first configure a network-policy profile on an interface, you cannot apply the **switchport voice vlan** command on the interface. If the **switchport voice vlan** *vlan-id* is already configured on an interface, you can apply a network-policy profile on the interface. This way the interface has the voice or voice-signaling VLAN network-policy profile applied on the interface.

- You cannot configure static secure MAC addresses on an interface that has a network-policy profile.
- For wired location to function, you must first enter the **ip device tracking** global configuration command.

How to Configure LLDP, LLDP-MED, and Wired Location Service

Enabling LLDP

Beginning in privileged EXEC mode, follow these steps to enable LLDP:

SUMMARY STEPS

1. **configure terminal**
2. **lldp run**
3. **interface** *interface-id*
4. **lldp transmit**
5. **lldp receive**
6. **end**
7. **show lldp**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**lldp run** **Example:** `Switch (config)# lldp run`	Enables LLDP globally on the switch.
Step 3	**interface** *interface-id* **Example:** `Switch (config)# interface gigabitethernet2/0/1`	Specifies the interface on which you are enabling LLDP, and enter interface configuration mode.

	Command or Action	Purpose
Step 4	**lldp transmit** **Example:** `Switch(config-if)# lldp transmit`	Enables the interface to send LLDP packets.
Step 5	**lldp receive** **Example:** `Switch(config-if)# lldp receive`	Enables the interface to receive LLDP packets.
Step 6	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 7	**show lldp** **Example:** `Switch# show lldp`	Verifies the configuration.
Step 8	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring LLDP Characteristics

You can configure the frequency of LLDP updates, the amount of time to hold the information before discarding it, and the initialization delay time. You can also select the LLDP and LLDP-MED TLVs to send and receive.

Beginning in privileged EXEC mode, follow these steps to configure the LLDP characteristics.

Note Steps 2 through 5 are optional and can be performed in any order.

SUMMARY STEPS

1. **configure terminal**
2. **lldp holdtime** *seconds*
3. **lldp reinit** *delay*
4. **lldp timer** *rate*
5. **lldp tlv-select**
6. **interface** *interface-id*
7. **lldp med-tlv-select**
8. **end**
9. **show lldp**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**lldp holdtime** *seconds* **Example:** `Switch(config)# lldp holdtime 120`	(Optional) Specifies the amount of time a receiving device should hold the information from your device before discarding it. The range is 0 to 65535 seconds; the default is 120 seconds.
Step 3	**lldp reinit** *delay* **Example:** `Switch(config)# lldp reinit 2`	(Optional) Specifies the delay time in seconds for LLDP to initialize on an interface. The range is 2 to 5 seconds; the default is 2 seconds.
Step 4	**lldp timer** *rate* **Example:** `Switch(config)# lldp timer 30`	(Optional) Sets the sending frequency of LLDP updates in seconds. The range is 5 to 65534 seconds; the default is 30 seconds.
Step 5	**lldp tlv-select** **Example:** `Switch(config)# tlv-select`	(Optional) Specifies the LLDP TLVs to send or receive.

	Command or Action	Purpose
Step 6	**interface** *interface-id* **Example:** `Switch (config)# interface gigabitethernet2/0/1`	Specifies the interface on which you are enabling LLDP, and enter interface configuration mode.
Step 7	**lldp med-tlv-select** **Example:** `Switch (config-if)# lldp med-tlv-select inventory management`	(Optional) Specifies the LLDP-MED TLVs to send or receive.
Step 8	**end** **Example:** `Switch (config-if)# end`	Returns to privileged EXEC mode.
Step 9	**show lldp** **Example:** `Switch# show lldp`	Verifies the configuration.
Step 10	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring LLDP-MED TLVs

By default, the switch only sends LLDP packets until it receives LLDP-MED packets from the end device. It then sends LLDP packets with MED TLVs, as well. When the LLDP-MED entry has been aged out, it again only sends LLDP packets.

By using the **lldp** interface configuration command, you can configure the interface not to send the TLVs listed in the following table.

Table 52: LLDP-MED TLVs

LLDP-MED TLV	Description
inventory-management	LLDP-MED inventory management TLV

LLDP-MED TLV	Description
location	LLDP-MED location TLV
network-policy	LLDP-MED network policy TLV
power-management	LLDP-MED power management TLV

Beginning in privileged EXEC mode, follow these steps to enable a TLV on an interface:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **lldp med-tlv-select**
4. **end**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/1`	Specifies the interface on which you are configuring an LLDP-MED TLV, and enter interface configuration mode.
Step 3	**lldp med-tlv-select** **Example:** `Switch(config-if)# lldp med-tlv-select inventory management`	Specifies the TLV to enable.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Network-Policy TLV

Beginning in privileged EXEC mode, follow these steps to create a network-policy profile, configure the policy attributes, and apply it to an interface.

SUMMARY STEPS

1. **configure terminal**
2. **network-policy profile** *profile number*
3. {**voice** | **voice-signaling**} **vlan** [*vlan-id* {**cos** *cvalue* | **dscp** *dvalue*}] | [[**dot1p** {**cos** *cvalue* | **dscp** *dvalue*}] | **none** | **untagged**]
4. **exit**
5. **interface** *interface-id*
6. **network-policy** *profile number*
7. **lldp med-tlv-select network-policy**
8. **end**
9. **show network-policy profile**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**network-policy profile** *profile number* **Example:** `Switch(config)# network-policy profile 1`	Specifies the network-policy profile number, and enter network-policy configuration mode. The range is 1 to 4294967295.

	Command or Action	Purpose
Step 3	{**voice** | **voice-signaling**} **vlan** [*vlan-id* {**cos** *cvalue* | **dscp** *dvalue*}] | [[**dot1p** {**cos** *cvalue* | **dscp** *dvalue*}] | **none** | **untagged**] **Example:** `Switch(config-network-policy)# voice vlan 100 cos 4`	Configures the policy attributes: • **voice—**Specifies the voice application type. • **voice-signaling—**Specifies the voice-signaling application type. • **vlan—**Specifies the native VLAN for voice traffic. • *vlan-id***—**(Optional) Specifies the VLAN for voice traffic. The range is 1 to 4094. • **cos** *cvalue***—**(Optional) Specifies the Layer 2 priority class of service (CoS) for the configured VLAN. The range is 0 to 7; the default is 5. • **dscp** *dvalue***—**(Optional) Specifies the differentiated services code point (DSCP) value for the configured VLAN. The range is 0 to 63; the default is 46. • **dot1p—**(Optional) Configures the telephone to use IEEE 802.1p priority tagging and use VLAN 0 (the native VLAN). • **none—**(Optional) Do not instruct the IP telephone about the voice VLAN. The telephone uses the configuration from the telephone key pad. • **untagged—**(Optional) Configures the telephone to send untagged voice traffic. This is the default for the telephone. • **untagged—**(Optional) Configures the telephone to send untagged voice traffic. This is the default for the telephone.
Step 4	**exit** **Example:** `Switch(config)# exit`	Returns to global configuration mode.
Step 5	**interface** *interface-id* **Example:** `Switch (config)# interface gigabitethernet2/0/1`	Specifies the interface on which you are configuring a network-policy profile, and enter interface configuration mode.
Step 6	**network-policy** *profile number* **Example:** `Switch(config-if)# network-policy 1`	Specifies the network-policy profile number.

	Command or Action	Purpose
Step 7	**lldp med-tlv-select network-policy** **Example:** `Switch(config-if)# lldp med-tlv-select network-policy`	Specifies the network-policy TLV.
Step 8	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 9	**show network-policy profile** **Example:** `Switch# show network-policy profile`	Verifies the configuration.
Step 10	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Location TLV and Wired Location Service

Beginning in privileged EXEC mode, follow these steps to configure location information for an endpoint and to apply it to an interface.

SUMMARY STEPS

1. **configure terminal**
2. **location** {**admin-tag** *string* | **civic-location identifier** {*id* | **host**} | **elin-location** *string* **identifier** *id* | **custom-location identifier** {*id* | **host**} | **geo-location identifier** {*id* | **host**}}
3. **exit**
4. **interface** *interface-id*
5. **location** {**additional-location-information** *word* | **civic-location-id** {*id* | **host**} | **elin-location-id** *id* | **custom-location-id** {*id* | **host**} | **geo-location-id** {*id* | **host**} }
6. **end**
7. Use one of the following:
   - **show location admin-tag** *string*
   - **show location civic-location identifier** *id*
   - **show location elin-location identifier** *id*
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

Step 2	**location** {**admin-tag** *string* | **civic-location identifier** {*id* | **host**} | **elin-location** *string* **identifier** *id* | **custom-location identifier** {*id* | **host**} | **geo-location identifier** {*id* | **host**}}
	
	Example:
	`Switch(config)# location civic-location identifier 1`
	`Switch(config-civic)# number 3550`
	`Switch(config-civic)# primary-road-name "Cisco Way"`
	`Switch(config-civic)# city "San Jose"`
	`Switch(config-civic)# state CA`
	`Switch(config-civic)# building 19`
	`Switch(config-civic)# room C6`
	`Switch(config-civic)# county "Santa Clara"`
	`Switch(config-civic)# country US`
		Specifies the location information for an endpoint.
		
		• **admin-tag—**Specifies an administrative tag or site information.
		• **civic-location—**Specifies civic location information.
		• **elin-location—**Specifies emergency location information (ELIN).
		• **custom-location—**Specifies custom location information.
		• **geo-location—**Specifies geo-spatial location information.
		• **identifier** *id***—**Specifies the ID for the civic, ELIN, custom, or geo location.
		• **host—**Specifies the host civic, custom, or geo location.
		• *string***—**Specifies the site or location information in alphanumeric format.

	Command or Action	Purpose
Step 3	**exit** **Example:** `Switch(config-civic)# exit`	Returns to global configuration mode.
Step 4	**interface** *interface-id* **Example:** `Switch (config)# interface gigabitethernet2/0/1`	Specifies the interface on which you are configuring the location information, and enter interface configuration mode.
Step 5	**location** {**additional-location-information** *word* | **civic-location-id** {*id* | **host**} | **elin-location-id** *id* | **custom-location-id** {*id* | **host**} | **geo-location-id** {*id* | **host**} } **Example:** `Switch(config-if)# location elin-location-id 1`	Enters location information for an interface: • **additional-location-information**—Specifies additional information for a location or place. • **civic-location-id**—Specifies global civic location information for an interface. • **elin-location-id**—Specifies emergency location information for an interface. • **custom-location-id**—Specifies custom location information for an interface. • **geo-location-id**—Specifies geo-spatial location information for an interface. • **host**—Specifies the host location identifier. • *word*—Specifies a word or phrase with additional location information. • *id*—Specifies the ID for the civic, ELIN, custom, or geo location. The ID range is 1 to 4095.
Step 6	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 7	Use one of the following: • **show location admin-tag** *string* • **show location civic-location identifier** *id* • **show location elin-location identifier** *id*	Verifies the configuration.

	Command or Action	Purpose

	Example:
	`Switch# show location admin-tag`
	or
	`Switch# show location civic-location identifier`
	or
	`Switch# show location elin-location identifier`	

Step 8	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Enabling Wired Location Service on the Switch

Beginning in privileged EXEC mode, follow these steps to enable wired location service on the switch.

SUMMARY STEPS

1. **configure terminal**
2. **nmsp notification interval** {**attachment** | **location**} *interval-seconds*
3. **end**
4. **show network-policy profile**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**nmsp notification interval** {**attachment** | **location**} *interval-seconds*	Specifies the NMSP notification interval.

	Command or Action	Purpose
	Example: `Switch(config)# nmsp notification interval location 10`	**attachment**—Specifies the attachment notification interval. **location**—Specifies the location notification interval. *interval-seconds*—Duration in seconds before the switch sends the MSE the location or attachment updates. The range is 1 to 30; the default is 30.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show network-policy profile** **Example:** `Switch# show network-policy profile`	Verifies the configuration.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuration Examples for LLDP, LLDP-MED, and Wired Location Service

Configuring Network-Policy TLV: Examples

This example shows how to configure VLAN 100 for voice application with CoS and to enable the network-policy profile and network-policy TLV on an interface:

```
Switch# configure terminal
Switch(config)# network-policy 1
Switch(config-network-policy)# voice vlan 100 cos 4
Switch(config-network-policy)# exit
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# network-policy profile 1
Switch(config-if)# lldp med-tlv-select network-policy
```

This example shows how to configure the voice application type for the native VLAN with priority tagging:

```
Switchconfig-network-policy)# voice vlan dot1p cos 4
Switchconfig-network-policy)# voice vlan dot1p dscp 34
```

Monitoring and Maintaining LLDP, LLDP-MED, and Wired Location Service

Commands for monitoring and maintaining LLDP, LLDP-MED, and wired location service.

Command	Description
clear lldp counters	Resets the traffic counters to zero.
clear lldp table	Deletes the LLDP neighbor information table.
clear nmsp statistics	Clears the NMSP statistic counters.
show lldp	Displays global information, such as frequency of transmissions, the holdtime for packets being sent, and the delay time before LLDP initializes on an interface.
show lldp entry *entry-name*	Displays information about a specific neighbor. You can enter an asterisk (*) to display all neighbors, or you can enter the neighbor name.
show lldp interface [*interface-id*]	Displays information about interfaces with LLDP enabled. You can limit the display to a specific interface.
show lldp neighbors [*interface-id*] [**detail**]	Displays information about neighbors, including device type, interface type and number, holdtime settings, capabilities, and port ID. You can limit the display to neighbors of a specific interface or expand the display for more detailed information.
show lldp traffic	Displays LLDP counters, including the number of packets sent and received, number of packets discarded, and number of unrecognized TLVs.
show location admin-tag *string*	Displays the location information for the specified administrative tag or site.
show location civic-location identifier *id*	Displays the location information for a specific global civic location.
show location elin-location identifier *id*	Displays the location information for an emergency location
show network-policy profile	Displays the configured network-policy profiles.
show nmsp	Displays the NMSP information

Additional References for LLDP, LLDP-MED, and Wired Location Service

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for LLDP, LLDP-MED, and Wired Location Service

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 23

Configuring System MTU

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information about the MTU

The default maximum transmission unit (MTU) size for frames received and sent on all switch interfaces is 1500 bytes. You can change the MTU size to support switched jumbo frames on all Gigabit Ethernet and 10-Gigabit Ethernet interfaces and to support routed frames on all routed ports.

System MTU Values

The following MTU values can be configured:

- System MTU--This value applies to switched packets on the Gigabit Ethernet and 10-Gigabit Ethernet ports of the switch. Use the **system mtu** *bytes* global configuration command to specify the system jumbo MTU value.
- Protocol-specific MTU--This value applies only to routed packets on all routed ports of the switch or switch stack. Use the **ip mtu** *bytes* or **ipv6 mtu** *bytes* interface configuration command to specify the protocol-specific MTU value.

System MTU Guidelines

When configuring the system MTU values, follow these guidelines:

- The switch does not support the MTU on a per-interface basis.
- If you enter the **system mtu** *bytes* global configuration command, the command does not take effect on the switch. This command only affects the system MTU size on Fast Ethernet switch ports.

System MTU Value Application

This table shows how the MTU values are applied.

Table 53: MTU Values

Configuration	system mtu command	ip mtu command	ipv6 mtu command
Standalone switch or switch stack	You can enter the **system mtu** command on a switch or switch stack, but system MTU value does not take effect. The range is from 1500 to 9198 bytes.	Use the **ip mtu** *bytes* command. The range is from 68 up to the system MTU value (in bytes). **Note** The IP MTU value is the applied value, not the configured value.	Use the **ipv6 mtu** *bytes* command. The range is from 1280 to the system jumbo MTU value (in bytes). **Note** The IPv6 MTU value is the applied value, not the configured value.

The upper limit of the IP or IPv6 MTU value is based on the switch or switch stack configuration and refers to the currently applied system MTU value. For more information about setting the MTU sizes, see the **system mtu** global configuration command in the command reference for this release.

How to Configure MTU Sizes

Configuring the System MTU

Beginning in privileged EXEC mode, follow these steps to change the MTU size for switched packets:

SUMMARY STEPS

1. **configure terminal**
2. **system mtu** *bytes*
3. **end**
4. **copy running-config startup-config**
5. **reload**
6. **show system mtu**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**system mtu** *bytes* **Example:** `Switch(config)# system mtu 7500`	(Optional) Changes the MTU size for all Gigabit Ethernet and 10-Gigabit Ethernet interfaces on the switch or the switch stack.
Step 3	**end** **Example:** `Switch(config)#  end`	Returns to privileged EXEC mode.
Step 4	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	Saves your entries in the configuration file.
Step 5	**reload** **Example:** `Switch#  reload`	Reloads the operating system.
Step 6	**show system mtu** **Example:** `Switch# show system mtu`	Verifies your settings.

Configuring Protocol-Specific MTU

Beginning in privileged EXEC mode, follow these steps to change the MTU size for routed ports:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface*
3. **ip mtu** *bytes*
4. **ipv6 mtu** *bytes*
5. **end**
6. **copy running-config startup-config**
7. **reload**
8. **show system mtu**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface* **Example:** `Switch(config)# interface gigabitethernet0/0`	Enters interface configuration mode.
Step 3	**ip mtu** *bytes* **Example:** `Switch(config-if)# ip mtu 68`	(Optional) Changes the IPv4 MTU size
Step 4	**ipv6 mtu** *bytes* **Example:** `Switch(config-if)# ipv6 mtu 1280`	(Optional) Changes the IPv6 MTU size.
Step 5	**end** **Example:** `Switch(config-if)#  end`	Returns to privileged EXEC mode.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	Saves your entries in the configuration file.
Step 7	**reload** **Example:** `Switch#  reload`	Reloads the operating system.

	Command or Action	Purpose
Step 8	**show system mtu** **Example:** `Switch# show system mtu`	Verifies your settings.

Configuration Examples for System MTU

This example shows how to set the maximum packet size for a Gigabit Ethernet port to 7500 bytes:

```
Switch(config)# system mtu 7500
Switch(config)#
Switch(config)# exit
```

If you enter a value that is outside the allowed range for the specific type of interface, the value is not accepted. This example shows the response when you try to set Gigabit Ethernet interfaces to an out-of-range number:

```
Switch(config)# system mtu 25000
                                   ^
% Invalid input detected at '^' marker.
```

This is an example of output from the **show system mtu** command:

```
Switch# show system mtu
Global Ethernet MTU is 1500 bytes.
```

Additional References for System MTU

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for System MTU

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 24

Configuring Internal Power Supplies

Information About Internal Power Supplies

See the switch installation guide for information about the power supplies.

How to Configure Internal Power Supplies

Configuring an Internal Power Supply

You can use the **power supply** EXEC command to configure and manage the internal power supply on the switch. The switch does not support the **no power supply** EXEC command.

SUMMARY STEPS

1. **power supply** *switch_number* **slot**{**A** | **B**} { **off** | **on** }
2. **show environment power**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**power supply** *switch_number* **slot**{**A** | **B**} { **off** | **on** } **Example:** `Switch# power supply 1 slot A on`	Sets the specified power supply to **off** or **on** by using one of these keywords: • **A** —Selects the power supply in slot A. • **B** —Selects power supply in slot B. **Note** Power supply slot B is the closest to the outer edge of the switch. • **off** —Set the power supply off. • **on** —Set the power supply on. By default, the switch power supply is **on**.
Step 2	**show environment power** **Example:** `Switch# show environment power`	Verifies your settings.

Monitoring Internal Power Supplies

Table 54: Show Commands for Power Supplies

Command	Purpose
show environment power [**all** | **switch** *switch_number*]	(Optional) Displays the status of the internal power supplies for each switch in the stack or for the specified switch. The range is 1 to 9, depending on the switch member numbers in the stack.

Configuration Examples for Internal Power Supplies

This example shows how to set the power supply in slot A to off:

```
Switch# power supply 1 slot A off
Disabling Power supply A may result in a power loss to PoE devices and/or switches ...
Continue? (yes/[no]): yes
Switch#
Jun 10 04:52:54.389: %PLATFORM_ENV-6-FRU_PS_OIR: FRU Power Supply 1 powered off
Jun 10 04:52:56.717: %PLATFORM_ENV-1-FAN_NOT_PRESENT: Fan is not present
Switch#
```

This example shows how to set the power supply in slot A to on:

```
Switch# power supply 1 slot A on
Jun 10 04:54:39.600: %PLATFORM_ENV-6-FRU_PS_OIR: FRU Power Supply 1 powered on
```

This example shows the output of the **show env power** command:

```
Switch# show env power

SW  PID                 Serial#     Status           Sys Pwr  PoE Pwr  Watts
--- ------------------  ----------  ---------------  -------  -------  -----
1A  PWR-C1-715WAC       LIT161010UE  OK              Good     Good     715
1B  Not Present

Switch#
```

Table 55: show env power Status Descriptions

Field	Description
OK	The power supply is present and power is good.
Not Present	No power supply is installed.
No Input Power	The power supply is present but there is no input power.
Disabled	The power supply and input power are present, but power supply is switched off by CLI.
Not Responding	The power supply is not recognizable or is faulty.
Failure-Fan	The power supply fan is faulty.

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Internal Power Supplies

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.
Cisco IOS XE 3.3SE	The **slot** keyword replaced the **frufep** keyword.

CHAPTER 25

Configuring Stack Power

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for StackPower

StackPower uses these terms:

- *Available power* is the total power available for PoE from all power supplies in the power stack. To see the available power in a stack, enter the **show power inline** privileged EXEC command.
- *Budgeted power* is the power allocated to all powered devices connected to PoE ports in the stack. Budgeted power is referred to as *Used (Watts)* in the output of the **show power inline** command.

- *Consumed power* is the actual power consumed by the powered devices. Consumed power is typically less that the budgeted power. To see the consumed power in a stack, enter the **show power inline police** privileged EXEC command.

Information About StackPower

Power Supplies for the Stacking Switch

The switch has two power supplies per system, allowing the power load to be split between them. This accommodates the increased maximum power of 30 watts per port provided to a powered device to meet the PoE+ standard (802.3at). With PoE+, a 48-port system would need 1440 Watts to provide 30 Watts per powered device for the PoE ports. Systems with fewer powered devices might require only one power supply. In this case, the additional power supply can provide one-to-one redundancy for the active supply.

In addition, the stacking switch supports StackPower, which allows the power supplies to share the load across multiple systems in a stack. By connecting the switches with power stack cables, you can manage the power supplies of up to four stack members as a one large power supply that provides power to all switches and to the powered devices connected to switch ports. Since power supplies are most effective when running at 30 to 90% of their maximum load, taking some of the power supplies offline provides maximum power efficiency. Switches in a power stack must be members of the same switch (data) stack.

StackPower Modes

You can configure the power stack to run in one of two modes:

- In power-sharing mode (the default), all input power is available to be used for power loads. The total available power in all switches in the power stack (up to four) is treated as a single large power supply, with power available to all switches and to all powered devices connected to PoE ports. In this mode, the total available power is used for power budgeting decisions and no power is reserved to accommodate power-supply failures. If a power supply fails, powered devices and switches could be shut down (load shedding).
- In redundant mode, the power from the largest power supply in the system is subtracted from the power budget, which reduces the total available power, but provides backup power in case of a power-supply failure. Although there is less available power in the pool for switches and powered devices to draw from, the possibility of having to shut down switches or powered devices in case of a power failure or extreme power load is reduced.

In addition, you can configure the mode to run a strict power budget or a non-strict (relaxed) power budget. In both modes, power is denied when there is no more power available in the power budget.

- In strict mode, when a power supply fails and the available power drops below the budgeted power, the system balances the budget through load shedding of powered devices, even if the actual power being consumed is less than the available power.
- In non-strict mode, the power stack is allowed to run in an over-allocated state and is stable as long as the actual power does not exceed the available power. In this mode, a powered device drawing more than normal power could cause the power stack to start shedding loads. This is normally not a problem because most devices do not run at full power and the chances of multiple powered devices in the stack requiring maximum power at the same time is small.

You configure power modes at a power-stack level (that is, the mode is the same for all switches in the power stack).

You can also configure a switch connected in a power stack to not participate in the power stack by setting the switch to standalone power mode. This mode shuts down both stack power ports.

Related Topics

Configuring PowerStack Parameters, on page 458
Configuring PowerStack Parameters: Example, on page 464

Power Priority

You can configure the priority of a switch or powered device to receive power. This priority determines the order in which devices are shut down in case of a power shortage. You can configure three priorities per system: the system (or switch) priority, the priority of the high-priority PoE ports on a switch, and the priority of the low-priority PoE ports on a switch.

You set port priority at the interface level for powered devices connected to a PoE port. By default, all ports are low priority.

You configure the priority values of each switch in the power stack and of all high and low priority ports on that switch to set the order in which switches and ports are shut down when power is lost and load shedding must occur. Priority values are from 1 to 27; switches and ports with highest values are shut down first.

Note The 27 priorities are used to accommodate power stacks connected in a star configuration with the expandable power supply. In this configuration, there would be nine members (switches) per system with three priorities per switch.

On any switch, the switch priority must be lower than port priorities. and the high priority value must be set lower than the low priority value. We recommend that you configure different priority values for each switch and for its high priority ports and low priority ports. This limits the number of devices shut down at one time during a loss of power. If you try to configure the same priority value on different switches in a power stack, the configuration is allowed, but you receive a warning message.

The default priority ranges, if none are configured, are 1-9 for switches, 10-18 for high-priority ports, and 19-27 for low-priority ports.

Related Topics

Configuring PowerStack Switch Power Parameters, on page 460
Configuring PowerStack Switch Power Parameters: Example, on page 464
Configuring PoE Port Priority, on page 461
Configuring PoE Port Priority: Example, on page 465

Load Shedding

Load shedding is the process of shutting down devices in case of power supply, cable, or system failures. For power stacks in power-sharing mode, there are two types of load-shedding: immediate and graceful.

- Immediate load shed occurs when a failure could cause the power stack to fail very quickly. For example, if the largest power supply in the power stack fails, this could cause the stack to immediately start shutting down powered devices.
- Graceful load-shedding can occur when a smaller power supply fails. Switches and powered devices are shut down in order of their configured priority, starting with devices with priority 27, until the power budget matches the input power.

Graceful load shedding is always enabled and immediate load shedding occurs only when necessary, so both can occur at the same time.

Note Load shedding does not occur in redundant mode unless two or more power supplies fail, because the largest power supply is used as a backup power source.

Notes on load shedding:

- The method (immediate or graceful) is not user-configurable, but is based on the power budget.
- Immediate load shedding also occurs in the order of configured priority, but occurs very quickly to prevent hardware damage caused by loss of power.
- If a switch is shut down because of load shedding, the output of the **show stack-power** privileged EXEC command still includes the MAC address of the shut down switch as a neighbor switch, even though the switch is down. This command output shows the StackPower topology, even if there is not enough power to power up a switch.

Related Topics

Immediate Load Shedding: Examples, on page 462

How to Configure StackPower

Configuring stack power includes these tasks:

Configuring PowerStack Parameters

SUMMARY STEPS

1. **configure terminal**
2. **stack-power stack** *power stack name*
3. **mode** {**power-sharing** | **redundant**} [**strict**]
4. **end**
5. **show stack-power**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**stack-power stack** *power stack name* **Example:** `Switch(config)# stack-power stack power 1`	Enters the stack power stack name and enter power stack configuration mode. The name can be up to 31 characters.
Step 3	**mode** {**power-sharing** | **redundant**} [**strict**] **Example:** `Switch(config-stackpower)# mode redundant`	Sets the operating mode for the power stack: • **power-sharing**—The input power from all switches in the power stack can be used for loads, and the total available power appears as one huge power supply. This is the default. • **redundant**—The largest power supply is removed from the power pool to be used as backup power in case one of the other power supplies fails. This is the recommended mode if enough power is available in the system. • **strict**—(Optional) Configures the power stack mode to run a strict power budget. The stack power needs cannot exceed the available power. The default is non-strict.
Step 4	**end** **Example:** `Switch(config-stackpower)# end`	Returns to privileged EXEC mode.
Step 5	**show stack-power** **Example:** `Switch# show stack-power`	Verifies your entries.

Related Topics

StackPower Modes, on page 456
Configuring PowerStack Parameters: Example, on page 464

Configuring PowerStack Switch Power Parameters

SUMMARY STEPS

1. **configure terminal**
2. **stack-power switch** *switch-number*
3. **stack** [*power-stack-name*]
4. **power-priority switch** *value*
5. **power-priority high** *value*
6. **power-priority low** *value*
7. **end**
8. **show stack-power**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	
Step 2	**stack-power switch** *switch-number* **Example:** `Switch(config)# stack-power switch 4`	Enters global configuration mode. Enters switch stack power configuration mode for the stack member switch number. The range is from 1 to 9. **Note** Only four switches can belong to the same power stack.
Step 3	**stack** [*power-stack-name*] **Example:** `Switch(config-switch-stackpower)# stack power2`	Enters the name of the power stack to which the switch belongs. The name can be up to 31 characters. If you do not enter a name and no other switches in the power stack have a name configured, a power-stack name is automatically generated.
Step 4	**power-priority switch** *value* **Example:** `Switch(config-switch-stackpower)# power-priority switch 5`	Sets the power priority of the switch. The range is from 1 to 27. This value must be lower than the value set for the low and high-priority ports.
Step 5	**power-priority high** *value* **Example:** `Switch(config-switch-stackpower)# power-priority high 12`	Sets the power priority of the PoE ports on the switch that are configured as high-priority ports. The range is from 1 to 27, with 1 as the highest priority. The **high** value must be lower than the value set for the low-priority ports and higher than the value set for the switch.

	Command or Action	Purpose
Step 6	**power-priority low** *value* **Example:** `Switch(config-switch-stackpower)#` **`power-priority low 20`**	Sets the power priority of the PoE ports on the switch that are configured as low-priority ports. The range is from 1 to 27. This value must be higher than the value set for the high-priority ports and the value set for the switch.
Step 7	**end** **Example:** `Switch(config-switch-stackpower)#` **`end`**	Returns to privileged EXEC mode.
Step 8	**show stack-power** **Example:** `Switch#` **`show stack-power`**	Verifies your entries.

Related Topics

Power Priority, on page 457
Configuring PowerStack Switch Power Parameters: Example, on page 464

Configuring PoE Port Priority

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **power inline port priority** {**high** | **low**}
4. **end**
5. **show power inline priority**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch#` **`configure terminal`**	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters the interface ID of the port in the stack and enters interface configuration mode. The interface must be a PoE port.
Step 3	**power inline port priority** {**high** | **low**} **Example:** `Switch(config-if)# power inline port priority high`	Sets the power priority of the port to **high** or **low**. Powered devices connected to low priority ports are shut down first in case of a power reduction. The default is low priority. **Note** Although the **power inline port priority** {**high** | **low**} command is visible on the nonstacking switch PoE ports, it has no effect because these switches do not participate in stack power.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show power inline priority** **Example:** `Switch# show power inline priority`	Verifies your entries.

Related Topics

Power Priority, on page 457
Configuring PoE Port Priority: Example, on page 465

Configuration Examples for Stack Power

Immediate Load Shedding: Examples

For power stacks in power-sharing mode, if a large power supply in the power stack fails, the stack immediately starts shutting down powered devices until the power budget matches the input power. This example has a power stack of four switches (*Powerstack1*) in power sharing mode and shows which devices would be shut down in the immediate load shedding process caused by loss of either of two power supplies.

The output of the **show env all** command shows that power supplies included in power sharing are a 715 W power supply in switch 1, and one 350 W and one 1100 W power supply in switch 4. Other power supplies are inactive (disabled or not present).

```
Switch# show env all
Switch 1 FAN 1 is OK
Switch 1 FAN 2 is OK
```

```
Switch 1 FAN 3 is OK
FAN PS-1 is OK
FAN PS-2 is OK
Switch 1: SYSTEM TEMPERATURE is OK
SW  PID                 Serial#     Status           Sys Pwr  PoE Pwr  Watts
--  ------------------  ----------  ---------------  -------  -------  -----
1A  PWR-C1-715WAC       LIT133705FH  OK              Good     Good     715
1B  PWR-C1-715WAC       DTN1341K018  Disabled        Good     Good     715
2A  Not Present
2B  PWR-C1-350WAC       LIT13330FNM  Disabled        Good     Good     350
3A  PWR-C1-350WAC       LIT13330FN3  Disabled        Good     Good     350
3B  Not Present
4A  PWR-C1-350WAC       DTN1342L00T  OK              Good     Good     350
4B  PWR-C1-1100WAC      LIT13370577  OK              Good     Good     1100
```

The output of the **show stack-power** privileged EXEC command shows the priorities of the powered devices and switches in the power stack.

```
Switch# show stack-power
Power stack name: Powerstack1
    Stack mode: Power sharing
    Switch 1:
        Power budget: 206
        Low port priority value: 17
        High port priority value: 16
        Switch priority value: 2
        Port A status: Not shut
        Port B status: Not shut
        Neighbor on port A: 0022.bdcf.ab00
        Neighbor on port B: 0022.bdd0.4380

    Switch 2:
        Power budget: 206
        Low port priority value: 12
        High port priority value: 11
        Switch priority value: 1
        Port A status: Not shut
        Port B status: Not shut
        Neighbor on port A: 0022.bdd0.6d00
        Neighbor on port B: 0022.bdcf.af80

    Switch 3:
        Power budget: 656
        Low port priority value: 22
        High port priority value: 21
        Switch priority value: 3
        Port A status: Not shut
        Port B status: Not shut
        Neighbor on port A: 0022.bdcf.af80
        Neighbor on port B: 0022.bdd0.6d00

    Switch 4:
        Power budget: 682
        Low port priority value: 27
        High port priority value: 26
        Switch priority value: 4
        Port A status: Not shut
        Port B status: Not shut
        Neighbor on port A: 0022.bdd0.4380
        Neighbor on port B: 0022.bdcf.ab00
```

If the 715 W or 1100 W power supply fails, devices (powered devices connected to PoE ports and the switches themselves) would be shut down in the this order until power consumption drops below 105% of the rated power of the remaining power supplies:

- Devices connected to Switch 4 low priority ports (priority 27)
- Devices connected to Switch 4 high priority ports (priority 26)

- Devices connected to Switch 3 low priority ports (priority 22)
- Devices connected to Switch 3 high priority ports (priority 21)
- Devices connected to Switch 1 low priority ports (priority 17)
- Devices connected to Switch 1 high priority ports (priority 16)
- Devices connected to Switch 2 low priority ports (priority 12)
- Devices connected to Switch 2 high priority ports (priority 11)
- Switch 4 (priority 4)
- Switch 3 (priority 3)
- Switch 1 (priority 2)

Switch 2 would never have to be shut down because all power would have been lost by the time priority 1 devices were reached.

Related Topics

Load Shedding, on page 457

Configuring PowerStack Parameters: Example

This is an example of setting the stack power mode for the stack named *power1* to redundant power mode. The largest power supply in the stack is removed from the power budget and used as a backup in case of power supply failure.

```
Switch(config)# stack-power stack power1
Switch(config-stackpower)# mode redundant
Switch(config-stackpower)# exit
```

Related Topics

StackPower Modes, on page 456
Configuring PowerStack Parameters, on page 458

Configuring PowerStack Switch Power Parameters: Example

This is an example of setting the switch stack power parameters for switch 3 in the stack that is connected to the power stack with the stack ID *power2*. If load-shedding becomes necessary, switches and powered devices in the power stack with the higher numbers are shut down first, with shutdown proceeding in order.

```
Switch(config)# stack-power switch 3
Switch(config-switch-stackpower)# stack power2
Switch(config-switch-stackpower)# power-priority switch 5
Switch(config-switch-stackpower)# power-priority high 12
Switch(config-switch-stackpower)# power-priority low 20
Switch(config-switch-stackpower)# exit
Switch(config-stackpower)# exit
```

Note Entering the **write erase** and **reload** privileged EXEC commands do not change the power priority or power mode non-default configuration saved in the switch flash memory.

Related Topics

Power Priority, on page 457
Configuring PowerStack Switch Power Parameters, on page 460

Configuring PoE Port Priority: Example

This is an example of setting the power priority of a port to high so that it is one of the last ports to shut down in case of a power failure.

```
Switch(config)# interface gigabitetherent1/0/1
Switch(config-if)# power inline port priority high
Switch(config-if)# exit
```

Related Topics

Power Priority, on page 457
Configuring PoE Port Priority, on page 461

Where to Go Next

See the hardware installation guide for information

- Designing and connecting the power stack.
- StackPower star and ring configuration

Additional References for StackPower

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for StackPower

Table 56: Feature Information for StackPower

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 26

Configuring PoE

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information about PoE

Power over Ethernet Ports

A PoE-capable switch port automatically supplies power to one of these connected devices if the switch senses that there is no power on the circuit:

- a Cisco pre-standard powered device (such as a Cisco IP Phone or a Cisco Aironet Access Point)
- an IEEE 802.3af-compliant powered device
- an IEEE 802.3at-compliant powered device

A powered device can receive redundant power when it is connected to a PoE switch port and to an AC power source. The device does not receive redundant power when it is only connected to the PoE port.

After the switch detects a powered device, the switch determines the device power requirements and then grants or denies power to the device. The switch can also sense the real-time power consumption of the device by monitoring and policing the power usage.

Supported Protocols and Standards

The switch uses these protocols and standards to support PoE:

- CDP with power consumption—The powered device notifies the switch of the amount of power it is consuming. The switch does not reply to the power-consumption messages. The switch can only supply power to or remove power from the PoE port.
- Cisco intelligent power management—The powered device and the switch negotiate through power-negotiation CDP messages for an agreed-upon power-consumption level. The negotiation allows a high-power Cisco powered device, which consumes more than 7 W, to operate at its highest power mode. The powered device first boots up in low-power mode, consumes less than 7 W, and negotiates to obtain enough power to operate in high-power mode. The device changes to high-power mode only when it receives confirmation from the switch.

  High-power devices can operate in low-power mode on switches that do not support power-negotiation CDP.

  Cisco intelligent power management is backward-compatible with CDP with power consumption; the switch responds according to the CDP message that it receives. CDP is not supported on third-party powered devices; therefore, the switch uses the IEEE classification to determine the power usage of the device.
- IEEE 802.3af—The major features of this standard are powered-device discovery, power administration, disconnect detection, and optional powered-device power classification. For more information, see the standard.
- IEEE 802.3at—The PoE+ standard increases the maximum power that can be drawn by a powered device from 15.4 W per port to 30 W per port.

Related Topics

Cisco Universal Power Over Ethernet

Powered-Device Detection and Initial Power Allocation

The switch detects a Cisco pre-standard or an IEEE-compliant powered device when the PoE-capable port is in the no-shutdown state, PoE is enabled (the default), and the connected device is not being powered by an AC adaptor.

After device detection, the switch determines the device power requirements based on its type:

- A Cisco prestandard powered device does not provide its power requirement when the switch detects it, so the switch allocates 15.4 W as the initial allocation for power budgeting.

  The initial power allocation is the maximum amount of power that a powered device requires. The switch initially allocates this amount of power when it detects and powers the powered device. As the switch receives CDP messages from the powered device and as the powered device negotiates power levels with the switch through CDP power-negotiation messages, the initial power allocation might be adjusted.

- The switch classifies the detected IEEE device within a power consumption class. Based on the available power in the power budget, the switch determines if a port can be powered. Table 57: IEEE Power Classifications, on page 469 lists these levels.

Table 57: IEEE Power Classifications

Class	Maximum Power Level Required from the Switch
0 (class status unknown)	15.4 W
1	4 W
2	7 W
3	15.4 W
4	30 W (For IEEE 802.3at Type 2 powered devices)

The switch monitors and tracks requests for power and grants power only when it is available. The switch tracks its power budget (the amount of power available on the switch for PoE). The switch performs power-accounting calculations when a port is granted or denied power to keep the power budget up to date.

After power is applied to the port, the switch uses CDP to determine the *CDP-specific* power consumption requirement of the connected Cisco powered devices, which is the amount of power to allocate based on the CDP messages. The switch adjusts the power budget accordingly. This does not apply to third-party PoE devices. The switch processes a request and either grants or denies power. If the request is granted, the switch updates the power budget. If the request is denied, the switch ensures that power to the port is turned off, generates a syslog message, and updates the LEDs. Powered devices can also negotiate with the switch for more power.

With PoE+, powered devices use IEEE 802.3at and LLDP power with media dependent interface (MDI) type, length, and value descriptions (TLVs), Power-via-MDI TLVs, for negotiating power up to 30 W. Cisco pre-standard devices and Cisco IEEE powered devices can use CDP or the IEEE 802.3at power-via-MDI power negotiation mechanism to request power levels up to 30 W.

Note The initial allocation for Class 0, Class 3, and Class 4 powered devices is 15.4 W. When a device starts up and uses CDP or LLDP to send a request for more than 15.4 W, it can be allocated up to the maximum of 30 W.

Note The CDP-specific power consumption requirement is referred to as the *actual* power consumption requirement in the software configuration guides and command references.

If the switch detects a fault caused by an undervoltage, overvoltage, overtemperature, oscillator-fault, or short-circuit condition, it turns off power to the port, generates a syslog message, and updates the power budget and LEDs.

The PoE feature operates the same whether or not the switch is a stack member. The power budget is per-switch and independent of any other switch in the stack. Election of a new active switch does not affect PoE operation.

The active switch keeps track of the PoE status for all switches and ports in the stack and includes the status in output displays.

The stacking-capable switch also supports StackPower, which allows the power supplies to share the load across multiple systems in a stack when you connect the switches with power stack cables. You can manage the power supplies of up to four stack members as a one large power supply.

Power Management Modes

The switch supports these PoE modes:

- **auto—**The switch automatically detects if the connected device requires power. If the switch discovers a powered device connected to the port and if the switch has enough power, it grants power, updates the power budget, turns on power to the port on a first-come, first-served basis, and updates the LEDs. For LED information, see the hardware installation guide.

  If the switch has enough power for all the powered devices, they all come up. If enough power is available for all powered devices connected to the switch, power is turned on to all devices. If there is not enough available PoE, or if a device is disconnected and reconnected while other devices are waiting for power, it cannot be determined which devices are granted or are denied power.

  If granting power would exceed the system power budget, the switch denies power, ensures that power to the port is turned off, generates a syslog message, and updates the LEDs. After power has been denied, the switch periodically rechecks the power budget and continues to attempt to grant the request for power.

  If a device being powered by the switch is then connected to wall power, the switch might continue to power the device. The switch might continue to report that it is still powering the device whether the device is being powered by the switch or receiving power from an AC power source.

  If a powered device is removed, the switch automatically detects the disconnect and removes power from the port. You can connect a nonpowered device without damaging it.

  You can specify the maximum wattage that is allowed on the port. If the IEEE class maximum wattage of the powered device is greater than the configured maximum value, the switch does not provide power to the port. If the switch powers a powered device, but the powered device later requests through CDP messages more than the configured maximum value, the switch removes power to the port. The power that was allocated to the powered device is reclaimed into the global power budget. If you do not specify a wattage, the switch delivers the maximum value. Use the **auto** setting on any PoE port. The auto mode is the default setting.

- **static—**The switch pre-allocates power to the port (even when no powered device is connected) and guarantees that power will be available for the port. The switch allocates the port configured maximum wattage, and the amount is never adjusted through the IEEE class or by CDP messages from the powered device. Because power is pre-allocated, any powered device that uses less than or equal to the maximum wattage is guaranteed to be powered when it is connected to the static port. The port no longer participates in the first-come, first-served model.

  However, if the powered-device IEEE class is greater than the maximum wattage, the switch does not supply power to it. If the switch learns through CDP messages that the powered device is consuming more than the maximum wattage, the switch shuts down the powered device.

  If you do not specify a wattage, the switch pre-allocates the maximum value. The switch powers the port only if it discovers a powered device. Use the **static** setting on a high-priority interface.

- **never—**The switch disables powered-device detection and never powers the PoE port even if an unpowered device is connected. Use this mode only when you want to make sure that power is never applied to a PoE-capable port, making the port a data-only port.

For most situations, the default configuration (auto mode) works well, providing plug-and-play operation. No further configuration is required. However, perform this task to configure a PoE port for a higher priority, to make it data only, or to specify a maximum wattage to disallow high-power powered devices on a port.

Stacking-capable switches also support StackPower, which allows switch power supplies to share the load across multiple systems in a stack by connecting up to four switches with power stack cables.

When you make PoE configuration changes, the port being configured drops power. Depending on the new configuration, the state of the other PoE ports, and the state of the power budget, the port might not be powered up again. For example, port 1 is in the auto and on state, and you configure it for static mode. The switch removes power from port 1, detects the powered device, and repowers the port. If port 1 is in the auto and on state and you configure it with a maximum wattage of 10 W, the switch removes power from the port and then redetects the powered device. The switch repowers the port only if the powered device is a class 1, class 2, or a Cisco-only powered device.

Power Monitoring and Power Policing

When policing of the real-time power consumption is enabled, the switch takes action when a powered device consumes more power than the maximum amount allocated, also referred to as the *cutoff-power value*.

When PoE is enabled, the switch senses the real-time power consumption of the powered device. The switch monitors the real-time power consumption of the connected powered device; this is called *power monitoring* or *power sensing*. The switch also polices the power usage with the *power policing* feature.

Power monitoring is backward-compatible with Cisco intelligent power management and CDP-based power consumption. It works with these features to ensure that the PoE port can supply power to the powered device.

The switch senses the real-time power consumption of the connected device as follows:

1. The switch monitors the real-time power consumption on individual ports.
2. The switch records the power consumption, including peak power usage. The switch reports the information through the CISCO-POWER-ETHERNET-EXT-MIB.
3. If power policing is enabled, the switch polices power usage by comparing the real-time power consumption to the maximum power allocated to the device. The maximum power consumption is also referred to as the *cutoff power* on a PoE port.

   If the device uses more than the maximum power allocation on the port, the switch can either turn off power to the port, or the switch can generate a syslog message and update the LEDs (the port LED is now blinking amber) while still providing power to the device based on the switch configuration. By default, power-usage policing is disabled on all PoE ports.

   If error recovery from the PoE error-disabled state is enabled, the switch automatically takes the PoE port out of the error-disabled state after the specified amount of time.

   If error recovery is disabled, you can manually re-enable the PoE port by using the **shutdown** and **no shutdown** interface configuration commands.
4. If policing is disabled, no action occurs when the powered device consumes more than the maximum power allocation on the PoE port, which could adversely affect the switch.

Power Consumption Values

You can configure the initial power allocation and the maximum power allocation on a port. However, these values are only the configured values that determine when the switch should turn on or turn off power on the

PoE port. The maximum power allocation is not the same as the actual power consumption of the powered device. The actual cutoff power value that the switch uses for power policing is not equal to the configured power value.

When power policing is enabled, the switch polices the power usage *at the switch port*, which is greater than the power consumption of the device. When you are manually set the maximum power allocation, you must consider the power loss over the cable from the switch port to the powered device. The cutoff power is the sum of the rated power consumption of the powered device and the worst-case power loss over the cable.

We recommend that you enable power policing when PoE is enabled on your switch. For example, if policing is disabled and you set the cutoff-power value by using the **power inline auto max 6300** interface configuration command, the configured maximum power allocation on the PoE port is 6.3 W (6300 mW). The switch provides power to the connected devices on the port if the device needs up to 6.3 W. If the CDP-power negotiated value or the IEEE classification value exceeds the configured cutoff value, the switch does not provide power to the connected device. After the switch turns on power on the PoE port, the switch does not police the real-time power consumption of the device, and the device can consume more power than the maximum allocated amount, which could adversely affect the switch and the devices connected to the other PoE ports.

Because a standalone switch supports internal power supplies, the total amount of power available for the powered devices varies depending on the power supply configuration.

- If a power supply is removed and replaced by a new power supply with less power and the switch does not have enough power for the powered devices, the switch denies power to the PoE ports in auto mode in descending order of the port numbers. If the switch still does not have enough power, the switch then denies power to the PoE ports in static mode in descending order of the port numbers.
- If the new power supply supports more power than the previous one and the switch now has more power available, the switch grants power to the PoE ports in static mode in ascending order of the port numbers. If it still has power available, the switch then grants power to the PoE ports in auto mode in ascending order of the port numbers.

The stacking-capable switch also supports StackPower, which allows power supplies to share the load across multiple systems in a stack by connecting the switches with power stack cables. You can collectively manage the power supplies of up to four stack members as a one large power supply.

How to Configure PoE

Configuring a Power Management Mode on a PoE Port

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **power inline** {**auto** [**max** *max-wattage*] | **never** | **static** [**max** *max-wattage*]}
4. **end**
5. **show power inline** [*interface-id* | **module** *switch-number*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/1`	Specifies the physical port to be configured, and enters interface configuration mode.
Step 3	**power inline** {**auto** [**max** *max-wattage*] | **never** | **static** [**max** *max-wattage*]} **Example:** `Switch(config-if)# power inline auto`	Configures the PoE mode on the port. The keywords have these meanings: • **auto—**Enables powered-device detection. If enough power is available, automatically allocates power to the PoE port after device detection. This is the default setting. • **max** *max-wattage***—**Limits the power allowed on the port. The range for PoE+ ports is 4000 to 30000 mW. If no value is specified, the maximum is allowed. • **never —**Disables device detection, and disable power to the port. **Note** If a port has a Cisco powered device connected to it, do not use the **power inline never** command to configure the port. A false link-up can occur, placing the port into the error-disabled state. • **static—**Enables powered-device detection. Pre-allocate (reserve) power for a port before the switch discovers the powered device. The switch reserves power for this port even when no device is connected and guarantees that power will be provided upon device detection. The switch allocates power to a port configured in static mode before it allocates power to a port configured in auto mode.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show power inline** [*interface-id* | **module** *switch-number*] **Example:** `Switch# show power inline`	Displays PoE status for a switch or a switch stack, for the specified interface, or for a specified stack member. The **module** *switch-number* keywords are supported only on stacking-capable switches.

Configuring Power Policing

By default, the switch monitors the real-time power consumption of connected powered devices. You can configure the switch to police the power usage. By default, policing is disabled.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **power inline police** [**action**{**log** | **errdisable**}]
4. **exit**
5. Use one of the following:
   - **errdisable detect cause inline-power**
   - **errdisable recovery cause inline-power**
   - **errdisable recovery interval** *interval*
6. **exit**
7. Use one of the following:
   - **show power inline police**
   - **show errdisable recovery**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/1`	Specifies the physical port to be configured, and enter interface configuration mode.
Step 3	**power inline police** [**action**{**log** | **errdisable**}] **Example:** `Switch(config-if)# power inline police`	If the real-time power consumption exceeds the maximum power allocation on the port, configures the switch to take one of these actions: • **power inline police—**Shuts down the PoE port, turns off power to it, and puts it in the error-disabled state.

	Command or Action	Purpose
		Note You can enable error detection for the PoE error-disabled cause by using the **errdisable detect cause inline-power** global configuration command. You can also enable the timer to recover from the PoE error-disabled state by using the **errdisable recovery cause inline-power interval** *interval* global configuration command. • **power inline police action errdisable—**Turns off power to the port if the real-time power consumption exceeds the maximum power allocation on the port. • **power inline police action log—**Generates a syslog message while still providing power to the port. If you do not enter the **action log** keywords, the default action shuts down the port and puts the port in the error-disabled state.
Step 4	**exit** **Example:** `Switch(config-if)# exit`	Returns to global configuration mode.

Step 5	Use one of the following:
	• **errdisable detect cause inline-power**
	• **errdisable recovery cause inline-power**
	• **errdisable recovery interval** *interval*
	Example:
	`Switch(config)# errdisable detect cause inline-power`
	`Switch(config)# errdisable recovery cause inline-power`
	`Switch(config)# errdisable recovery interval 100`
		(Optional) Enables error recovery from the PoE error-disabled state, and configures the PoE recover mechanism variables.
		By default, the recovery interval is 300 seconds.
		For **interval** *interval*, specifies the time in seconds to recover from the error-disabled state. The range is 30 to 86400.

Step 6	**exit** **Example:** `Switch(config)# exit`	Returns to privileged EXEC mode.
Step 7	Use one of the following: • **show power inline police** • **show errdisable recovery**	Displays the power monitoring status, and verify the error recovery settings.

	Command or Action	Purpose

	Example:
	`Switch# show power inline police`
	
	`Switch# show errdisable recovery`	

Monitoring Power Status

Table 58: Show Commands for Power Status

Command	Purpose
show env power switch [*switch-number*]	(Optional) Displays the status of the internal power supplies for each switch in the stack or for the specified switch. The range is 1 to 9, depending on the switch member numbers in the stack. These keywords are available only on stacking-capable switches.
show power inline [*interface-id* | **module** *switch-number*]	Displays PoE status for a switch or switch stack, for an interface, or for a specific switch in the stack.
show power inline police	Displays the power policing data.

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for PoE

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.
Cisco IOS XE 3.3SE	The **four-pair forced** keywords were added.

CHAPTER 27

Configuring EEE

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information About EEE

EEE Overview

Energy Efficient Ethernet (EEE) is an IEEE 802.3az standard that is designed to reduce power consumption in Ethernet networks during idle periods.

EEE can be enabled on devices that support low power idle (LPI) mode. Such devices can save power by entering LPI mode during periods of low utilization. In LPI mode, systems on both ends of the link can save power by shutting down certain services. EEE provides the protocol needed to transition into and out of LPI mode in a way that is transparent to upper layer protocols and applications.

Default EEE Configuration

EEE is disabled by default.

Restrictions for EEE

EEE has the following restrictions:

- Changing the EEE configuration resets the interface because the device has to restart Layer 1 autonegotiation.
- You might want to enable the Link Layer Discovery Protocol (LLDP) for devices that require longer wakeup times before they are able to accept data on their receive paths. Doing so enables the device to negotiate for extended system wakeup times from the transmitting link partner.

How to Configure EEE

You can enable or disable EEE on an interface that is connected to an EEE-capable link partner.

Enabling or Disabling EEE

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **power efficient-ethernet auto**
4. **no power efficient-ethernet auto**
5. **end**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Switch(config)#` **`interface gigabitethernet1/0/1`**	Specifies the interface to be configured, and enter interface configuration mode.
Step 3	**power efficient-ethernet auto** **Example:** `Switch(config-if)#` **`power efficient-ethernet auto`**	Enables EEE on the specified interface. When EEE is enabled, the device advertises and autonegotiates EEE to its link partner.
Step 4	**no power efficient-ethernet auto** **Example:** `Switch(config-if)#` **`no power efficient-ethernet auto`**	Disables EEE on the specified interface.
Step 5	**end** **Example:** `Switch(config-if)#` **`end`**	Returns to privileged EXEC mode.
Step 6	**copy running-config startup-config** **Example:** `Switch#` **`copy running-config startup-config`**	(Optional) Saves your entries in the configuration file.

Monitoring EEE

Table 59: Commands for Displaying EEE Settings

Command	Purpose
show eee capabilities interface *interface-id*	Displays EEE capabilities for the specified interface.
show eee status interface *interface-id*	Displays EEE status information for the specified interface.

Configuration Examples for Configuring EEE

This example shows how to enable EEE for an interface:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# power efficient-ethernet auto
```

This example shows how to disable EEE for an interface:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# no power efficient-ethernet auto
```

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Configuring EEE

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 28

Configuring Storm Control

This module contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information About Storm Control

Storm Control

Storm control prevents traffic on a LAN from being disrupted by a broadcast, multicast, or unicast storm on one of the physical interfaces. A LAN storm occurs when packets flood the LAN, creating excessive traffic and degrading network performance. Errors in the protocol-stack implementation, mistakes in network configurations, or users issuing a denial-of-service attack can cause a storm.

Storm control (or traffic suppression) monitors packets passing from an interface to the switching bus and determines if the packet is unicast, multicast, or broadcast. The switch counts the number of packets of a specified type received within the 1-second time interval and compares the measurement with a predefined suppression-level threshold.

How Traffic Activity is Measured

Storm control uses one of these methods to measure traffic activity:

- Bandwidth as a percentage of the total available bandwidth of the port that can be used by the broadcast, multicast, or unicast traffic
- Traffic rate in packets per second at which broadcast, multicast, or unicast packets are received
- Traffic rate in bits per second at which broadcast, multicast, or unicast packets are received
- Traffic rate in packets per second and for small frames. This feature is enabled globally. The threshold for small frames is configured for each interface.

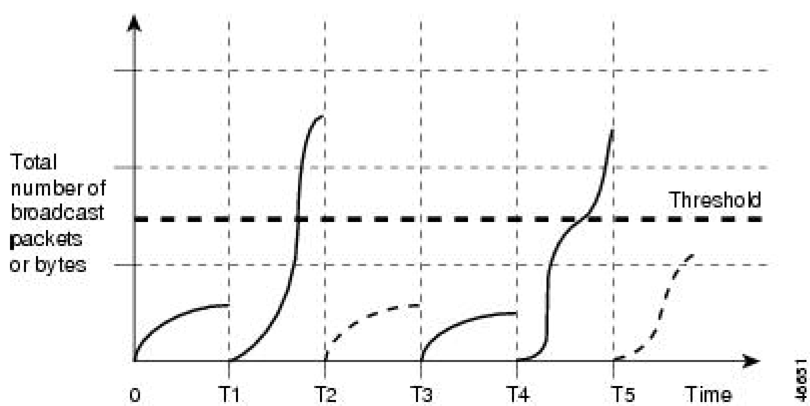
With each method, the port blocks traffic when the rising threshold is reached. The port remains blocked until the traffic rate drops below the falling threshold (if one is specified) and then resumes normal forwarding. If the falling suppression level is not specified, the switch blocks all traffic until the traffic rate drops below the rising suppression level. In general, the higher the level, the less effective the protection against broadcast storms.

Note When the storm control threshold for multicast traffic is reached, all multicast traffic except control traffic, such as bridge protocol data unit (BDPU) and Cisco Discovery Protocol (CDP) frames, are blocked. However, the switch does not differentiate between routing updates, such as OSPF, and regular multicast data traffic, so both types of traffic are blocked.

Traffic Patterns

This example shows broadcast traffic patterns on an interface over a given period of time.

Figure 11: Broadcast Storm Control Example



Broadcast traffic being forwarded exceeded the configured threshold between time intervals T1 and T2 and between T4 and T5. When the amount of specified traffic exceeds the threshold, all traffic of that kind is dropped for the next time period. Therefore, broadcast traffic is blocked during the intervals following T2 and T5. At the next time interval (for example, T3), if broadcast traffic does not exceed the threshold, it is again forwarded.

The combination of the storm-control suppression level and the 1-second time interval controls the way the storm control algorithm works. A higher threshold allows more packets to pass through. A threshold value of 100 percent means that no limit is placed on the traffic. A value of 0.0 means that all broadcast, multicast, or unicast traffic on that port is blocked.

Note Because packets do not arrive at uniform intervals, the 1-second time interval during which traffic activity is measured can affect the behavior of storm control.

You use the **storm-control** interface configuration commands to set the threshold value for each traffic type.

How to Configure Storm Control

Configuring Storm Control and Threshold Levels

You configure storm control on a port and enter the threshold level that you want to be used for a particular type of traffic.

However, because of hardware limitations and the way in which packets of different sizes are counted, threshold percentages are approximations. Depending on the sizes of the packets making up the incoming traffic, the actual enforced threshold might differ from the configured level by several percentage points.

Before You Begin

Storm control is supported on physical interfaces. You can also configure storm control on an EtherChannel. When storm control is configured on an EtherChannel, the storm control settings propagate to the EtherChannel physical interfaces.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **storm-control** {**broadcast** | **multicast** | **unicast**} **level** {*level* [*level-low*] | **bps** *bps* [*bps-low*] | **pps** *pps* [*pps-low*]}
4. **storm-control action** {**shutdown** | **trap**}
5. **end**
6. **show storm-control** [*interface-id*] [**broadcast** | **multicast** | **unicast**]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Switch(config)# ` **`interface gigabitethernet1/0/1`**	Specifies the interface to be configured, and enter interface configuration mode.
Step 3	**storm-control** {**broadcast** | **multicast** | **unicast**} **level** {*level* [*level-low*] | **bps** *bps* [*bps-low*] | **pps** *pps* [*pps-low*]} **Example:** `Switch(config-if)# ` **`storm-control unicast level 87 65`**	Configures broadcast, multicast, or unicast storm control. By default, storm control is disabled. The keywords have these meanings: • For *level*, specifies the rising threshold level for broadcast, multicast, or unicast traffic as a percentage (up to two decimal places) of the bandwidth. The port blocks traffic when the rising threshold is reached. The range is 0.00 to 100.00. • (Optional) For *level-low*, specifies the falling threshold level as a percentage (up to two decimal places) of the bandwidth. This value must be less than or equal to the rising suppression value. The port forwards traffic when traffic drops below this level. If you do not configure a falling suppression level, it is set to the rising suppression level. The range is 0.00 to 100.00. If you set the threshold to the maximum value (100 percent), no limit is placed on the traffic. If you set the threshold to 0.0, all broadcast, multicast, and unicast traffic on that port is blocked. • For **bps** *bps*, specifies the rising threshold level for broadcast, multicast, or unicast traffic in bits per second (up to one decimal place). The port blocks traffic when the rising threshold is reached. The range is 0.0 to 10000000000.0. • (Optional) For *bps-low*, specifies the falling threshold level in bits per second (up to one decimal place). It can be less than or equal to the rising threshold level. The port forwards traffic when traffic drops below this level. The range is 0.0 to 10000000000.0. • For **pps** *pps*, specifies the rising threshold level for broadcast, multicast, or unicast traffic in packets per second (up to one decimal place). The port blocks traffic when the rising threshold is reached. The range is 0.0 to 10000000000.0. • (Optional) For *pps-low*, specifies the falling threshold level in packets per second (up to one decimal place). It can be less than or equal to the rising threshold level. The port forwards traffic when traffic drops below this level. The range is **0.0 to** 10000000000.0. For BPS and PPS settings, you can use metric suffixes such as k, m, and g for large number thresholds.
Step 4	**storm-control action** {**shutdown** | **trap**}	Specifies the action to be taken when a storm is detected. The default is to filter out the traffic and not to send traps.

	Command or Action	Purpose
	Example: `Switch(config-if)# storm-control action trap`	• Select the **shutdown** keyword to error-disable the port during a storm. • Select the **trap** keyword to generate an SNMP trap when a storm is detected.
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**show storm-control** [*interface-id*] [**broadcast** | **multicast** | **unicast**] **Example:** `Switch# show storm-control gigabitethernet1/0/1 unicast`	Verifies the storm control suppression levels set on the interface for the specified traffic type. If you do not enter a traffic type, broadcast storm control settings are displayed.
Step 7	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring Storm Control

Table 60: Commands for Displaying Storm Control Status and Configuration

Command	Purpose
show interfaces [*interface-id*] **switchport**	Displays the administrative and operational status of all switching (nonrouting) ports or the specified port, including port blocking and port protection settings.
show storm-control [*interface-id*] [**broadcast** | **multicast** | **unicast**]	Displays storm control suppression levels set on all interfaces or the specified interface for the specified traffic type or for broadcast traffic if no traffic type is entered.

CHAPTER 29

Configuring Protected Port

This module contains the following topics:

Information About Protected Ports

Protected Ports

Some applications require that no traffic be forwarded at Layer 2 between ports on the same switch so that one neighbor does not see the traffic generated by another neighbor. In such an environment, the use of protected ports ensures that there is no exchange of unicast, broadcast, or multicast traffic between these ports on the switch.

Protected ports have these features:

- A protected port does not forward any traffic (unicast, multicast, or broadcast) to any other port that is also a protected port. Data traffic cannot be forwarded between protected ports at Layer 2; only control traffic, such as PIM packets, is forwarded because these packets are processed by the CPU and forwarded in software. All data traffic passing between protected ports must be forwarded through a Layer 3 device.
- Forwarding behavior between a protected port and a nonprotected port proceeds as usual.

Because a switch stack represents a single logical switch, Layer 2 traffic is not forwarded between any protected ports in the switch stack, whether they are on the same or different switches in the stack.

Default Protected Port Configuration

The default is to have no protected ports defined.

Protected Ports Guidelines

You can configure protected ports on a physical interface (for example, Gigabit Ethernet port 1) or an EtherChannel group (for example, port-channel 5). When you enable protected ports for a port channel, it is enabled for all ports in the port-channel group.

How to Configure Protected Ports

Configuring a Protected Port

Before You Begin

Protected ports are not pre-defined. This is the task to configure one.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport protected**
4. **end**
5. **show interfaces** *interface-id* **switchport**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Specifies the interface to be configured, and enter interface configuration mode.
Step 3	**switchport protected** **Example:** `Switch(config-if)# switchport protected`	Configures the interface to be a protected port.

	Command or Action	Purpose
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show interfaces** *interface-id* **switchport** **Example:** `Switch# show interfaces gigabitethernet1/0/1 switchport`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring Protected Ports

Table 61: Commands for Displaying Protected Port Settings

Command	Purpose
show interfaces [*interface-id*] **switchport**	Displays the administrative and operational status of all switching (nonrouting) ports or the specified port, including port blocking and port protection settings.

CHAPTER 30

Configuring Port Blocking

This module contains the following topics:

Information About Port Blocking

Port Blocking

By default, the switch floods packets with unknown destination MAC addresses out of all ports. If unknown unicast and multicast traffic is forwarded to a protected port, there could be security issues. To prevent unknown unicast or multicast traffic from being forwarded from one port to another, you can block a port (protected or nonprotected) from flooding unknown unicast or multicast packets to other ports.

Note With multicast traffic, the port blocking feature blocks only pure Layer 2 packets. Multicast packets that contain IPv4 or IPv6 information in the header are not blocked.

How to Configure Port Blocking

Blocking Flooded Traffic on an Interface

Before You Begin

The interface can be a physical interface or an EtherChannel group. When you block multicast or unicast traffic for a port channel, it is blocked on all ports in the port-channel group.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport block multicast**
4. **switchport block unicast**
5. **end**
6. **show interfaces** *interface-id* **switchport**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Specifies the interface to be configured, and enter interface configuration mode.
Step 3	**switchport block multicast** **Example:** `Switch(config-if)# switchport block multicast`	Blocks unknown multicast forwarding out of the port. **Note** Only pure Layer 2 multicast traffic is blocked. Multicast packets that contain IPv4 or IPv6 information in the header are not blocked.
Step 4	**switchport block unicast** **Example:** `Switch(config-if)# switchport block unicast`	Blocks unknown unicast forwarding out of the port.
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**show interfaces** *interface-id* **switchport** **Example:** `Switch# show interfaces gigabitethernet1/0/1`	Verifies your entries.

	Command or Action	Purpose
	`switchport`	
Step 7	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring Port Blocking

Table 62: Commands for Displaying Port Blocking Settings

Command	Purpose
show interfaces [*interface-id*] **switchport**	Displays the administrative and operational status of all switching (nonrouting) ports or the specified port, including port blocking and port protection settings.

CHAPTER 31

Configuring Port Security

This module contains the following topics:

Prerequisites for Port Security

Note If you try to set the maximum value to a number less than the number of secure addresses already configured on an interface, the command is rejected.

Restrictions for Port Security

The maximum number of secure MAC addresses that you can configure on a switch or switch stack is set by the maximum number of available MAC addresses allowed in the system. This number is determined by the active Switch Database Management (SDM) template. This number is the total of available MAC addresses, including those used for other Layer 2 functions and any other secure MAC addresses configured on interfaces.

Information About Port Security

Port Security

You can use the port security feature to restrict input to an interface by limiting and identifying MAC addresses of the stations allowed to access the port. When you assign secure MAC addresses to a secure port, the port does not forward packets with source addresses outside the group of defined addresses. If you limit the number of secure MAC addresses to one and assign a single secure MAC address, the workstation attached to that port is assured the full bandwidth of the port.

If a port is configured as a secure port and the maximum number of secure MAC addresses is reached, when the MAC address of a station attempting to access the port is different from any of the identified secure MAC addresses, a security violation occurs. Also, if a station with a secure MAC address configured or learned on one secure port attempts to access another secure port, a violation is flagged.

Related Topics

Enabling and Configuring Port Security, on page 504
Configuration Examples for Port Security, on page 511
Enabling and Configuring Port Security, on page 504
Configuration Examples for Port Security, on page 511

Types of Secure MAC Addresses

The switch supports these types of secure MAC addresses:

- Static secure MAC addresses—These are manually configured by using the **switchport port-security mac-address** *mac-address* interface configuration command, stored in the address table, and added to the switch running configuration.
- Dynamic secure MAC addresses—These are dynamically configured, stored only in the address table, and removed when the switch restarts.
- Sticky secure MAC addresses—These can be dynamically learned or manually configured, stored in the address table, and added to the running configuration. If these addresses are saved in the configuration file, when the switch restarts, the interface does not need to dynamically reconfigure them.

Sticky Secure MAC Addresses

You can configure an interface to convert the dynamic MAC addresses to sticky secure MAC addresses and to add them to the running configuration by enabling sticky learning. The interface converts all the dynamic secure MAC addresses, including those that were dynamically learned before sticky learning was enabled, to sticky secure MAC addresses. All sticky secure MAC addresses are added to the running configuration.

The sticky secure MAC addresses do not automatically become part of the configuration file, which is the startup configuration used each time the switch restarts. If you save the sticky secure MAC addresses in the configuration file, when the switch restarts, the interface does not need to relearn these addresses. If you do not save the sticky secure addresses, they are lost.

If sticky learning is disabled, the sticky secure MAC addresses are converted to dynamic secure addresses and are removed from the running configuration.

Security Violations

It is a security violation when one of these situations occurs:

- The maximum number of secure MAC addresses have been added to the address table, and a station whose MAC address is not in the address table attempts to access the interface.
- An address learned or configured on one secure interface is seen on another secure interface in the same VLAN.

You can configure the interface for one of three violation modes, based on the action to be taken if a violation occurs:

- protect—when the number of secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. You are not notified that a security violation has occurred.

  Note We do not recommend configuring the protect violation mode on a trunk port. The protect mode disables learning when any VLAN reaches its maximum limit, even if the port has not reached its maximum limit.

- restrict—when the number of secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. In this mode, you are notified that a security violation has occurred. An SNMP trap is sent, a syslog message is logged, and the violation counter increments.
- shutdown—a port security violation causes the interface to become error-disabled and to shut down immediately, and the port LED turns off. When a secure port is in the error-disabled state, you can bring it out of this state by entering the **errdisable recovery cause** *psecure-violation* global configuration command, or you can manually re-enable it by entering the **shutdown** and **no shut down** interface configuration commands. This is the default mode.
- shutdown vlan—Use to set the security violation mode per-VLAN. In this mode, the VLAN is error disabled instead of the entire port when a violation occurs

This table shows the violation mode and the actions taken when you configure an interface for port security.

Table 63: Security Violation Mode Actions

Violation Mode	Traffic is forwarded [3]	Sends SNMP trap	Sends syslog message	Displays error message [4]	Violation counter increments	Shuts down port
protect	No	No	No	No	No	No

Violation Mode	Traffic is forwarded [3]	Sends SNMP trap	Sends syslog message	Displays error message [4]	Violation counter increments	Shuts down port
restrict	No	Yes	Yes	No	Yes	No
shutdown	No	No	No	No	Yes	Yes
shutdown vlan	No	No	Yes	No	Yes	No [5]

3 Packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses.
4 The switch returns an error message if you manually configure an address that would cause a security violation.
5 Shuts down only the VLAN on which the violation occurred.

Port Security Aging

You can use port security aging to set the aging time for all secure addresses on a port. Two types of aging are supported per port:

- Absolute—The secure addresses on the port are deleted after the specified aging time.
- Inactivity—The secure addresses on the port are deleted only if the secure addresses are inactive for the specified aging time.

Related Topics

Enabling and Configuring Port Security Aging, on page 508
Enabling and Configuring Port Security Aging, on page 508

Port Security and Switch Stacks

When a switch joins a stack, the new switch will get the configured secure addresses. All dynamic secure addresses are downloaded by the new stack member from the other stack members.

When a switch (either the active switch or a stack member) leaves the stack, the remaining stack members are notified, and the secure MAC addresses configured or learned by that switch are deleted from the secure MAC address table.

Default Port Security Configuration

Table 64: Default Port Security Configuration

Feature	Default Setting
Port security	Disabled on a port.
Sticky address learning	Disabled.

Feature	Default Setting
Maximum number of secure MAC addresses per port	1.
Violation mode	Shutdown. The port shuts down when the maximum number of secure MAC addresses is exceeded.
Port security aging	Disabled. Aging time is 0. Static aging is disabled. Type is absolute.

Port Security Configuration Guidelines

- Port security can only be configured on static access ports or trunk ports.
- A secure port cannot be a destination port for Switched Port Analyzer (SPAN).
- **Note** Voice VLAN is only supported on access ports and not on trunk ports, even though the configuration is allowed.
- When you enable port security on an interface that is also configured with a voice VLAN, set the maximum allowed secure addresses on the port to two. When the port is connected to a Cisco IP phone, the IP phone requires one MAC address. The Cisco IP phone address is learned on the voice VLAN, but is not learned on the access VLAN. If you connect a single PC to the Cisco IP phone, no additional MAC addresses are required. If you connect more than one PC to the Cisco IP phone, you must configure enough secure addresses to allow one for each PC and one for the phone.
- When a trunk port configured with port security and assigned to an access VLAN for data traffic and to a voice VLAN for voice traffic, entering the **switchport voice** and **switchport priority extend** interface configuration commands has no effect.

  When a connected device uses the same MAC address to request an IP address for the access VLAN and then an IP address for the voice VLAN, only the access VLAN is assigned an IP address.
- When you enter a maximum secure address value for an interface, and the new value is greater than the previous value, the new value overwrites the previously configured value. If the new value is less than the previous value and the number of configured secure addresses on the interface exceeds the new value, the command is rejected.
- The switch does not support port security aging of sticky secure MAC addresses.

This table summarizes port security compatibility with other port-based features.

Table 65: Port Security Compatibility with Other Switch Features

Type of Port or Feature on Port	Compatible with Port Security
DTP [6] port [7]	No
Trunk port	Yes
Routed port	No
SPAN source port	Yes
SPAN destination port	No
EtherChannel	Yes
Tunneling port	Yes
Protected port	Yes
IEEE 802.1x port	Yes
Voice VLAN port [8]	Yes
IP source guard	Yes
Dynamic Address Resolution Protocol (ARP) inspection	Yes
Flex Links	Yes

6 DTP=Dynamic Trunking Protocol

7 A port configured with the **switchport mode dynamic** interface configuration command.

8 You must set the maximum allowed secure addresses on the port to two plus the maximum number of secure addresses allowed on the access VLAN.

How to Configure Port Security

Enabling and Configuring Port Security

Before You Begin

This task restricts input to an interface by limiting and identifying MAC addresses of the stations allowed to access the port:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode** {**access** | **trunk**}
4. **switchport voice vlan** *vlan-id*
5. **switchport port-security**
6. **switchport port-security [maximum** *value* [**vlan** {*vlan-list* | {**access** | **voice**}}]]
7. **switchport port-security violation** {**protect** | **restrict** | **shutdown** | **shutdown vlan**}
8. **switchport port-security [mac-address** *mac-address* [**vlan** {*vlan-id* | {**access** | **voice**}}]
9. **switchport port-security mac-address sticky**
10. **switchport port-security mac-address sticky** [*mac-address* | **vlan** {*vlan-id* | {**access** | **voice**}}]
11. **end**
12. **show port-security**
13. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Specifies the interface to be configured, and enter interface configuration mode.
Step 3	**switchport mode** {**access** | **trunk**} **Example:** `Switch(config-if)# switchport mode access`	Sets the interface switchport mode as access or trunk; an interface in the default mode (dynamic auto) cannot be configured as a secure port.
Step 4	**switchport voice vlan** *vlan-id* **Example:** `Switch(config-if)# switchport voice vlan 22`	Enables voice VLAN on a port. vlan-id—Specifies the VLAN to be used for voice traffic.

	Command or Action	Purpose
Step 5	**switchport port-security** **Example:** `Switch(config-if)# switchport port-security`	Enable port security on the interface.
Step 6	**switchport port-security [maximum** *value* [**vlan** {*vlan-list* | {**access** | **voice**}}]] **Example:** `Switch(config-if)# switchport port-security maximum 20`	(Optional) Sets the maximum number of secure MAC addresses for the interface. The maximum number of secure MAC addresses that you can configure on a switch or switch stack is set by the maximum number of available MAC addresses allowed in the system. This number is set by the active Switch Database Management (SDM) template. This number is the total of available MAC addresses, including those used for other Layer 2 functions and any other secure MAC addresses configured on interfaces. (Optional) **vlan—**sets a per-VLAN maximum value Enter one of these options after you enter the **vlan** keyword: • *vlan-list*—On a trunk port, you can set a per-VLAN maximum value on a range of VLANs separated by a hyphen or a series of VLANs separated by commas. For nonspecified VLANs, the per-VLAN maximum value is used. • **access—**On an access port, specifies the VLAN as an access VLAN. • **voice—**On an access port, specifies the VLAN as a voice VLAN. **Note** The **voice** keyword is available only if a voice VLAN is configured on a port and if that port is not the access VLAN. If an interface is configured for voice VLAN, configure a maximum of two secure MAC addresses.
Step 7	**switchport port-security violation** {**protect** | **restrict** | **shutdown** | **shutdown vlan**} **Example:** `Switch(config-if)# switchport port-security violation restrict`	(Optional) Sets the violation mode, the action to be taken when a security violation is detected, as one of these: • **protect—**When the number of port secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. You are not notified that a security violation has occurred. **Note** We do not recommend configuring the protect mode on a trunk port. The protect mode disables learning when any VLAN reaches its maximum limit, even if the port has not reached its maximum limit. • **restrict—**When the number of secure MAC addresses reaches the limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses or increase the number of maximum allowable addresses. An SNMP trap is sent, a syslog message is logged, and the violation counter increments.

	Command or Action	Purpose
		• **shutdown—**The interface is error-disabled when a violation occurs, and the port LED turns off. An SNMP trap is sent, a syslog message is logged, and the violation counter increments. • **shutdown vlan—**Use to set the security violation mode per VLAN. In this mode, the VLAN is error disabled instead of the entire port when a violation occurs. **Note** When a secure port is in the error-disabled state, you can bring it out of this state by entering the **errdisable recovery cause psecure-violation** global configuration command. You can manually re-enable it by entering the **shutdown** and **no shutdown** interface configuration commands or by using the **clear errdisable interface vlan** privileged EXEC command.
Step 8	**switchport port-security [mac-address** *mac-address* [**vlan** {*vlan-id* | {**access** | **voice**}}] **Example:** `Switch(config-if)# switchport port-security mac-address 00:A0:C7:12:C9:25 vlan 3 voice`	(Optional) Enters a secure MAC address for the interface. You can use this command to enter the maximum number of secure MAC addresses. If you configure fewer secure MAC addresses than the maximum, the remaining MAC addresses are dynamically learned. **Note** If you enable sticky learning after you enter this command, the secure addresses that were dynamically learned are converted to sticky secure MAC addresses and are added to the running configuration. (Optional) **vlan—**sets a per-VLAN maximum value. Enter one of these options after you enter the **vlan** keyword: • *vlan-id*—On a trunk port, you can specify the VLAN ID and the MAC address. If you do not specify a VLAN ID, the native VLAN is used. • **access—**On an access port, specifies the VLAN as an access VLAN. • **voice—**On an access port, specifies the VLAN as a voice VLAN. **Note** The **voice** keyword is available only if a voice VLAN is configured on a port and if that port is not the access VLAN. If an interface is configured for voice VLAN, configure a maximum of two secure MAC addresses.
Step 9	**switchport port-security mac-address sticky** **Example:** `Switch(config-if)# switchport port-security mac-address sticky`	(Optional) Enables sticky learning on the interface.
Step 10	**switchport port-security mac-address sticky** [*mac-address* | **vlan** {*vlan-id* | {**access** | **voice**}}] **Example:** `Switch(config-if)# switchport port-security mac-address sticky`	(Optional) Enters a sticky secure MAC address, repeating the command as many times as necessary. If you configure fewer secure MAC addresses than the maximum, the remaining MAC addresses are dynamically learned, are converted to sticky secure MAC addresses, and are added to the running configuration. **Note** If you do not enable sticky learning before this command is entered, an error message appears, and you cannot enter a sticky secure MAC address. (Optional) **vlan—**sets a per-VLAN maximum value.

	Command or Action	Purpose
	`00:A0:C7:12:C9:25 vlan voice`	Enter one of these options after you enter the **vlan** keyword: • *vlan-id*—On a trunk port, you can specify the VLAN ID and the MAC address. If you do not specify a VLAN ID, the native VLAN is used. • **access**—On an access port, specifies the VLAN as an access VLAN. • **voice**—On an access port, specifies the VLAN as a voice VLAN. **Note** The **voice** keyword is available only if a voice VLAN is configured on a port and if that port is not the access VLAN.
Step 11	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 12	**show port-security** **Example:** `Switch# show port-security`	Verifies your entries.
Step 13	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Port Security, on page 500
Configuration Examples for Port Security, on page 511
Port Security, on page 500
Configuration Examples for Port Security, on page 511

Enabling and Configuring Port Security Aging

Use this feature to remove and add devices on a secure port without manually deleting the existing secure MAC addresses and to still limit the number of secure addresses on a port. You can enable or disable the aging of secure addresses on a per-port basis.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport port-security aging** {**static** | **time** *time* | **type** {**absolute** | **inactivity**}}
4. **end**
5. **show port-security** [**interface** *interface-id*] [**address**]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Specifies the interface to be configured, and enter interface configuration mode.
Step 3	**switchport port-security aging** {**static** | **time** *time* | **type** {**absolute** | **inactivity**}} **Example:** `Switch(config-if)# switchport port-security aging time 120`	Enables or disable static aging for the secure port, or set the aging time or type. **Note** The switch does not support port security aging of sticky secure addresses. Enter **static** to enable aging for statically configured secure addresses on this port. For *time*, specifies the aging time for this port. The valid range is from 0 to 1440 minutes. For **type**, select one of these keywords: • **absolute—**Sets the aging type as absolute aging. All the secure addresses on this port age out exactly after the time (minutes) specified lapses and are removed from the secure address list. • **inactivity—**Sets the aging type as inactivity aging. The secure addresses on this port age out only if there is no data traffic from the secure source addresses for the specified time period.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	**show port-security** [**interface** *interface-id*] [**address**] **Example:** `Switch# show port-security interface gigabitethernet1/0/1`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Port Security Aging, on page 502

Port Security Aging, on page 502

Monitoring Port Security

This table displays port security information.

Table 66: Commands for Displaying Port Security Status and Configuration

Command	Purpose
show port-security [**interface** *interface-id*]	Displays port security settings for the switch or for the specified interface, including the maximum allowed number of secure MAC addresses for each interface, the number of secure MAC addresses on the interface, the number of security violations that have occurred, and the violation mode.
show port-security [**interface** *interface-id*] **address**	Displays all secure MAC addresses configured on all switch interfaces or on a specified interface with aging information for each address.
show port-security interface *interface-id* **vlan**	Displays the number of secure MAC addresses configured per VLAN on the specified interface.

Configuration Examples for Port Security

This example shows how to enable port security on a port and to set the maximum number of secure addresses to 50. The violation mode is the default, no static secure MAC addresses are configured, and sticky learning is enabled.

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security maximum 50
Switch(config-if)# switchport port-security mac-address sticky
```

This example shows how to configure a static secure MAC address on VLAN 3 on a port:

```
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security mac-address 0000.0200.0004 vlan 3
```

This example shows how to enable sticky port security on a port, to manually configure MAC addresses for data VLAN and voice VLAN, and to set the total maximum number of secure addresses to 20 (10 for data VLAN and 10 for voice VLAN).

```
Switch(config)# interface tengigabitethernet1/0/1
Switch(config-if)# switchport access vlan 21
Switch(config-if)# switchport mode access
Switch(config-if)# switchport voice vlan 22
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security maximum 20
Switch(config-if)# switchport port-security violation restrict
Switch(config-if)# switchport port-security mac-address sticky
Switch(config-if)# switchport port-security mac-address sticky 0000.0000.0002
Switch(config-if)# switchport port-security mac-address 0000.0000.0003
Switch(config-if)# switchport port-security mac-address sticky 0000.0000.0001 vlan voice
Switch(config-if)# switchport port-security mac-address 0000.0000.0004 vlan voice
Switch(config-if)# switchport port-security maximum 10 vlan access
Switch(config-if)# switchport port-security maximum 10 vlan voice
```

Related Topics

Port Security, on page 500
Enabling and Configuring Port Security, on page 504
Port Security, on page 500
Enabling and Configuring Port Security, on page 504

PART **IV**

VLAN

CHAPTER 32

Configuring VTP

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for VTP

Before you create VLANs, you must decide whether to use the VLAN Trunking Protocol (VTP) in your network. Using VTP, you can make configuration changes centrally on one or more switches and have those changes automatically communicated to all the other switches in the network. Without VTP, you cannot send information about VLANs to other switches.

VTP is designed to work in an environment where updates are made on a single switch and are sent through VTP to other switches in the domain. It does not work well in a situation where multiple updates to the VLAN database occur simultaneously on switches in the same domain, which would result in an inconsistency in the VLAN database.

The switch supports 1005 VLANs when running the IP base or IP services feature set and 255 VLANs when running the LAN base feature set. The switch supports a total of 4094 VLANs. However, the number of routed ports, SVIs, and other configured features affects the usage of the switch hardware. If the switch is notified by VTP of a new VLAN and the switch is already using the maximum available hardware resources, it sends a message that there are not enough hardware resources available and shuts down the VLAN. The output of the **show vlan** user EXEC command shows the VLAN in a suspended state.

Because trunk ports send and receive VTP advertisements, you must ensure that at least one trunk port is configured on the switch or switch stack and that this trunk port is connected to the trunk port of another switch. Otherwise, the switch cannot receive any VTP advertisements.

Related Topics

VTP Advertisements, on page 519

Adding a VTP Client Switch to a VTP Domain , on page 534

VTP Domain, on page 517

VTP Modes, on page 518

Restrictions for VTP

The following are restrictions for a VTP:

- You cannot have a switch stack containing a mix of Catalyst 3850 and Catalyst 3650 switches.

Caution Before adding a VTP client switch to a VTP domain, always verify that its VTP configuration revision number is lower than the configuration revision number of the other switches in the VTP domain. Switches in a VTP domain always use the VLAN configuration of the switch with the highest VTP configuration revision number. If you add a switch that has a revision number higher than the revision number in the VTP domain, it can erase all VLAN information from the VTP server and VTP domain.

Information About VTP

VTP

VTP is a Layer 2 messaging protocol that maintains VLAN configuration consistency by managing the addition, deletion, and renaming of VLANs on a network-wide basis. VTP minimizes misconfigurations and configuration inconsistencies that can cause several problems, such as duplicate VLAN names, incorrect VLAN-type specifications, and security violations.

VTP functionality is supported across the stack, and all switches in the stack maintain the same VLAN and VTP configuration inherited from the active switch. When a switch learns of a new VLAN through VTP messages or when a new VLAN is configured by the user, the new VLAN information is communicated to all switches in the stack.

When a switch joins the stack or when stacks merge, the new switches get VTP information from the active switch.

VTP Domain

A VTP domain (also called a VLAN management domain) consists of one switch or several interconnected switches or switch stacks under the same administrative responsibility sharing the same VTP domain name. A switch can be in only one VTP domain. You make global VLAN configuration changes for the domain.

By default, the switch is in the VTP no-management-domain state until it receives an advertisement for a domain over a trunk link (a link that carries the traffic of multiple VLANs) or until you configure a domain name. Until the management domain name is specified or learned, you cannot create or modify VLANs on a VTP server, and VLAN information is not propagated over the network.

If the switch receives a VTP advertisement over a trunk link, it inherits the management domain name and the VTP configuration revision number. The switch then ignores advertisements with a different domain name or an earlier configuration revision number.

When you make a change to the VLAN configuration on a VTP server, the change is propagated to all switches in the VTP domain. VTP advertisements are sent over all IEEE trunk connections, including IEEE 802.1Q. VTP dynamically maps VLANs with unique names and internal index associates across multiple LAN types. Mapping eliminates excessive device administration required from network administrators.

If you configure a switch for VTP transparent mode, you can create and modify VLANs, but the changes are not sent to other switches in the domain, and they affect only the individual switch. However, configuration changes made when the switch is in this mode are saved in the switch running configuration and can be saved to the switch startup configuration file.

Related Topics

Adding a VTP Client Switch to a VTP Domain , on page 534

Prerequisites for VTP, on page 515

VTP Modes

Table 67: VTP Modes

VTP Mode	Description
VTP server	In VTP server mode, you can create, modify, and delete VLANs, and specify other configuration parameters (such as the VTP version) for the entire VTP domain. VTP servers advertise their VLAN configurations to other switches in the same VTP domain and synchronize their VLAN configurations with other switches based on advertisements received over trunk links. VTP server is the default mode. In VTP server mode, VLAN configurations are saved in NVRAM. If the switch detects a failure while writing a configuration to NVRAM, VTP mode automatically changes from server mode to client mode. If this happens, the switch cannot be returned to VTP server mode until the NVRAM is functioning.
VTP client	A VTP client functions like a VTP server and transmits and receives VTP updates on its trunks, but you cannot create, change, or delete VLANs on a VTP client. VLANs are configured on another switch in the domain that is in server mode. In VTP versions 1 and 2 in VTP client mode, VLAN configurations are not saved in NVRAM. In VTP version 3, VLAN configurations are saved in NVRAM in client mode.
VTP transparent	VTP transparent switches do not participate in VTP. A VTP transparent switch does not advertise its VLAN configuration and does not synchronize its VLAN configuration based on received advertisements. However, in VTP version 2 or version 3, transparent switches do forward VTP advertisements that they receive from other switches through their trunk interfaces. You can create, modify, and delete VLANs on a switch in VTP transparent mode. In VTP versions 1 and 2, the switch must be in VTP transparent mode when you create extended-range VLANs. VTP version 3 also supports creating extended-range VLANs in client or server mode. When the switch is in VTP transparent mode, the VTP and VLAN configurations are saved in NVRAM, but they are not advertised to other switches. In this mode, VTP mode and domain name are saved in the switch running configuration, and you can save this information in the switch startup configuration file by using the **copy running-config startup-config** privileged EXEC command. In a switch stack, the running configuration and the saved configuration are the same for all switches in a stack.
VTP off	A switch in VTP off mode functions in the same manner as a VTP transparent switch, except that it does not forward VTP advertisements on trunks.

Related Topics

Prerequisites for VTP, on page 515

Configuring VTP Mode , on page 526

VTP Advertisements

Each switch in the VTP domain sends periodic global configuration advertisements from each trunk port to a reserved multicast address. Neighboring switches receive these advertisements and update their VTP and VLAN configurations as necessary.

VTP advertisements distribute this global domain information:

- VTP domain name
- VTP configuration revision number
- Update identity and update timestamp
- MD5 digest VLAN configuration, including maximum transmission unit (MTU) size for each VLAN
- Frame format

VTP advertisements distribute this VLAN information for each configured VLAN:

- VLAN IDs (including IEEE 802.1Q)
- VLAN name
- VLAN type
- VLAN state
- Additional VLAN configuration information specific to the VLAN type

In VTP version 3, VTP advertisements also include the primary server ID, an instance number, and a start index.

Related Topics

Prerequisites for VTP, on page 515

VTP Version 2

If you use VTP in your network, you must decide which version of VTP to use. By default, VTP operates in version 1.

VTP version 2 supports these features that are not supported in version 1:

- Token Ring support—VTP version 2 supports Token Ring Bridge Relay Function (TrBRF) and Token Ring Concentrator Relay Function (TrCRF) VLANs.
- Unrecognized Type-Length-Value (TLV) support—A VTP server or client propagates configuration changes to its other trunks, even for TLVs it is not able to parse. The unrecognized TLV is saved in NVRAM when the switch is operating in VTP server mode.
- Version-Dependent Transparent Mode—In VTP version 1, a VTP transparent switch inspects VTP messages for the domain name and version and forwards a message only if the version and domain name match. Although VTP version 2 supports only one domain, a VTP version 2 transparent switch forwards a message only when the domain name matches.
- Consistency Checks—In VTP version 2, VLAN consistency checks (such as VLAN names and values) are performed only when you enter new information through the CLI or SNMP. Consistency checks are

not performed when new information is obtained from a VTP message or when information is read from NVRAM. If the MD5 digest on a received VTP message is correct, its information is accepted.

Related Topics

Enabling the VTP Version , on page 530

VTP Version 3

VTP version 3 supports these features that are not supported in version 1 or version 2:

- Enhanced authentication—You can configure the authentication as **hidden** or **secret**. When **hidden**, the secret key from the password string is saved in the VLAN database file, but it does not appear in plain text in the configuration. Instead, the key associated with the password is saved in hexadecimal format in the running configuration. You must reenter the password if you enter a takeover command in the domain. When you enter the **secret** keyword, you can directly configure the password secret key.
- Support for extended range VLAN (VLANs 1006 to 4094) database propagation—VTP versions 1 and 2 propagate only VLANs 1 to 1005. If extended VLANs are configured, you cannot convert from VTP version 3 to version 1 or 2.

  Note VTP pruning still applies only to VLANs 1 to 1005, and VLANs 1002 to 1005 are still reserved and cannot be modified.

- Support for any database in a domain—In addition to propagating VTP information, version 3 can propagate Multiple Spanning Tree (MST) protocol database information. A separate instance of the VTP protocol runs for each application that uses VTP.
- VTP primary server and VTP secondary servers—A VTP primary server updates the database information and sends updates that are honored by all devices in the system. A VTP secondary server can only back up the updated VTP configurations received from the primary server to its NVRAM.

  By default, all devices come up as secondary servers. You can enter the **vtp primary** privileged EXEC command to specify a primary server. Primary server status is only needed for database updates when the administrator issues a takeover message in the domain. You can have a working VTP domain without any primary servers. Primary server status is lost if the device reloads or domain parameters change, even when a password is configured on the switch.
- The option to turn VTP on or off on a per-trunk (per-port) basis—You can enable or disable VTP per port by entering the [**no**] **vtp** interface configuration command. When you disable VTP on trunking ports, all VTP instances for that port are disabled. You cannot set VTP to *off* for the MST database and *on* for the VLAN database on the same port.

  When you globally set VTP mode to off, it applies to all the trunking ports in the system. However, you can specify on or off on a per-VTP instance basis. For example, you can configure the switch as a VTP server for the VLAN database but with VTP *off* for the MST database.

Related Topics

Enabling the VTP Version , on page 530

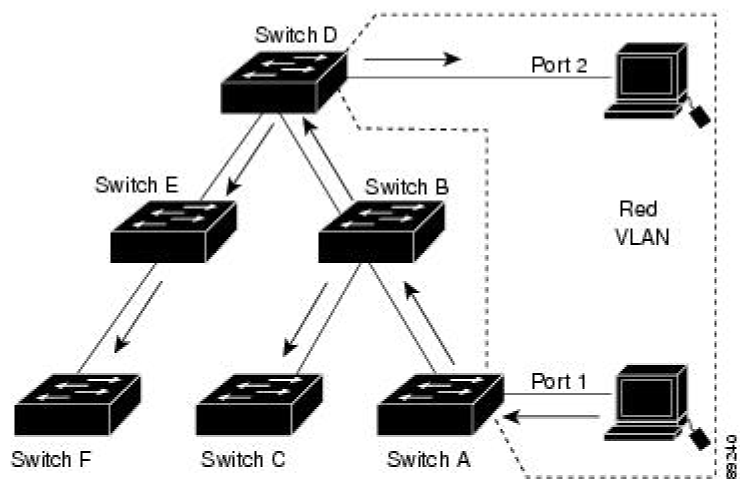
VTP Pruning

VTP pruning increases network available bandwidth by restricting flooded traffic to those trunk links that the traffic must use to reach the destination devices. Without VTP pruning, a switch floods broadcast, multicast, and unknown unicast traffic across all trunk links within a VTP domain even though receiving switches might discard them. VTP pruning is disabled by default.

VTP pruning blocks unneeded flooded traffic to VLANs on trunk ports that are included in the pruning-eligible list. Only VLANs included in the pruning-eligible list can be pruned. By default, VLANs 2 through 1001 are pruning eligible switch trunk ports. If the VLANs are configured as pruning-ineligible, the flooding continues. VTP pruning is supported in all VTP versions.

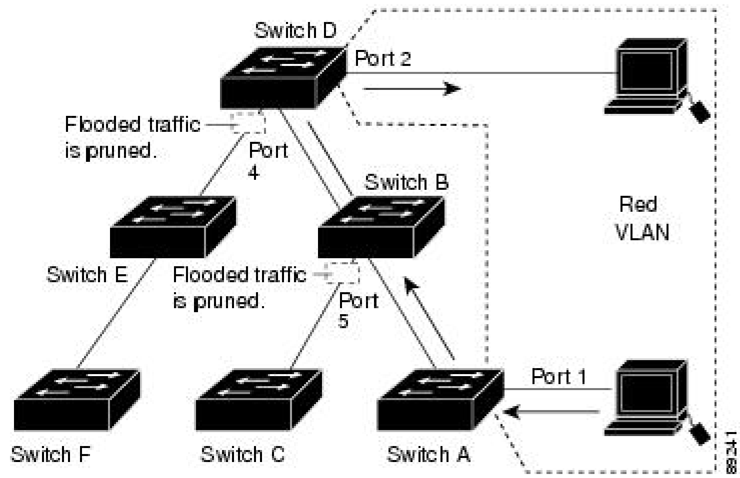
VTP pruning is disabled in the switched network. Port 1 on Switch A and Port 2 on Switch D are assigned to the Red VLAN. If a broadcast is sent from the host connected to Switch A, Switch A floods the broadcast and every switch in the network receives it, even though Switches C, E, and F have no ports in the Red VLAN.

Figure 12: Flooding Traffic without VTP Pruning



VTP pruning is enabled in the switched network. The broadcast traffic from Switch A is not forwarded to Switches C, E, and F because traffic for the Red VLAN has been pruned on the links shown (Port 5 on Switch B and Port 4 on Switch D).

Figure 13: Optimized Flooded Traffic VTP Pruning



Enabling VTP pruning on a VTP server enables pruning for the entire management domain. Making VLANs pruning-eligible or pruning-ineligible affects pruning eligibility for those VLANs on that trunk only (not on all switches in the VTP domain).

VTP pruning takes effect several seconds after you enable it. VTP pruning does not prune traffic from VLANs that are pruning-ineligible. VLAN 1 and VLANs 1002 to 1005 are always pruning-ineligible; traffic from these VLANs cannot be pruned. Extended-range VLANs (VLAN IDs higher than 1005) are also pruning-ineligible.

Related Topics

Enabling VTP Pruning , on page 531

VTP and Switch Stacks

VTP configuration is the same in all members of a switch stack. When the switch stack is in VTP server or client mode, all switches in the stack carry the same VTP configuration. When VTP mode is transparent, the stack is not taking part in VTP.

- When a switch joins the stack, it inherits the VTP and VLAN properties of the active switch.
- All VTP updates are carried across the stack.
- When VTP mode is changed in a switch in the stack, the other switches in the stack also change VTP mode, and the switch VLAN database remains consistent.

VTP version 3 functions the same on a standalone switch or a stack except when the switch stack is the primary server for the VTP database. In this case, the MAC address of the active switch is used as the primary server ID. If the active switch reloads or is powered off, a new active switch is elected.

- If you do not configure the persistent MAC address feature, when the new active switch is elected, it sends a takeover message with the new active MAC address as the primary server.

- If a persistent MAC address is configured, the new active switch waits for the configured timer value. If the previous active switch does not rejoin the stack during this time, then the new active switch issues the takeover message.

VTP Configuration Guidelines

VTP Configuration Requirements

When you configure VTP, you must configure a trunk port so that the switch can send and receive VTP advertisements to and from other switches in the domain.

In VTP versions 1 and 2, when you configure extended-range VLANs on the switch, the switch must be in VTP transparent mode. VTP version 3 also supports creating extended-range VLANs in client or server mode.

VTP Settings

The VTP information is saved in the VTP VLAN database. When VTP mode is transparent, the VTP domain name and mode are also saved in the switch running configuration file, and you can save it in the switch startup configuration file by entering the **copy running-config startup-config** privileged EXEC command. You must use this command if you want to save VTP mode as transparent, even if the switch resets.

When you save VTP information in the switch startup configuration file and reboot the switch, the switch configuration is selected as follows:

- If the VTP mode is transparent in the startup configuration and the VLAN database and the VTP domain name from the VLAN database matches that in the startup configuration file, the VLAN database is ignored (cleared), and the VTP and VLAN configurations in the startup configuration file are used. The VLAN database revision number remains unchanged in the VLAN database.
- If the VTP mode or domain name in the startup configuration do not match the VLAN database, the domain name and VTP mode and configuration for VLAN IDs 1 to 1005 use the VLAN database information.

Related Topics

Configuring VTP on a Per-Port Basis , on page 533

Configuring a VTP Version 3 Primary Server , on page 529

Domain Names for Configuring VTP

When configuring VTP for the first time, you must always assign a domain name. You must configure all switches in the VTP domain with the same domain name. Switches in VTP transparent mode do not exchange VTP messages with other switches, and you do not need to configure a VTP domain name for them.

Note If the NVRAM and DRAM storage is sufficient, all switches in a VTP domain should be in VTP server mode.

> **Caution** Do not configure a VTP domain if all switches are operating in VTP client mode. If you configure the domain, it is impossible to make changes to the VLAN configuration of that domain. Make sure that you configure at least one switch in the VTP domain for VTP server mode.

Related Topics

Adding a VTP Client Switch to a VTP Domain , on page 534

Passwords for the VTP Domain

You can configure a password for the VTP domain, but it is not required. If you do configure a domain password, all domain switches must share the same password and you must configure the password on each switch in the management domain. Switches without a password or with the wrong password reject VTP advertisements.

If you configure a VTP password for a domain, a switch that is booted without a VTP configuration does not accept VTP advertisements until you configure it with the correct password. After the configuration, the switch accepts the next VTP advertisement that uses the same password and domain name in the advertisement.

If you are adding a new switch to an existing network with VTP capability, the new switch learns the domain name only after the applicable password has been configured on it.

> **Caution** When you configure a VTP domain password, the management domain does not function properly if you do not assign a management domain password to each switch in the domain.

Related Topics

Configuring a VTP Version 3 Password , on page 528

Example: Configuring a Switch as the Primary Server, on page 538

VTP Version

Follow these guidelines when deciding which VTP version to implement:

- All switches in a VTP domain must have the same domain name, but they do not need to run the same VTP version.
- A VTP version 2-capable switch can operate in the same VTP domain as a switch running VTP version 1 if version 2 is disabled on the version 2-capable switch (version 2 is disabled by default).
- If a switch running VTP version 1, but capable of running VTP version 2, receives VTP version 3 advertisements, it automatically moves to VTP version 2.
- If a switch running VTP version 3 is connected to a switch running VTP version 1, the VTP version 1 switch moves to VTP version 2, and the VTP version 3 switch sends scaled-down versions of the VTP packets so that the VTP version 2 switch can update its database.
- A switch running VTP version 3 cannot move to version 1 or 2 if it has extended VLANs.
- Do not enable VTP version 2 on a switch unless all of the switches in the same VTP domain are version-2-capable. When you enable version 2 on a switch, all of the version-2-capable switches in the

domain enable version 2. If there is a version 1-only switch, it does not exchange VTP information with switches that have version 2 enabled.

- Cisco recommends placing VTP version 1 and 2 switches at the edge of the network because they do not forward VTP version 3 advertisements.
- If there are TrBRF and TrCRF Token Ring networks in your environment, you must enable VTP version 2 or version 3 for Token Ring VLAN switching to function properly. To run Token Ring and Token Ring-Net, disable VTP version 2.
- VTP version 1 and version 2 do not propagate configuration information for extended range VLANs (VLANs 1006 to 4094). You must configure these VLANs manually on each device. VTP version 3 supports extended-range VLANs and support for extended range VLAN database propagation. You cannot convert from VTP version 3 to VTP version 2 if extended VLANs are configured.
- When a VTP version 3 device trunk port receives messages from a VTP version 2 device, it sends a scaled-down version of the VLAN database on that particular trunk in VTP version 2 format. A VTP version 3 device does not send VTP version 2-formatted packets on a trunk unless it first receives VTP version 2 packets on that trunk port.
- When a VTP version 3 device detects a VTP version 2 device on a trunk port, it continues to send VTP version 3 packets, in addition to VTP version 2 packets, to allow both kinds of neighbors to coexist on the same trunk.
- A VTP version 3 device does not accept configuration information from a VTP version 2 or version 1 device.
- Two VTP version 3 regions can only communicate in transparent mode over a VTP version 1 or version 2 region.
- Devices that are only VTP version 1 capable cannot interoperate with VTP version 3 devices.
- For VTP version 1 and version 2, if extended-range VLANs are configured on the switch stack, you cannot change VTP mode to client or server. You receive an error message, and the configuration is not allowed. VTP version 1 and version 2 do not propagate configuration information for extended range VLANs (VLANs 1006 to 4094). You must manually configure these VLANs on each device.

Note For VTP version 1 and 2, before you create extended-range VLANs (VLAN IDs 1006 to 4094), you must set VTP mode to transparent by using the **vtp mode transparent** global configuration command. Save this configuration to the startup configuration so that the switch starts in VTP transparent mode. Otherwise, you lose the extended-range VLAN configuration if the switch resets and boots up in VTP server mode (the default).

- VTP version 3 supports extended-range VLANs. If extended VLANs are configured, you cannot convert from VTP version 3 to VTP version 2.
- If you configure the switch for VTP client mode, the switch does not create the VLAN database file (vlan.dat). If the switch is then powered off, it resets the VTP configuration to the default. To keep the VTP configuration with VTP client mode after the switch restarts, you must first configure the VTP domain name before the VTP mode.

> **Caution** If all switches are operating in VTP client mode, do not configure a VTP domain name. If you do, it is impossible to make changes to the VLAN configuration of that domain. Therefore, make sure you configure at least one switch as a VTP server.

Related Topics

Enabling the VTP Version , on page 530

How to Configure VTP

Configuring VTP Mode

You can configure VTP mode as one of these:

- VTP server mode—In VTP server mode, you can change the VLAN configuration and have it propagated throughout the network.
- VTP client mode—In VTP client mode, you cannot change its VLAN configuration. The client switch receives VTP updates from a VTP server in the VTP domain and then modifies its configuration accordingly.
- VTP transparent mode—In VTP transparent mode, VTP is disabled on the switch. The switch does not send VTP updates and does not act on VTP updates received from other switch. However, a VTP transparent switch running VTP version 2 does forward received VTP advertisements on its trunk links.
- VTP off mode—VTP off mode is the same as VTP transparent mode except that VTP advertisements are not forwarded.

When you configure a domain name, it cannot be removed; you can only reassign a switch to a different domain.

SUMMARY STEPS

1. **configure terminal**
2. **vtp domain** *domain-name*
3. **vtp mode** {**client** | **server** | **transparent** | **off**} {**vlan** | **mst** | **unknown**}
4. **vtp password** *password*
5. **end**
6. **show vtp status**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**vtp domain** *domain-name* **Example:** `Switch(config)# vtp domain eng_group`	Configures the VTP administrative-domain name. The name can be 1 to 32 characters. All switches operating in VTP server or client mode under the same administrative responsibility must be configured with the same domain name. This command is optional for modes other than server mode. VTP server mode requires a domain name. If the switch has a trunk connection to a VTP domain, the switch learns the domain name from the VTP server in the domain. You should configure the VTP domain before configuring other VTP parameters. **Note**
Step 3	**vtp mode** {**client** | **server** | **transparent** | **off**} {**vlan** | **mst** | **unknown**} **Example:** `Switch(config)# vtp mode server`	Configures the switch for VTP mode (client, server, transparent, or off). • **vlan—**The VLAN database is the default if none are configured. • **mst—**The multiple spanning tree (MST) database. • **unknown—**An unknown database type.
Step 4	**vtp password** *password* **Example:** `Switch(config)# vtp password mypassword`	(Optional) Sets the password for the VTP domain. The password can be 8 to 64 characters. If you configure a VTP password, the VTP domain does not function properly if you do not assign the same password to each switch in the domain.
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 6	**show vtp status** **Example:** `Switch# show vtp status`	Verifies your entries in the *VTP Operating Mode* and the *VTP Domain Name* fields of the display.
Step 7	**copy running-config startup-config**	(Optional) Saves the configuration in the startup configuration file.

	Command or Action	Purpose
	Example: `Switch# copy running-config startup-config`	Only VTP mode and domain name are saved in the switch running configuration and can be copied to the startup configuration file.

Related Topics

VTP Modes, on page 518

Configuring a VTP Version 3 Password

You can configure a VTP version 3 password on the switch.

SUMMARY STEPS

1. **configure terminal**
2. **vtp password** *password* [**hidden** | **secret**]
3. **end**
4. **show vtp password**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**vtp password** *password* [**hidden** | **secret**] **Example:** `Switch(config)# vtp password mypassword hidden`	(Optional) Sets the password for the VTP domain. The password can be 8 to 64 characters. • (Optional) **hidden—**Saves the secret key generated from the password string in the nvram:vlan.dat file. If you configure a takeover by configuring a VTP primary server, you are prompted to reenter the password. • (Optional) **secret—**Directly configures the password. The secret password must contain 32 hexadecimal characters.

	Command or Action	Purpose
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show vtp password** **Example:** `Switch# show vtp password`	Verifies your entries. The output appears like this: VTP password: 89914640C8D90868B6A0D8103847A733
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves the configuration in the startup configuration file.

Related Topics

Passwords for the VTP Domain, on page 524
Example: Configuring a Switch as the Primary Server, on page 538

Configuring a VTP Version 3 Primary Server

When you configure a VTP server as a VTP primary server, the takeover operation starts.

SUMMARY STEPS

1. **vtp primary** [**vlan** | **mst**] [**force**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**vtp primary** [**vlan** | **mst**] [**force**] **Example:** `Switch# vtp primary vlan force`	Changes the operational state of a switch from a secondary server (the default) to a primary server and advertises the configuration to the domain. If the switch password is configured as **hidden**, you are prompted to reenter the password. • (Optional) **vlan**—Selects the VLAN database as the takeover feature. This is the default. • (Optional) **mst**—Selects the multiple spanning tree (MST) database as the takeover feature.

	Command or Action	Purpose
		• (Optional) **force**—Overwrites the configuration of any conflicting servers. If you do not enter **force**, you are prompted for confirmation before the takeover.

Related Topics

VTP Settings, on page 523

Enabling the VTP Version

VTP version 2 and version 3 are disabled by default.

- When you enable VTP version 2 on a switch , every VTP version 2-capable switch in the VTP domain enables version 2. To enable VTP version 3, you must manually configure it on each switch .
- With VTP versions 1 and 2, you can configure the version only on switches in VTP server or transparent mode. If a switch is running VTP version 3, you can change to version 2 when the switch is in client mode if no extended VLANs exist, and no hidden password was configured.

Caution VTP version 1 and VTP version 2 are not interoperable on switches in the same VTP domain. Do not enable VTP version 2 unless every switch in the VTP domain supports version 2.

- In TrCRF and TrBRF Token Ring environments, you must enable VTP version 2 or VTP version 3 for Token Ring VLAN switching to function properly. For Token Ring and Token Ring-Net media, disable VTP version 2.

Caution In VTP version 3, both the primary and secondary servers can exist on an instance in the domain.

SUMMARY STEPS

1. **configure terminal**
2. **vtp version** {**1** | **2** | **3**}
3. **end**
4. **show vtp status**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**vtp version** {**1** | **2** | **3**} **Example:** `Switch(config)# vtp version 2`	Enables the VTP version on the switch. The default is VTP version 1.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show vtp status** **Example:** `Switch# show vtp status`	Verifies that the configured VTP version is enabled.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves the configuration in the startup configuration file.

Related Topics

VTP Version, on page 524
VTP Version 2, on page 519
VTP Version 3, on page 520

Enabling VTP Pruning

Pruning increases available bandwidth by restricting flooded traffic to those trunk links that the traffic must use to access the destination devices. You can only enable VTP pruning on a switch in VTP server mode.

With VTP versions 1 and 2, when you enable pruning on the VTP server, it is enabled for the entire VTP domain. In VTP version 3, you must manually enable pruning on each switch in the domain.

Only VLANs included in the pruning-eligible list can be pruned. By default, VLANs 2 through 1001 are pruning-eligible on trunk ports. Reserved VLANs and extended-range VLANs cannot be pruned.

Before You Begin

VTP pruning is not designed to function in VTP transparent mode. If one or more switches in the network are in VTP transparent mode, you should do one of these actions:

- Turn off VTP pruning in the entire network.
- Turn off VTP pruning by making all VLANs on the trunk of the switch upstream to the VTP transparent switch pruning ineligible.

To configure VTP pruning on an interface, use the **switchport trunk pruning vlan** interface configuration command. VTP pruning operates when an interface is trunking. You can set VLAN pruning-eligibility, whether or not VTP pruning is enabled for the VTP domain, whether or not any given VLAN exists, and whether or not the interface is currently trunking.

SUMMARY STEPS

1. **configure terminal**
2. **vtp pruning**
3. **end**
4. **show vtp status**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**vtp pruning** **Example:** `Switch(config)# vtp pruning`	Enables pruning in the VTP administrative domain. By default, pruning is disabled. You need to enable pruning on only one switch in VTP server mode.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 4	**show vtp status** **Example:** `Switch# show vtp status`	Verifies your entries in the *VTP Pruning Mode* field of the display.

Related Topics

VTP Pruning, on page 521

Configuring VTP on a Per-Port Basis

With VTP version 3, you can enable or disable VTP on a per-port basis. You can enable VTP only on ports that are in trunk mode. Incoming and outgoing VTP traffic are blocked, not forwarded.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **vtp**
4. **end**
5. **show running-config interface** *interface-id*
6. **show vtp status**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Identifies an interface, and enters interface configuration mode.

	Command or Action	Purpose
Step 3	**vtp** **Example:** `Switch(config)# vtp`	Enables VTP on the specified port.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config interface** *interface-id* **Example:** `Switch# show running-config interface gigabitethernet1/0/1`	Verifies the change to the port.
Step 6	**show vtp status** **Example:** `Switch# show vtp status`	Verifies the configuration.

Related Topics

VTP Settings, on page 523

Adding a VTP Client Switch to a VTP Domain

Follow these steps to verify and reset the VTP configuration revision number on a switch *before* adding it to a VTP domain.

Before You Begin

Before adding a VTP client to a VTP domain, always verify that its VTP configuration revision number is *lower* than the configuration revision number of the other switches in the VTP domain. Switches in a VTP domain always use the VLAN configuration of the switch with the highest VTP configuration revision number. With VTP versions 1 and 2, adding a switch that has a revision number higher than the revision number in the VTP domain can erase all VLAN information from the VTP server and VTP domain. With VTP version 3, the VLAN information is not erased.

You can use the **vtp mode transparent** global configuration command to disable VTP on the switch and then to change its VLAN information without affecting the other switches in the VTP domain.

SUMMARY STEPS

1. **show vtp status**
2. **configure terminal**
3. **vtp domain** *domain-name*
4. **end**
5. **show vtp status**
6. **configure terminal**
7. **vtp domain** *domain-name*
8. **end**
9. **show vtp status**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show vtp status** **Example:** `Switch# show vtp status`	Checks the VTP configuration revision number. If the number is 0, add the switch to the VTP domain. If the number is greater than 0, follow these substeps: • Write down the domain name. • Write down the configuration revision number. • Continue with the next steps to reset the switch configuration revision number.
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 3	**vtp domain** *domain-name* **Example:** `Switch(config)# vtp domain domain123`	Changes the domain name from the original one displayed in Step 1 to a new name.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. The VLAN information on the switch is updated and the configuration revision number is reset to 0.

	Command or Action	Purpose
Step 5	**show vtp status** **Example:** `Switch# show vtp status`	Verifies that the configuration revision number has been reset to 0.
Step 6	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 7	**vtp domain** *domain-name* **Example:** `Switch(config)# vtp domain domain012`	Enters the original domain name on the switch
Step 8	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. The VLAN information on the switch is updated.
Step 9	**show vtp status** **Example:** `Switch# show vtp status`	(Optional) Verifies that the domain name is the same as in Step 1 and that the configuration revision number is 0.

Related Topics

VTP Domain, on page 517
Prerequisites for VTP, on page 515
Domain Names for Configuring VTP, on page 523

Monitoring VTP

This section describes commands used to display and monitor the VTP configuration.

You monitor VTP by displaying VTP configuration information: the domain name, the current VTP revision, and the number of VLANs. You can also display statistics about the advertisements sent and received by the switch.

Table 68: VTP Monitoring Commands

Command	Purpose
show vtp counters	Displays counters about VTP messages that have been sent and received.
show vtp devices [**conflict**]	Displays information about all VTP version 3 devices in the domain. Conflicts are VTP version 3 devices with conflicting primary servers. The **show vtp devices** command does not display information when the switch is in transparent or off mode.
show vtp interface [*interface-id*]	Displays VTP status and configuration for all interfaces or the specified interface.
show vtp password	Displays the VTP password. The form of the password displayed depends on whether or not the **hidden** keyword was entered and if encryption is enabled on the switch.
show vtp status	Displays the VTP switch configuration information.

Troubleshooting VTP

The following commands can be used to troubleshoot VTP on the switch.

Table 69: VTP Troubleshooting Commands

Command	Description
debug platform vlan [**error** [**switch** *number*] | **event** [**switch** *number*] | **switch** *number*]	Debugs platform related VLAN issues.
debug switchport {**backup** { **all** | **errors** | **events** | **vlan-load-balancing** } }	Debugs the switchport configuration. The following debug switchport options are available: • all—All switch backup interface debugging messages. • errors—Switch backup interface errors or exceptions. • events—Switch backup interface events. • vlan-load-balancing—Switch backup interface VLAN load balancing.

Command	Description
debug sw-vlan { **badpmcookies** | **cfg-vlan** | **events** | **ifs** | **management** | **mapping** | **notification** | **packets** | **redundancy** | **registries** | **vtp** }	Debugs the VLAN manager. The following command options are available: • badpmcookies—Debugs VLAN manager incidents of bad Port Manager (pm) cookies. • cfg-vlan—Debugs the VLAN configuration. • events—Debugs VLAN manager events. • ifs—Debugs VLAN manager ifs error tests. • management—Debugs VLAN manager management of internal VLANs. • mapping—Debugs VLAN mapping. • notification—Debugs VLAN manager notifications. • packets—Debugs VLAN manager packets. • redundancy—Debugs VTP VLAN redundancy. • registries—Debugs VLAN manager registries. • vtp—VTP protocol debugging.
debug vlan { **configuration** *vlan* }	Debugs the VLAN. Specify the VLAN or VLAN range for debugging.

Configuration Examples for VTP

Example: Configuring a Switch as the Primary Server

This example shows how to configure a switch as the primary server for the VLAN database (the default) when a hidden or secret password was configured:

```
Switch# vtp primary vlan
Enter VTP password: mypassword
This switch is becoming Primary server for vlan feature in the VTP  domain

VTP Database Conf Switch ID      Primary Server Revision System Name
------------ ---- -------------- -------------- -------- --------------------
VLANDB       Yes  00d0.00b8.1400=00d0.00b8.1400 1        stp7

Do you want to continue (y/n)  [n]? y
```

Related Topics

Configuring a VTP Version 3 Password , on page 528

Passwords for the VTP Domain, on page 524

Where to Go Next

After configuring VTP, you can configure the following:

- VLANs
- VLAN groups
- VLAN trunking
- VLAN Membership Policy Server (VMPS)
- Voice VLANs
- Private VLANs

Additional References

Related Documents

Related Topic	Document Title
CLI commands	*Layer 2/3 Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)* *VLAN Command Reference (Catalyst 3850 Switches)*
Configuration Commands	*LAN Switching Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)* *Layer 2/3 Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*

Standards and RFCs

Standard/RFC	Title
RFC 1573	Evolution of the Interfaces Group of MIB-II
RFC 1757	Remote Network Monitoring Management
RFC 2021	SNMPv2 Management Information Base for the Transmission Control Protocol using SMIv2

MIBs

MIB	MIBs Link
All supported MIBs for this release. • BRIDGE-MIB (RFC1493) • CISCO-BRIDGE-EXT-MIB • CISCO-CDP-MIB • CISCO-PAGP-MIB • CISCO-PRIVATE-VLAN-MIB • CISCO-LAG-MIB • CISCO-L2L3-INTERFACE-CONFIG-MIB • CISCO-MAC-NOTIFICATION-MIB • CISCO-STP-EXTENSIONS-MIB • CISCO-VLAN-IFTABLE-RELATIONSHIP-MIB • CISCO-VLAN-MEMBERSHIP-MIB • CISCO-VTP-MIB • IEEE8023-LAG-MIB • IF-MIB (RFC 1573) • RMON-MIB (RFC 1757) • RMON2-MIB (RFC 2021)	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for VTP

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 33

Configuring VLANs

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for VLANs

The following are prerequisites and considerations for configuring VLANs:

- Before you create VLANs, you must decide whether to use VLAN Trunking Protocol (VTP) to maintain global VLAN configuration for your network.

- If you plan to configure many VLANs on the switch and to not enable routing, you can set the Switch Database Management (SDM) feature to the VLAN template, which configures system resources to support the maximum number of unicast MAC addresses.
- Switches running the LAN Base feature set support only static routing on SVIs.
- A VLAN should be present in the switch to be able to add it to the VLAN group.

Restrictions for VLANs

The following are restrictions for VLANs:

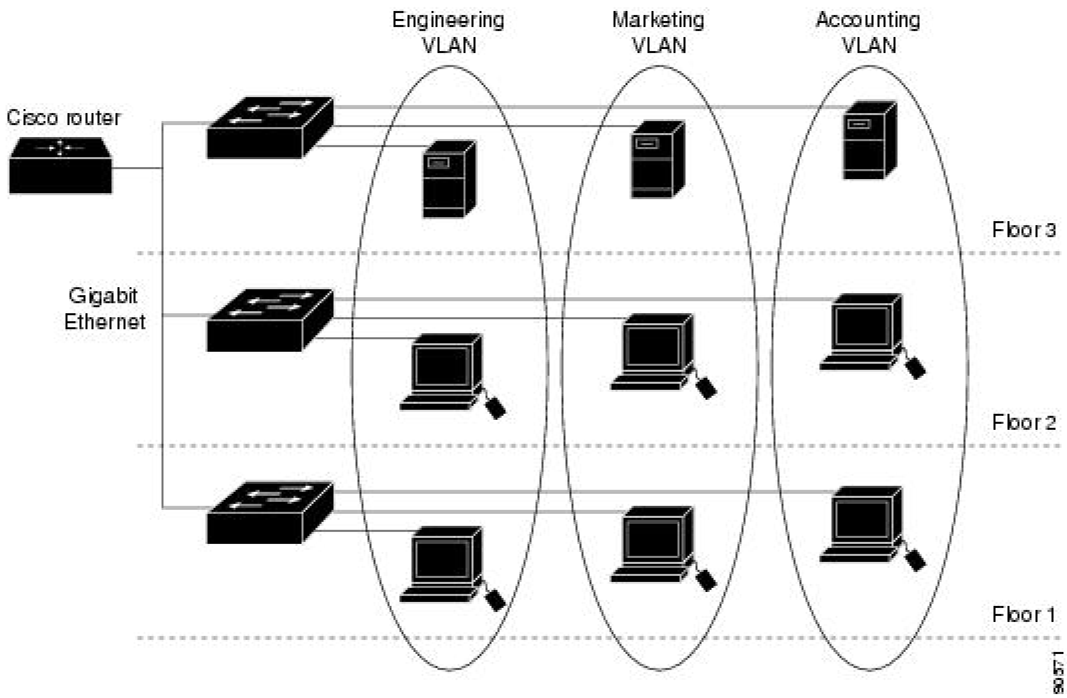
- The switch supports per-VLAN spanning-tree plus (PVST+) or rapid PVST+ with a maximum of 128 spanning-tree instances. One spanning-tree instance is allowed per VLAN.
- The switch supports both Inter-Switch Link (ISL) and IEEE 802.1Q trunking methods for sending VLAN traffic over Ethernet ports.
- Configuring an interface VLAN router's MAC address is not supported. The interface VLAN already has an MAC address assigned by default.
- Private VLANs are not supported on the switch.
- You cannot have a switch stack containing a mix of Catalyst 3850 and Catalyst 3650 switches.

Information About VLANs

Logical Networks

A VLAN is a switched network that is logically segmented by function, project team, or application, without regard to the physical locations of the users. VLANs have the same attributes as physical LANs, but you can group end stations even if they are not physically located on the same LAN segment. Any switch port can belong to a VLAN, and unicast, broadcast, and multicast packets are forwarded and flooded only to end stations in the VLAN. Each VLAN is considered a logical network, and packets destined for stations that do not belong to the VLAN must be forwarded through a router or a switch supporting fallback bridging. In a switch stack, VLANs can be formed with ports across the stack. Because a VLAN is considered a separate logical network, it contains its own bridge Management Information Base (MIB) information and can support its own implementation of spanning tree.

Figure 14: VLANs as Logically Defined Networks



VLANs are often associated with IP subnetworks. For example, all the end stations in a particular IP subnet belong to the same VLAN. Interface VLAN membership on the switch is assigned manually on an interface-by-interface basis. When you assign switch interfaces to VLANs by using this method, it is known as interface-based, or static, VLAN membership.

Traffic between VLANs must be routed.

The switch can route traffic between VLANs by using switch virtual interfaces (SVIs). An SVI must be explicitly configured and assigned an IP address to route traffic between VLANs.

Supported VLANs

The switch supports VLANs in VTP client, server, and transparent modes. VLANs are identified by a number from 1 to 4094. VLAN 1 is the default VLAN and is created during system initialization. VLAN IDs 1002 through 1005 are reserved for Token Ring and FDDI VLANs. All of the VLANs except 1002 to 1005 are available for user configuration.

There are 3 VTP versions. VTP version 1 and version 2 support only normal-range VLANs (VLAN IDs 1 to 1005). In these versions, the switch must be in VTP transparent mode when you create VLAN IDs from 1006 to 4094. VTP version 3 supports the entire VLAN range (VLANs 1 to 4094). Extended range VLANs (VLANs 1006 to 4094) are supported only in VTP version 3. You cannot convert from VTP version 3 to VTP version 2 if extended VLANs are configured in the domain.

There are 3 VTP versions: VTP version 1, version 2, and version 3. All VTP versions support both normal and extended range VLANs, but only with VTP version 3, does the switch propagate extended range VLAN configuration information. When extended range VLANs are created in VTP versions 1 and 2, their configuration information is not propagated. Even the local VTP database entries on the switch are not updated, but the extended range VLANs configuration information is created and stored in the running configuration file.

You can configure up to 4094 VLANs on the switch.

Related Topics

Creating or Modifying an Ethernet VLAN , on page 553

Deleting a VLAN , on page 556

Assigning Static-Access Ports to a VLAN, on page 559
Monitoring VLANs, on page 568

Creating an Extended-Range VLAN , on page 562

Creating an Extended-Range VLAN with an Internal VLAN ID, on page 565

Monitoring VLANs, on page 568

Creating or Modifying an Ethernet VLAN , on page 553

Deleting a VLAN , on page 556

Assigning Static-Access Ports to a VLAN, on page 559
Monitoring VLANs, on page 568

Creating an Extended-Range VLAN , on page 562

Creating an Extended-Range VLAN with an Internal VLAN ID, on page 565

Monitoring VLANs, on page 568

Creating or Modifying an Ethernet VLAN , on page 553

Deleting a VLAN , on page 556

Assigning Static-Access Ports to a VLAN, on page 559
Monitoring VLANs, on page 568

Creating an Extended-Range VLAN , on page 562

Creating an Extended-Range VLAN with an Internal VLAN ID, on page 565

Monitoring VLANs, on page 568

Creating or Modifying an Ethernet VLAN , on page 553

Deleting a VLAN , on page 556

Assigning Static-Access Ports to a VLAN, on page 559
Monitoring VLANs, on page 568

Creating an Extended-Range VLAN , on page 562

Creating an Extended-Range VLAN with an Internal VLAN ID, on page 565

Monitoring VLANs, on page 568

Creating or Modifying an Ethernet VLAN , on page 553

Deleting a VLAN , on page 556

Assigning Static-Access Ports to a VLAN, on page 559
Monitoring VLANs, on page 568

Creating an Extended-Range VLAN , on page 562

Creating an Extended-Range VLAN with an Internal VLAN ID, on page 565

Monitoring VLANs, on page 568

Creating or Modifying an Ethernet VLAN , on page 553

Deleting a VLAN , on page 556

Assigning Static-Access Ports to a VLAN, on page 559
Monitoring VLANs, on page 568

Creating an Extended-Range VLAN , on page 562

Creating an Extended-Range VLAN with an Internal VLAN ID, on page 565

Monitoring VLANs, on page 568

Creating or Modifying an Ethernet VLAN , on page 553

Deleting a VLAN , on page 556

Assigning Static-Access Ports to a VLAN, on page 559
Monitoring VLANs, on page 568

Creating an Extended-Range VLAN , on page 562

Creating an Extended-Range VLAN with an Internal VLAN ID, on page 565

Monitoring VLANs, on page 568

VLAN Port Membership Modes

You configure a port to belong to a VLAN by assigning a membership mode that specifies the kind of traffic the port carries and the number of VLANs to which it can belong.

When a port belongs to a VLAN, the switch learns and manages the addresses associated with the port on a per-VLAN basis.

Table 70: Port Membership Modes and Characteristics

Membership Mode	VLAN Membership Characteristics	VTP Characteristics
Static-access	A static-access port can belong to one VLAN and is manually assigned to that VLAN.	VTP is not required. If you do not want VTP to globally propagate information, set the VTP mode to transparent. To participate in VTP, there must be at least one trunk port on the switch or the switch stack connected to a trunk port of a second switch or switch stack.

Membership Mode	VLAN Membership Characteristics	VTP Characteristics
Trunk (IEEE 802.1Q) : • IEEE 802.1Q—Industry-standard trunking encapsulation.	A trunk port is a member of all VLANs by default, including extended-range VLANs, but membership can be limited by configuring the allowed-VLAN list. You can also modify the pruning-eligible list to block flooded traffic to VLANs on trunk ports that are included in the list.	VTP is recommended but not required. VTP maintains VLAN configuration consistency by managing the addition, deletion, and renaming of VLANs on a network-wide basis. VTP exchanges VLAN configuration messages with other switches over trunk links.
Dynamic access **Note** Dynamic-access ports and VMPS is not supported on the switch.	A dynamic-access port can belong to one VLAN (VLAN ID 1 to 4094) and is dynamically assigned by a VLAN Member Policy Server (VMPS). You can have dynamic-access ports and trunk ports on the same switch, but you must connect the dynamic-access port to an end station or hub and not to another switch.	VTP is required. Configure the VMPS and the client with the same VTP domain name. To participate in VTP, at least one trunk port on the switch or a switch stack must be connected to a trunk port of a second switch or switch stack.
Voice VLAN	A voice VLAN port is an access port attached to a Cisco IP Phone, configured to use one VLAN for voice traffic and another VLAN for data traffic from a device attached to the phone.	VTP is not required; it has no effect on a voice VLAN.

Related Topics

Assigning Static-Access Ports to a VLAN, on page 559
Monitoring VLANs, on page 568

VLAN Configuration Files

Configurations for VLAN IDs 1 to 1005 are written to the vlan.dat file (VLAN database), and you can display them by entering the **show vlan** privileged EXEC command. The vlan.dat file is stored in flash memory. If the VTP mode is transparent, they are also saved in the switch running configuration file.

In a switch stack, the whole stack uses the same vlan.dat file and running configuration. On some switches, the vlan.dat file is stored in flash memory on the active switch.

You use the interface configuration mode to define the port membership mode and to add and remove ports from VLANs. The results of these commands are written to the running-configuration file, and you can display the file by entering the **show running-config** privileged EXEC command.

When you save VLAN and VTP information (including extended-range VLAN configuration information) in the startup configuration file and reboot the switch, the switch configuration is selected as follows:

- If the VTP mode is transparent in the startup configuration, and the VLAN database and the VTP domain name from the VLAN database matches that in the startup configuration file, the VLAN database is ignored (cleared), and the VTP and VLAN configurations in the startup configuration file are used. The VLAN database revision number remains unchanged in the VLAN database.
- If the VTP mode or domain name in the startup configuration does not match the VLAN database, the domain name and VTP mode and configuration for the VLAN IDs 1 to 1005 use the VLAN database information.
- In VTP versions 1 and 2, if VTP mode is server, the domain name and VLAN configuration for VLAN IDs 1 to 1005 use the VLAN database information. VTP version 3 also supports VLANs 1006 to 4094.

Normal-Range VLAN Configuration Guidelines

Normal-range VLANs are VLANs with IDs from 1 to 1005.

Follow these guidelines when creating and modifying normal-range VLANs in your network:

- Normal-range VLANs are identified with a number between 1 and 1001. VLAN numbers 1002 through 1005 are reserved for Token Ring and FDDI VLANs.
- VLAN configurations for VLANs 1 to 1005 are always saved in the VLAN database. If the VTP mode is transparent, VTP and VLAN configurations are also saved in the switch running configuration file.
- If the switch is in VTP server or VTP transparent mode, you can add, modify or remove configurations for VLANs 2 to 1001 in the VLAN database. (VLAN IDs 1 and 1002 to 1005 are automatically created and cannot be removed.)
- Extended-range VLANs created in VTP transparent mode are not saved in the VLAN database and are not propagated. VTP version 3 supports extended range VLAN (VLANs 1006 to 4094) database propagation in VTP server mode.
- Before you can create a VLAN, the switch must be in VTP server mode or VTP transparent mode. If the switch is a VTP server, you must define a VTP domain or VTP will not function.
- The switch does not support Token Ring or FDDI media. The switch does not forward FDDI, FDDI-Net, TrCRF, or TrBRF traffic, but it does propagate the VLAN configuration through VTP.
- The switch supports 128 spanning tree instances. If a switch has more active VLANs than supported spanning-tree instances, spanning tree can be enabled on 128 VLANs and is disabled on the remaining VLANs. If you have already used all available spanning-tree instances on a switch, adding another VLAN anywhere in the VTP domain creates a VLAN on that switch that is not running spanning-tree. If you have the default allowed list on the trunk ports of that switch (which is to allow all VLANs), the new VLAN is carried on all trunk ports. Depending on the topology of the network, this could create a loop in the new VLAN that would not be broken, particularly if there are several adjacent switches that all have run out of spanning-tree instances. You can prevent this possibility by setting allowed lists on the trunk ports of switches that have used up their allocation of spanning-tree instances.

  If the number of VLANs on the switch exceeds the number of supported spanning-tree instances, we recommend that you configure the IEEE 802.1s Multiple STP (MSTP) on your switch to map multiple VLANs to a single spanning-tree instance.

- When a switch in a stack learns a new VLAN or deletes or modifies an existing VLAN (either through VTP over network ports or through the CLI), the VLAN information is communicated to all stack members.
- When a switch joins a stack or when stacks merge, VTP information (the vlan.dat file) on the new switches will be consistent with the active switch.

Related Topics

Creating or Modifying an Ethernet VLAN , on page 553
Deleting a VLAN , on page 556
Assigning Static-Access Ports to a VLAN, on page 559
Monitoring VLANs, on page 568
Creating or Modifying an Ethernet VLAN , on page 553
Deleting a VLAN , on page 556
Assigning Static-Access Ports to a VLAN, on page 559
Monitoring VLANs, on page 568
Creating or Modifying an Ethernet VLAN , on page 553
Deleting a VLAN , on page 556
Assigning Static-Access Ports to a VLAN, on page 559
Monitoring VLANs, on page 568
Creating or Modifying an Ethernet VLAN , on page 553
Monitoring VLANs, on page 568
Creating or Modifying an Ethernet VLAN , on page 553
Deleting a VLAN , on page 556
Assigning Static-Access Ports to a VLAN, on page 559
Monitoring VLANs, on page 568
Creating or Modifying an Ethernet VLAN , on page 553
Deleting a VLAN , on page 556
Assigning Static-Access Ports to a VLAN, on page 559
Monitoring VLANs, on page 568
Creating or Modifying an Ethernet VLAN , on page 553
Deleting a VLAN , on page 556
Assigning Static-Access Ports to a VLAN, on page 559
Monitoring VLANs, on page 568
Creating or Modifying an Ethernet VLAN , on page 553
Deleting a VLAN , on page 556
Assigning Static-Access Ports to a VLAN, on page 559
Monitoring VLANs, on page 568

Extended-Range VLAN Configuration Guidelines

Extended-range VLANs are VLANs with IDs from 1006 to 4094.

Follow these guidelines when creating extended-range VLANs:

- VLAN IDs in the extended range are not saved in the VLAN database and are not recognized by VTP unless the switch is running VTP version 3.
- You cannot include extended-range VLANs in the pruning eligible range.
- In VTP version 1 and 2, a switch must be in VTP transparent mode when you create extended-range VLANs. If VTP mode is server or client, an error message is generated, and the extended-range VLAN is rejected. VTP version 3 supports extended VLANs in server and transparent modes.
- For VTP version 1 or 2, you can set the VTP mode to transparent in global configuration mode. You should save this configuration to the startup configuration so that the switch boots up in VTP transparent mode. Otherwise, you lose the extended-range VLAN configuration if the switch resets. If you create extended-range VLANs in VTP version 3, you cannot convert to VTP version 1 or 2.
- . When the maximum number of spanning-tree instances are on the switch, spanning tree is disabled on any newly created VLANs. If the number of VLANs on the switch exceeds the maximum number of spanning-tree instances, we recommend that you configure the IEEE 802.1s Multiple STP (MSTP) on your switch to map multiple VLANs to a single spanning-tree instance.
- In a switch stack, the whole stack uses the same running configuration and saved configuration, and extended-range VLAN information is shared across the stack.

Related Topics

Creating an Extended-Range VLAN , on page 562
Creating an Extended-Range VLAN with an Internal VLAN ID, on page 565
Monitoring VLANs, on page 568
Creating an Extended-Range VLAN , on page 562
Creating an Extended-Range VLAN with an Internal VLAN ID, on page 565
Monitoring VLANs, on page 568
Creating an Extended-Range VLAN , on page 562
Creating an Extended-Range VLAN with an Internal VLAN ID, on page 565
Monitoring VLANs, on page 568
Creating an Extended-Range VLAN , on page 562
Monitoring VLANs, on page 568
Creating an Extended-Range VLAN , on page 562
Creating an Extended-Range VLAN with an Internal VLAN ID, on page 565
Monitoring VLANs, on page 568
Creating an Extended-Range VLAN , on page 562
Creating an Extended-Range VLAN with an Internal VLAN ID, on page 565
Monitoring VLANs, on page 568

Creating an Extended-Range VLAN , on page 562

Creating an Extended-Range VLAN with an Internal VLAN ID, on page 565

Monitoring VLANs, on page 568

Creating an Extended-Range VLAN , on page 562

Creating an Extended-Range VLAN with an Internal VLAN ID, on page 565

Monitoring VLANs, on page 568

Information About VLAN Groups

Whenever a wireless client connects to a wireless network (WLAN), the client is placed in a VLAN that is associated with the WLAN. In a large venue such as an auditorium, a stadium, or a conference room where there are numerous wireless clients, having only a single WLAN to accommodate many clients might be a challenge.

The VLAN group feature uses a single WLAN that can support multiple VLANs. The clients can get assigned to one of the configured VLANs. This feature maps a WLAN to a single VLAN or multiple VLANs using the VLAN groups. When a wireless client associates to the WLAN, the VLAN is derived by an algorithm based on the MAC address of the wireless client. A VLAN is assigned to the client and the client gets the IP address from the assigned VLAN. This feature also extends the current AP group architecture and AAA override architecture, where the AP groups and AAA override can override a VLAN or a VLAN group to which the WLAN is mapped.

Related Topics

Creating VLAN Groups (CLI), on page 558

How to Configure VLANs

How to Configure Normal-Range VLANs

You can set these parameters when you create a new normal-range VLAN or modify an existing VLAN in the VLAN database:

- VLAN ID
- VLAN name
- VLAN type
    - Ethernet
    - Fiber Distributed Data Interface [FDDI]
    - FDDI network entity title [NET]
    - TrBRF or TrCRF
    - Token Ring
    - Token Ring-Net
- VLAN state (active or suspended)

- Maximum transmission unit (MTU) for the VLAN
- Security Association Identifier (SAID)
- Bridge identification number for TrBRF VLANs
- Ring number for FDDI and TrCRF VLANs
- Parent VLAN number for TrCRF VLANs
- Spanning Tree Protocol (STP) type for TrCRF VLANs
- VLAN number to use when translating from one VLAN type to another

You can cause inconsistency in the VLAN database if you attempt to manually delete the vlan.dat file. If you want to modify the VLAN configuration, follow the procedures in this section.

Creating or Modifying an Ethernet VLAN

Before You Begin

With VTP version 1 and 2, if the switch is in VTP transparent mode, you can assign VLAN IDs greater than 1006, but they are not added to the VLAN database.

The switch supports only Ethernet interfaces. Because FDDI and Token Ring VLANs are not locally supported, you only configure FDDI and Token Ring media-specific characteristics for VTP global advertisements to other switches.

Although the switch does not support Token Ring connections, a remote device with Token Ring connections could be managed from one of the supported switches. Switches running VTP Version 2 advertise information about these Token Ring VLANs:

- Token Ring TrBRF VLANs
- Token Ring TrCRF VLANs

SUMMARY STEPS

1. **configure terminal**
2. **vlan** *vlan-id*
3. **name** *vlan-name*
4. **mtu** *mtu-size*
5. **media** { **ethernet** | **fd-net** | **fddi** | **tokenring** | **trn-net** }
6. **remote-span**
7. **end**
8. **show vlan** {**name** *vlan-name* | **id** *vlan-id*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**vlan** *vlan-id* **Example:** `Switch(config)# vlan 20`	Enters a VLAN ID, and enters VLAN configuration mode. Enter a new VLAN ID to create a VLAN, or enter an existing VLAN ID to modify that VLAN. **Note** The available VLAN ID range for this command is 1 to 4094. Additional **vlan** command options include: • **access-map—**Creates VLAN access-maps or enters the vlan access map command mode. • **configuration—**Enters the vlan feature configuration mode. • **dot1q—**Configures VLAN dot1q tag native parameters. • **filter—**Applies a VLAN filter map to a VLAN list. • **group—**Creates a VLAN group.
Step 3	**name** *vlan-name* **Example:** `Switch(config-vlan)# name test20`	(Optional) Enters a name for the VLAN. If no name is entered for the VLAN, the default is to append the *vlan-id* value with leading zeros to the word VLAN. For example, VLAN0004 is a default VLAN name for VLAN 4. The following additional VLAN configuration command options are available: • **are—**Sets the maximum number of All Router Explorer (ARE) hops for the VLAN. • **backupcrf—**Enables or disables the backup concentrator relay function (CRF) mode for the VLAN. • **bridge—**Sets the value of the bridge number for the FDDI net or Token Ring net type VLANs. • **exit—**Applies changes, bumps the revision number, and exits. • **media—**Sets the media type of the VLAN. • **no—**Negates the command or default. • **parent—**Sets the value of the ID for the parent VLAN for FDDI or Token Ring type VLANs. • **remote-span—**Configures a remote SPAN VLAN. • **ring—**Sets the ring number value for FDDI or Token Ring type VLANs. • **said—**Sets the IEEE 802.10 SAID value.

	Command or Action	Purpose
		• **shutdown—**Shuts down the VLAN switching. • **state—**Sets the operational VLAN state to active or suspended. • **ste—**Sets the maximum number of Spanning Tree Explorer (STE) hops for the VLAN. • **stp—**Sets the Spanning Tree characteristics of the VLAN.
Step 4	**mtu** *mtu-size* **Example:** `Switch(config-vlan)# mtu 256`	(Optional) Changes the MTU size (or other VLAN characteristic).
Step 5	**media** { **ethernet** | **fd-net** | **fddi** | **tokenring** | **trn-net** } **Example:** `Switch(config-vlan)# media ethernet`	Configures the VLAN media type. Command options include: • **ethernet—**Sets the VLAN media type as Ethernet. • **fd-net—**Sets the VLAN media type as FDDI net. • **fddi—**Sets the VLAN media type as FDDI. • **tokenring—**Sets the VLAN media type as Token Ring. • **trn-net—**Sets the VLAN media type as Token Ring net.
Step 6	**remote-span** **Example:** `Switch(config-vlan)# remote-span`	(Optional) Configures the VLAN as the RSPAN VLAN for a remote SPAN session. For more information on remote SPAN, see the *Catalyst 3850 Network Management Configuration Guide*.
Step 7	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 8	**show vlan** {**name** *vlan-name* | **id** *vlan-id*} **Example:** `Switch# show vlan name test20 id 20`	Verifies your entries.

Related Topics

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549

Monitoring VLANs, on page 568

Normal-Range VLAN Configuration Guidelines, on page 549

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549

Monitoring VLANs, on page 568

Deleting a VLAN

When you delete a VLAN from a switch that is in VTP server mode, the VLAN is removed from the VLAN database for all switches in the VTP domain. When you delete a VLAN from a switch that is in VTP transparent mode, the VLAN is deleted only on that specific switch or a switch stack.

You cannot delete the default VLANs for the different media types: Ethernet VLAN 1 and FDDI or Token Ring VLANs 1002 to 1005.

Caution When you delete a VLAN, any ports assigned to that VLAN become inactive. They remain associated with the VLAN (and thus inactive) until you assign them to a new VLAN.

SUMMARY STEPS

1. **configure terminal**
2. **no vlan** *vlan-id*
3. **end**
4. **show vlan brief**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**no vlan** *vlan-id* **Example:** `Switch(config)# no vlan 4`	Removes the VLAN by entering the VLAN ID.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show vlan brief** **Example:** `Switch# show vlan brief`	Verifies the VLAN removal.

Related Topics

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549

Monitoring VLANs, on page 568
Monitoring VLANs, on page 568
Supported VLANs, on page 545
Normal-Range VLAN Configuration Guidelines, on page 549
Monitoring VLANs, on page 568
Supported VLANs, on page 545
Normal-Range VLAN Configuration Guidelines, on page 549
Monitoring VLANs, on page 568
Supported VLANs, on page 545
Normal-Range VLAN Configuration Guidelines, on page 549
Monitoring VLANs, on page 568
Supported VLANs, on page 545
Normal-Range VLAN Configuration Guidelines, on page 549
Monitoring VLANs, on page 568

Creating VLAN Groups (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **vlan group** *WORD* **vlan-list** *vlan-ID*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.
Step 2	**vlan group** *WORD* **vlan-list** *vlan-ID* **Example:** `Switch(config)#vlan group vlangrp1 vlan-list 91-95`	Creates a VLAN group with the given group name (vlangrp1) and adds all the VLANs listed in the command. The VLAN list ranges from 1 to 4096 and the recommended number of VLANs in a group is 32.
Step 3	**end** **Example:** `Switch(config)#end`	Exits the global configuration mode and returns to privileged EXEC mode. Alternatively, press **CTRL-Z** to exit the global configuration mode.

Related Topics

Information About VLAN Groups, on page 552

Adding a VLAN Group to WLAN (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *WORD number*
3. **client vlan** *WORD*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.
Step 2	**wlan** *WORD number* **Example:** `Switch(config)#wlan wlanname 512`	Enables the WLAN to map a VLAN group using an identifier. The WLAN identifier values range from 1 to 512.
Step 3	**client vlan** *WORD* **Example:** `Switch(config-wlan)#client vlan vlangrp1`	Maps the VLAN group to the WLAN by entering the VLAN identifier, VLAN group, or the VLAN name.
Step 4	**end** **Example:** `Switch(config-wlan)#end`	Exits the global configuration mode and returns to privileged EXEC mode . Alternatively, press **CTRL-Z** to exit the global configuration mode.

Assigning Static-Access Ports to a VLAN

You can assign a static-access port to a VLAN without having VTP globally propagate VLAN configuration information by disabling VTP (VTP transparent mode).

If you assign an interface to a VLAN that does not exist, the new VLAN is created.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **switchport access vlan** *vlan-id*
5. **end**
6. **show running-config interface** *interface-id*
7. **show interfaces** *interface-id* **switchport**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/1`	Enters the interface to be added to the VLAN.
Step 3	**switchport mode access** **Example:** `Switch(config-if)# switchport mode access`	Defines the VLAN membership mode for the port (Layer 2 access port).
Step 4	**switchport access vlan** *vlan-id* **Example:** `Switch(config-if)# switchport access vlan 2`	Assigns the port to a VLAN. Valid VLAN IDs are 1 to 4094.
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**show running-config interface** *interface-id* **Example:** `Switch# show running-config interface`	Verifies the VLAN membership mode of the interface.

	Command or Action	Purpose
	`gigabitethernet2/0/1`	
Step 7	**show interfaces** *interface-id* **switchport** **Example:** `Switch# show interfaces gigabitethernet2/0/1 switchport`	Verifies your entries in the *Administrative Mode* and the *Access Mode VLAN* fields of the display.

Related Topics

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549

Monitoring VLANs, on page 568

VLAN Port Membership Modes, on page 547

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549

Monitoring VLANs, on page 568

How to Configure Extended-Range VLANs

With VTP version 1 and version 2, when the switch is in VTP transparent mode (VTP disabled), you can create extended-range VLANs (in the range 1006 to 4094). VTP 3 version supports extended-range VLANs in server or transparent move.

Extended-range VLANs enable service providers to extend their infrastructure to a greater number of customers. The extended-range VLAN IDs are allowed for any **switchport** commands that allow VLAN IDs.

With VTP version 1 or 2, extended-range VLAN configurations are not stored in the VLAN database, but because VTP mode is transparent, they are stored in the switch running configuration file, and you can save the configuration in the startup configuration file. Extended-range VLANs created in VTP version 3 are stored in the VLAN database.

You can change only the MTU size and the remote SPAN configuration state on extended-range VLANs; all other characteristics must remain at the default state.

Creating an Extended-Range VLAN

In VTP version 1 or 2, if you enter an extended-range VLAN ID when the switch is not in VTP transparent mode, an error message is generated when you exit VLAN configuration mode, and the extended-range VLAN is not created.

Before you create an extended-range VLAN, you can verify that the VLAN ID is not used internally by entering the **show vlan internal usage** privileged EXEC command.

SUMMARY STEPS

1. **configure terminal**
2. **vtp mode transparent**
3. **vlan** *vlan-id*
4. **remote-span**
5. **mtu** *mtu size*
6. **exit**
7. **interface vlan**
8. **ip mtu** *mtu-size*
9. **end**
10. **show vlan id** *vlan-id*
11. **copy running-config startup config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	**vtp mode transparent** **Example:** `Switch(config)# vtp mode transparent`	Configures the switch for VTP transparent mode, disabling VTP. **Note** This step is not required for VTP version 3.

Step 3	**vlan** *vlan-id*
	
	Example:
	
	`Switch(config)# vlan 2000`
	`Switch(config-vlan)#`
		Enters an extended-range VLAN ID and enters VLAN configuration mode. The range is 1006 to 4094.

Step 4	**remote-span** **Example:** `Switch(config-vlan)# remote-span`	(Optional) Configures the VLAN as the RSPAN VLAN.
Step 5	**mtu** *mtu size* **Example:** `Switch(config-vlan)# mtu 1024`	Modifies the VLAN by changing the MTU size.

Step 6	**exit**
	
	Example:
	
	`Switch(config-vlan)# exit`
	`Switch(config)#`
		Returns to configuration mode.

Step 7	**interface vlan**
	
	Example:
	
	`Switch(config)# interface vlan 200`
	`Switch(config-if)#`
		Enters the interface configuration mode for the selected VLAN.

Step 8	**ip mtu** *mtu-size*
	
	Example:
	
	`Switch(config-if)# ip mtu 1024`
	`Switch(config-if)#`
		(Optional) Modifies the VLAN by changing the MTU size. You can configure the MTU size between 68 to 1500 bytes.
		
		Note Although all VLAN commands appear in the CLI help, only the **ip mtu** *mtu-size*, **private-vlan**, and **remote-span** commands are supported for extended-range VLANs.

Step 9	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 10	**show vlan id** *vlan-id* **Example:** `Switch# show vlan id 2000`	Verifies that the VLAN has been created.
Step 11	**copy running-config startup config** **Example:** `Switch# copy running-config startup-config`	Saves your entries in the switch startup configuration file. To save an extended-range VLAN configuration, you need to save the VTP transparent mode configuration and the extended-range VLAN configuration in the switch startup configuration file. Otherwise, if the switch resets, it will default to VTP server mode, and the extended-range VLAN IDs will not be saved. **Note** This step is not required for VTP version 3 because VLANs are saved in the VLAN database.

Related Topics

Supported VLANs, on page 545

Extended-Range VLAN Configuration Guidelines, on page 551

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Extended-Range VLAN Configuration Guidelines, on page 551

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Extended-Range VLAN Configuration Guidelines, on page 551

Monitoring VLANs, on page 568

Extended-Range VLAN Configuration Guidelines, on page 551

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Extended-Range VLAN Configuration Guidelines, on page 551

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Extended-Range VLAN Configuration Guidelines, on page 551

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Extended-Range VLAN Configuration Guidelines, on page 551

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Extended-Range VLAN Configuration Guidelines, on page 551

Monitoring VLANs, on page 568

Creating an Extended-Range VLAN with an Internal VLAN ID

If you enter an extended-range VLAN ID that is already assigned to an internal VLAN, an error message is generated, and the extended-range VLAN is rejected. To manually free an internal VLAN ID, you must temporarily shut down the routed port that is using the internal VLAN ID.

SUMMARY STEPS

1. **show vlan internal usage**
2. **configure terminal**
3. **interface** *interface-id*
4. **shutdown**
5. **exit**
6. **vtp mode transparent**
7. **vlan** *vlan-id*
8. **exit**
9. **interface** *interface-id*
10. **no shutdown**
11. **end**
12. **copy running-config startup config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show vlan internal usage** **Example:** `Switch# show vlan internal usage`	Displays the VLAN IDs being used internally by the switch. If the VLAN ID that you want to use is an internal VLAN, the display shows the routed port that is using the VLAN ID. Enter that port number in Step 3.
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/3`	Specifies the interface ID for the routed port that is using the VLAN ID, and enters interface configuration mode.

	Command or Action	Purpose
Step 4	**shutdown** **Example:** `Switch(config-if)# shutdown`	Shuts down the port to free the internal VLAN ID.
Step 5	**exit** **Example:** `Switch(config-if)# exit`	Returns to global configuration mode.
Step 6	**vtp mode transparent** **Example:** `Switch(config)# vtp mode transparent`	Sets the VTP mode to transparent for creating extended-range VLANs. **Note** This step is not required for VTP version 3.
Step 7	**vlan** *vlan-id* **Example:** `Switch(config-vlan)# vlan 2000`	Enters the new extended-range VLAN ID, and enters VLAN configuration mode.
Step 8	**exit** **Example:** `Switch(config-vlan)# exit`	Exits from VLAN configuration mode, and returns to global configuration mode.
Step 9	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/3`	Specifies the interface ID for the routed port that you shut down in Step 4, and enters interface configuration mode.
Step 10	**no shutdown** **Example:** `Switch(config)# no shutdown`	Reenables the routed port. It will be assigned a new internal VLAN ID.
Step 11	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 12	**copy running-config startup config** **Example:** `Switch# copy running-config startup-config`	Saves your entries in the switch startup configuration file. To save an extended-range VLAN configuration, you need to save the VTP transparent mode configuration and the extended-range VLAN configuration in the switch startup configuration file. Otherwise, if the switch resets, it will default to VTP server mode, and the extended-range VLAN IDs will not be saved. **Note** This step is not required for VTP version 3 because VLANs are saved in the VLAN database.

Related Topics

Supported VLANs, on page 545

Extended-Range VLAN Configuration Guidelines, on page 551

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Extended-Range VLAN Configuration Guidelines, on page 551

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Extended-Range VLAN Configuration Guidelines, on page 551

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Extended-Range VLAN Configuration Guidelines, on page 551

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Extended-Range VLAN Configuration Guidelines, on page 551

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Extended-Range VLAN Configuration Guidelines, on page 551

Monitoring VLANs, on page 568

Supported VLANs, on page 545

Extended-Range VLAN Configuration Guidelines, on page 551

Monitoring VLANs, on page 568

Monitoring VLANs

Table 71: Privileged EXEC show Commands

Command	Purpose
show interfaces [**vlan** *vlan-id*]	Displays characteristics for all interfaces or for the specified VLAN configured on the switch .
show vlan [**access-map** *name* | **brief** | **dot1q** { **tag native** } | **filter** [**access-map** | **vlan**] | **group** [**group-name** *name*] | **id** *vlan-id* | **ifindex** | **mtu** | **name** *name* | **remote-span** | **summary**]	Displays parameters for all VLANs or the specified VLAN on the switch. The following command options are available: • **access-map—**Displays the VLAN access-maps. • **brief—**Displays VTP VLAN status in brief. • **dot1q—**Displays the dot1q parameters. • **filter—**Displays VLAN filter information. • **group—**Displays the VLAN group with its name and the connected VLANs that are available. • **id—**Displays VTP VLAN status by identification number. • **ifindex—**Displays SNMP ifIndex. • **mtu—**Displays VLAN MTU information. • **name—**Displays the VTP VLAN information by specified name. • **remote-span–**Displays the remote SPAN VLANs. • **summary—**Displays a summary of VLAN information.

Related Topics

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549

Creating or Modifying an Ethernet VLAN , on page 553

Deleting a VLAN , on page 556

Assigning Static-Access Ports to a VLAN, on page 559
Supported VLANs, on page 545

Extended-Range VLAN Configuration Guidelines, on page 551

Creating an Extended-Range VLAN , on page 562

Creating an Extended-Range VLAN with an Internal VLAN ID, on page 565

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549

Creating or Modifying an Ethernet VLAN , on page 553

Deleting a VLAN , on page 556

Assigning Static-Access Ports to a VLAN, on page 559
Supported VLANs, on page 545

Extended-Range VLAN Configuration Guidelines, on page 551

Creating an Extended-Range VLAN , on page 562

Creating an Extended-Range VLAN with an Internal VLAN ID, on page 565

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549

Creating or Modifying an Ethernet VLAN , on page 553

Deleting a VLAN , on page 556

Assigning Static-Access Ports to a VLAN, on page 559
Supported VLANs, on page 545

Extended-Range VLAN Configuration Guidelines, on page 551

Creating an Extended-Range VLAN , on page 562

Creating an Extended-Range VLAN with an Internal VLAN ID, on page 565

Creating or Modifying an Ethernet VLAN , on page 553

Normal-Range VLAN Configuration Guidelines, on page 549

Deleting a VLAN , on page 556

Assigning Static-Access Ports to a VLAN, on page 559
VLAN Port Membership Modes, on page 547

Creating an Extended-Range VLAN , on page 562

Extended-Range VLAN Configuration Guidelines, on page 551

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549

Creating or Modifying an Ethernet VLAN , on page 553

Deleting a VLAN , on page 556

Assigning Static-Access Ports to a VLAN, on page 559
Supported VLANs, on page 545

Extended-Range VLAN Configuration Guidelines, on page 551

Creating an Extended-Range VLAN , on page 562

Creating an Extended-Range VLAN with an Internal VLAN ID, on page 565

Supported VLANs, on page 545

Normal-Range VLAN Configuration Guidelines, on page 549
Creating or Modifying an Ethernet VLAN , on page 553
Deleting a VLAN , on page 556
Assigning Static-Access Ports to a VLAN, on page 559
Supported VLANs, on page 545
Extended-Range VLAN Configuration Guidelines, on page 551
Creating an Extended-Range VLAN , on page 562
Creating an Extended-Range VLAN with an Internal VLAN ID, on page 565
Supported VLANs, on page 545
Normal-Range VLAN Configuration Guidelines, on page 549
Creating or Modifying an Ethernet VLAN , on page 553
Deleting a VLAN , on page 556
Assigning Static-Access Ports to a VLAN, on page 559
Supported VLANs, on page 545
Extended-Range VLAN Configuration Guidelines, on page 551
Creating an Extended-Range VLAN , on page 562
Creating an Extended-Range VLAN with an Internal VLAN ID, on page 565
Supported VLANs, on page 545
Normal-Range VLAN Configuration Guidelines, on page 549
Creating or Modifying an Ethernet VLAN , on page 553
Deleting a VLAN , on page 556
Assigning Static-Access Ports to a VLAN, on page 559
Supported VLANs, on page 545
Extended-Range VLAN Configuration Guidelines, on page 551
Creating an Extended-Range VLAN , on page 562
Creating an Extended-Range VLAN with an Internal VLAN ID, on page 565

Troubleshooting VLANs

This section describes commands used to troubleshoot VLANs.

The following commands can be used to troubleshoot VLANs on the switch .

Table 72: VLAN Troubleshooting Commands

Command	Description
debug platform vlan [**error** [**switch** *number*] | **event** [**switch** *number*] | **switch** *number*]	Debugs platform related VLAN issues

Command	Description
debug switchport {**backup** { **all** | **errors** | **events** | **vlan-load-balancing** } }	Debugs the switchport configuration. The following debug switchport options are available: • **all—**All switch backup interface debugging messages. • **errors—**Switch backup interface errors or exceptions. • **events—**Switch backup interface events. • **vlan-load-balancing—**Switch backup interface VLAN load balancing.
debug sw-vlan { **badpmcookies** | **cfg-vlan** | **events** | **ifs** | **management** | **mapping** | **notification** | **packets** | **redundancy** | **registries** | **vtp** }	Debugs the VLAN manager. The following command options are available: • **badpmcookies—**Debugs VLAN manager incidents of bad Port Manager (pm) cookies. • **cfg-vlan—**Debugs the VLAN configuration. • **events—**Debugs VLAN manager events. • **ifs—**Debugs VLAN manager ifs error tests. • **management—**Debugs VLAN manager management of internal vlans. • **mapping—**Debugs VLAN mapping. • **notification—**Debugs VLAN manager notifications. • **packets—**Debugs VLAN manager packets. • **redundancy—**Debugs VTP VLAN redundancy. • **registries—**Debugs VLAN manager registries. • **vtp—**VTP protocol debugging.
debug vlan { **configuration** *vlan* }	Debugs the VLAN. Specify the VLAN or VLAN range for debugging.

Where to Go Next

After configuring VLANs, you can configure the following:

- VLAN groups
- VLAN Trunking Protocol (VTP)

- VLAN trunks
- VLAN Membership Policy Server (VMPS)
- Voice VLANs

Additional References

Related Documents

Related Topic	Document Title
CLI commands	*Layer 2/3 Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)* *VLAN Command Reference (Catalyst 3850 Switches)*
VLAN access-maps	*Security Configuration Guide, Cisco IOS Release 3SE (Catalyst 3850 Switches)* *Security Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*
Cisco Flexible NetFlow	*Cisco Flexible NetFlow Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)* *Flexible Netflow Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*
IGMP Snooping	*IP Multicast Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)* *IP Multicast Routing Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*
IPv6	*Catalyst 3850 IPv6 Configuration Guide, Release 3.2SE* *IPv6 Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*
SPAN	*Network Management Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)* *Network Management Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*

Standards and RFCs

Standard/RFC	Title
RFC 1573	Evolution of the Interfaces Group of MIB-II
RFC 1757	Remote Network Monitoring Management
RFC 2021	SNMPv2 Management Information Base for the Transmission Control Protocol using SMIv2

MIBs

MIB	MIBs Link
All supported MIBs for this release. • BRIDGE-MIB (RFC1493) • CISCO-BRIDGE-EXT-MIB • CISCO-CDP-MIB • CISCO-PAGP-MIB • CISCO-PRIVATE-VLAN-MIB • CISCO-LAG-MIB • CISCO-L2L3-INTERFACE-CONFIG-MIB • CISCO-MAC-NOTIFICATION-MIB • CISCO-STP-EXTENSIONS-MIB • CISCO-VLAN-IFTABLE-RELATIONSHIP-MIB • CISCO-VLAN-MEMBERSHIP-MIB • CISCO-VTP-MIB • IEEE8023-LAG-MIB • IF-MIB (RFC 1573) • RMON-MIB (RFC 1757) • RMON2-MIB (RFC 2021)	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for VLANs

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.
Cisco IOS XE 3.3SE	VLAN GUI support.

CHAPTER 34

Configuring VLAN Trunks

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for VLAN Trunks

The IEEE 802.1Q trunks impose these limitations on the trunking strategy for a network:

- In a network of Cisco switches connected through IEEE 802.1Q trunks, the switches maintain one spanning-tree instance for each VLAN allowed on the trunks. Non-Cisco devices might support one spanning-tree instance for all VLANs.

  When you connect a Cisco switch to a non-Cisco device through an IEEE 802.1Q trunk, the Cisco switch combines the spanning-tree instance of the VLAN of the trunk with the spanning-tree instance of the non-Cisco IEEE 802.1Q switch. However, spanning-tree information for each VLAN is maintained by Cisco switches separated by a cloud of non-Cisco IEEE 802.1Q switches. The non-Cisco IEEE 802.1Q cloud separating the Cisco switches is treated as a single trunk link between the switches.

- Make sure the native VLAN for an IEEE 802.1Q trunk is the same on both ends of the trunk link. If the native VLAN on one end of the trunk is different from the native VLAN on the other end, spanning-tree loops might result.
- Disabling spanning tree on the native VLAN of an IEEE 802.1Q trunk without disabling spanning tree on every VLAN in the network can potentially cause spanning-tree loops. We recommend that you leave spanning tree enabled on the native VLAN of an IEEE 802.1Q trunk or disable spanning tree on every VLAN in the network. Make sure your network is loop-free before disabling spanning tree.

Restrictions for VLAN Trunks

The following are restrictions for VLAN trunks:

- Dynamic Trunking Protocol (DTP) is not supported on private-VLAN ports or tunnel ports.
- The switch does not support Layer 3 trunks; you cannot configure subinterfaces or use the **encapsulation** keyword on Layer 3 interfaces. The switch does support Layer 2 trunks and Layer 3 VLAN interfaces, which provide equivalent capabilities.
- You cannot have a switch stack containing a mix of Catalyst 3850 and Catalyst 3650 switches.

Information About VLAN Trunks

Trunking Overview

A trunk is a point-to-point link between one or more Ethernet switch interfaces and another networking device such as a router or a switch. Ethernet trunks carry the traffic of multiple VLANs over a single link, and you can extend the VLANs across an entire network.

The following trunking encapsulations are available on all Ethernet interfaces:

- IEEE 802.1Q— Industry-standard trunking encapsulation.

Trunking Modes

Ethernet trunk interfaces support different trunking modes. You can set an interface as trunking or nontrunking or to negotiate trunking with the neighboring interface. To autonegotiate trunking, the interfaces must be in the same VTP domain.

Trunk negotiation is managed by the Dynamic Trunking Protocol (DTP), which is a Point-to-Point Protocol (PPP). However, some internetworking devices might forward DTP frames improperly, which could cause misconfigurations.

Related Topics

Configuring a Trunk Port , on page 581

Layer 2 Interface Modes, on page 577

Layer 2 Interface Modes

Table 73: Layer 2 Interface Modes

Mode	Function
switchport mode access	Puts the interface (access port) into permanent nontrunking mode and negotiates to convert the link into a nontrunk link. The interface becomes a nontrunk interface regardless of whether or not the neighboring interface is a trunk interface.
switchport mode dynamic auto	Makes the interface able to convert the link to a trunk link. The interface becomes a trunk interface if the neighboring interface is set to **trunk** or **desirable** mode. The default switchport mode for all Ethernet interfaces is **dynamic auto**.
switchport mode dynamic desirable	Makes the interface actively attempt to convert the link to a trunk link. The interface becomes a trunk interface if the neighboring interface is set to **trunk**, **desirable**, or **auto** mode.
switchport mode trunk	Puts the interface into permanent trunking mode and negotiates to convert the neighboring link into a trunk link. The interface becomes a trunk interface even if the neighboring interface is not a trunk interface.
switchport nonegotiate	Prevents the interface from generating DTP frames. You can use this command only when the interface switchport mode is **access** or **trunk**. You must manually configure the neighboring interface as a trunk interface to establish a trunk link.
switchport mode dot1q-tunnel	Configures the interface as a tunnel (nontrunking) port to be connected in an asymmetric link with an IEEE 802.1Q trunk port. The IEEE 802.1Q tunneling is used to maintain customer VLAN integrity across a service provider network.

Related Topics

Configuring a Trunk Port , on page 581

Trunking Modes, on page 576

Ethernet Trunk Encapsulation Types

This table lists the Ethernet trunk encapsulation types and keywords.

Table 74: Ethernet Trunk Encapsulation Types and Keywords

Encapsulation	Function
switchport trunk encapsulation isl	Specifies ISL encapsulation on the trunk link.
switchport trunk encapsulation dot1q	Specifies IEEE 802.1Q encapsulation on the trunk link.
switchport trunk encapsulation negotiate	Specifies that the interface negotiate with the neighboring interface to become an ISL (preferred) or IEEE 802.1Q trunk, depending on the configuration and capabilities of the neighboring interface. This is the default for the switch.

The trunking mode, the trunk encapsulation type, and the hardware capabilities of the two connected interfaces decide whether a link becomes an ISL or IEEE 802.1Q trunk.

Allowed VLANs on a Trunk

By default, a trunk port sends traffic to and receives traffic from all VLANs. All VLAN IDs, 1 to 4094, are allowed on each trunk. However, you can remove VLANs from the allowed list, preventing traffic from those VLANs from passing over the trunk.

To reduce the risk of spanning-tree loops or storms, you can disable VLAN 1 on any individual VLAN trunk port by removing VLAN 1 from the allowed list. When you remove VLAN 1 from a trunk port, the interface continues to send and receive management traffic, for example, Cisco Discovery Protocol (CDP), Port Aggregation Protocol (PAgP), Link Aggregation Control Protocol (LACP), DTP, and VTP in VLAN 1.

If a trunk port with VLAN 1 disabled is converted to a nontrunk port, it is added to the access VLAN. If the access VLAN is set to 1, the port will be added to VLAN 1, regardless of the **switchport trunk allowed** setting. The same is true for any VLAN that has been disabled on the port.

A trunk port can become a member of a VLAN if the VLAN is enabled, if VTP knows of the VLAN, and if the VLAN is in the allowed list for the port. When VTP detects a newly enabled VLAN and the VLAN is in the allowed list for a trunk port, the trunk port automatically becomes a member of the enabled VLAN. When VTP detects a new VLAN and the VLAN is not in the allowed list for a trunk port, the trunk port does not become a member of the new VLAN.

Related Topics

Defining the Allowed VLANs on a Trunk , on page 583

Load Sharing on Trunk Ports

Load sharing divides the bandwidth supplied by parallel trunks connecting switches. To avoid loops, STP normally blocks all but one parallel link between switches. Using load sharing, you divide the traffic between the links according to which VLAN the traffic belongs.

You configure load sharing on trunk ports by using STP port priorities or STP path costs. For load sharing using STP port priorities, both load-sharing links must be connected to the same switch. For load sharing using STP path costs, each load-sharing link can be connected to the same switch or to two different switches.

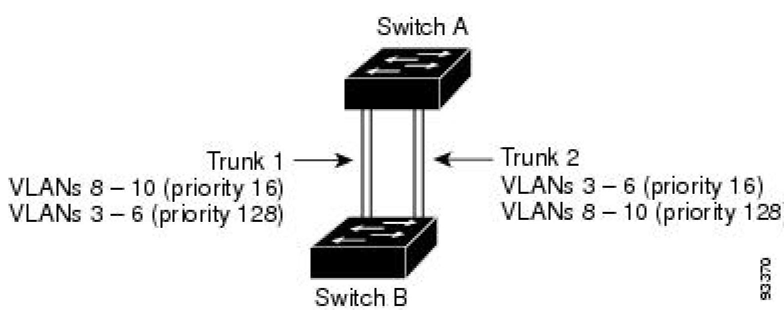
Network Load Sharing Using STP Priorities

When two ports on the same switch form a loop, the switch uses the STP port priority to decide which port is enabled and which port is in a blocking state. You can set the priorities on a parallel trunk port so that the port carries all the traffic for a given VLAN. The trunk port with the higher priority (lower values) for a VLAN is forwarding traffic for that VLAN. The trunk port with the lower priority (higher values) for the same VLAN remains in a blocking state for that VLAN. One trunk port sends or receives all traffic for the VLAN.

This figure shows two trunks connecting supported switches.

- VLANs 8 through 10 are assigned a port priority of 16 on Trunk 1.
- VLANs 3 through 6 retain the default port priority of 128 on Trunk 1.
- VLANs 3 through 6 are assigned a port priority of 16 on Trunk 2.
- VLANs 8 through 10 retain the default port priority of 128 on Trunk 2.

Figure 15: Load Sharing by Using STP Port Priorities



Trunk 1 carries traffic for VLANs 8 through 10, and Trunk 2 carries traffic for VLANs 3 through 6. If the active trunk fails, the trunk with the lower priority takes over and carries the traffic for all of the VLANs. No duplication of traffic occurs over any trunk port.

Related Topics

Configuring Load Sharing Using STP Port Priorities , on page 587

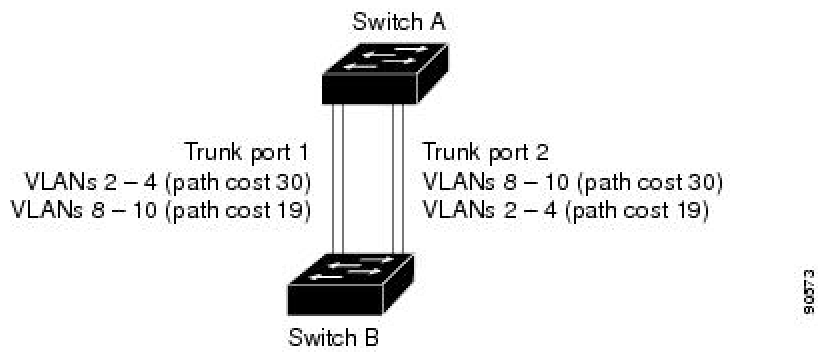
Network Load Sharing Using STP Path Cost

You can configure parallel trunks to share VLAN traffic by setting different path costs on a trunk and associating the path costs with different sets of VLANs, blocking different ports for different VLANs. The VLANs keep the traffic separate and maintain redundancy in the event of a lost link.

Trunk ports 1 and 2 are configured as 100BASE-T ports. These VLAN path costs are assigned:

- VLANs 2 through 4 are assigned a path cost of 30 on Trunk port 1.
- VLANs 8 through 10 retain the default 100BASE-T path cost on Trunk port 1 of 19.
- VLANs 8 through 10 are assigned a path cost of 30 on Trunk port 2.
- VLANs 2 through 4 retain the default 100BASE-T path cost on Trunk port 2 of 19.

Figure 16: Load-Sharing Trunks with Traffic Distributed by Path Cost



Related Topics

Configuring Load Sharing Using STP Path Cost , on page 591

Feature Interactions

Trunking interacts with other features in these ways:

- A trunk port cannot be a secure port.
- A trunk port cannot be a tunnel port.
- Trunk ports can be grouped into EtherChannel port groups, but all trunks in the group must have the same configuration. When a group is first created, all ports follow the parameters set for the first port to be added to the group. If you change the configuration of one of these parameters, the switch propagates the setting that you entered to all ports in the group:
    - Allowed-VLAN list.
    - STP port priority for each VLAN.
    - STP Port Fast setting.
    - Trunk status:

      If one port in a port group ceases to be a trunk, all ports cease to be trunks.
- We recommend that you configure no more than 24 trunk ports in Per VLAN Spanning Tree (PVST) mode and no more than 40 trunk ports in Multiple Spanning Tree (MST) mode.
- If you try to enable IEEE 802.1x on a trunk port, an error message appears, and IEEE 802.1x is not enabled. If you try to change the mode of an IEEE 802.1x-enabled port to trunk, the port mode is not changed.
- A port in dynamic mode can negotiate with its neighbor to become a trunk port. If you try to enable IEEE 802.1x on a dynamic port, an error message appears, and IEEE 802.1x is not enabled. If you try to change the mode of an IEEE 802.1x-enabled port to dynamic, the port mode is not changed.

How to Configure VLAN Trunks

To avoid trunking misconfigurations, configure interfaces connected to devices that do not support DTP to not forward DTP frames, that is, to turn off DTP.

- If you do not intend to trunk across those links, use the **switchport mode access** interface configuration command to disable trunking.
- To enable trunking to a device that does not support DTP, use the **switchport mode trunk** and **switchport nonegotiate** interface configuration commands to cause the interface to become a trunk but to not generate DTP frames.

Configuring an Ethernet Interface as a Trunk Port

Configuring a Trunk Port

Because trunk ports send and receive VTP advertisements, to use VTP you must ensure that at least one trunk port is configured on the switch and that this trunk port is connected to the trunk port of a second switch. Otherwise, the switch cannot receive any VTP advertisements.

Before You Begin

By default, an interface is in Layer 2 mode. The default mode for Layer 2 interfaces is **switchport mode dynamic auto**. If the neighboring interface supports trunking and is configured to allow trunking, the link is a Layer 2 trunk or, if the interface is in Layer 3 mode, it becomes a Layer 2 trunk when you enter the **switchport** interface configuration command.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode** {**dynamic** {**auto** | **desirable**} | **trunk**}
4. **switchport access vlan** *vlan-id*
5. **switchport trunk native vlan** *vlan-id*
6. **end**
7. **show interfaces** *interface-id* **switchport**
8. **show interfaces** *interface-id* **trunk**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/2`	Specifies the port to be configured for trunking, and enters interface configuration mode.
Step 3	**switchport mode** {**dynamic** {**auto** | **desirable**} | **trunk**} **Example:** `Switch(config-if)# switchport mode dynamic desirable`	Configures the interface as a Layer 2 trunk (required only if the interface is a Layer 2 access port or tunnel port or to specify the trunking mode). • **dynamic auto—**Sets the interface to a trunk link if the neighboring interface is set to trunk or desirable mode. This is the default. • **dynamic desirable—**Sets the interface to a trunk link if the neighboring interface is set to trunk, desirable, or auto mode. • **trunk—**Sets the interface in permanent trunking mode and negotiate to convert the link to a trunk link even if the neighboring interface is not a trunk interface.
Step 4	**switchport access vlan** *vlan-id* **Example:** `Switch(config-if)# switchport access vlan 200`	(Optional) Specifies the default VLAN, which is used if the interface stops trunking.
Step 5	**switchport trunk native vlan** *vlan-id* **Example:** `Switch(config-if)# switchport trunk native vlan 200`	Specifies the native VLAN for IEEE 802.1Q trunks.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 7	**show interfaces** *interface-id* **switchport** **Example:** `Switch# show interfaces gigabitethernet1/0/2 switchport`	Displays the switch port configuration of the interface in the *Administrative Mode* and the *Administrative Trunking Encapsulation* fields of the display.
Step 8	**show interfaces** *interface-id* **trunk** **Example:** `Switch# show interfaces gigabitethernet1/0/2 trunk`	Displays the trunk configuration of the interface.
Step 9	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Trunking Modes, on page 576

Layer 2 Interface Modes, on page 577

Defining the Allowed VLANs on a Trunk

VLAN 1 is the default VLAN on all trunk ports in all Cisco switches, and it has previously been a requirement that VLAN 1 always be enabled on every trunk link. You can use the VLAN 1 minimization feature to disable VLAN 1 on any individual VLAN trunk link so that no user traffic (including spanning-tree advertisements) is sent or received on VLAN 1.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode trunk**
4. **switchport trunk allowed vlan** { *word* | **add** | **all** | **except** | **none** | **remove**} *vlan-list*
5. **end**
6. **show interfaces** *interface-id* **switchport**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Specifies the port to be configured, and enters interface configuration mode.
Step 3	**switchport mode trunk** **Example:** `Switch(config-if)# switchport mode trunk`	Configures the interface as a VLAN trunk port.
Step 4	**switchport trunk allowed vlan** { *word* | **add** | **all** | **except** | **none** | **remove**} *vlan-list* **Example:** `Switch(config-if)# switchport trunk allowed vlan remove 2`	(Optional) Configures the list of VLANs allowed on the trunk. The *vlan-list* parameter is either a single VLAN number from 1 to 4094 or a range of VLANs described by two VLAN numbers, the lower one first, separated by a hyphen. Do not enter any spaces between comma-separated VLAN parameters or in hyphen-specified ranges. All VLANs are allowed by default.
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 6	**show interfaces** *interface-id* **switchport** **Example:** `Switch# show interfaces gigabitethernet1/0/1 switchport`	Verifies your entries in the *Trunking VLANs Enabled* field of the display.
Step 7	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Allowed VLANs on a Trunk, on page 578

Changing the Pruning-Eligible List

The pruning-eligible list applies only to trunk ports. Each trunk port has its own eligibility list. VTP pruning must be enabled for this procedure to take effect.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport trunk pruning vlan** {**add** | **except** | **none** | **remove**} *vlan-list* [,*vlan* [,*vlan* [,,,]]
4. **end**
5. **show interfaces** *interface-id* **switchport**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/1`	Selects the trunk port for which VLANs should be pruned, and enters interface configuration mode.
Step 3	**switchport trunk pruning vlan** {**add** | **except** | **none** | **remove**} *vlan-list* [,*vlan* [,*vlan* [,,,]]	Configures the list of VLANs allowed to be pruned from the trunk. For explanations about using the **add**, **except**, **none**, and **remove** keywords, see the command reference for this release. Separate non-consecutive VLAN IDs with a comma and no spaces; use a hyphen to designate a range of IDs. Valid IDs are 2 to 1001. Extended-range VLANs (VLAN IDs 1006 to 4094) cannot be pruned. VLANs that are pruning-ineligible receive flooded traffic. The default list of VLANs allowed to be pruned contains VLANs 2 to 1001.

	Command or Action	Purpose
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show interfaces** *interface-id* **switchport** **Example:** `Switch# show interfaces gigabitethernet2/0/1 switchport`	Verifies your entries in the *Pruning VLANs Enabled* field of the display.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring the Native VLAN for Untagged Traffic

A trunk port configured with IEEE 802.1Q tagging can receive both tagged and untagged traffic. By default, the switch forwards untagged traffic in the native VLAN configured for the port. The native VLAN is VLAN 1 by default.

The native VLAN can be assigned any VLAN ID.

If a packet has a VLAN ID that is the same as the outgoing port native VLAN ID, the packet is sent untagged; otherwise, the switch sends the packet with a tag.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport trunk native vlan** *vlan-id*
4. **end**
5. **show interfaces** *interface-id* **switchport**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/2`	Defines the interface that is configured as the IEEE 802.1Q trunk, and enters interface configuration mode.
Step 3	**switchport trunk native vlan** *vlan-id* **Example:** `Switch(config-if)# switchport trunk native vlan 12`	Configures the VLAN that is sending and receiving untagged traffic on the trunk port. For *vlan-id*, the range is 1 to 4094.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show interfaces** *interface-id* **switchport** **Example:** `Switch# show interfaces gigabitethernet1/0/2 switchport`	Verifies your entries in the *Trunking Native Mode VLAN* field.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Trunk Ports for Load Sharing

Configuring Load Sharing Using STP Port Priorities

If your switch is a member of a switch stack, you must use the **spanning-tree** [**vlan** *vlan-id*] **cost** *cost* interface configuration command instead of the **spanning-tree** [**vlan** *vlan-id*] **port-priority** *priority* interface configuration command to select an interface to put in the forwarding state. Assign lower cost values to interfaces that you want selected first and higher cost values that you want selected last.

These steps describe how to configure a network with load sharing using STP port priorities.

SUMMARY STEPS

1. **configure terminal**
2. **vtp domain** *domain-name*
3. **vtp mode server**
4. **end**
5. **show vtp status**
6. **show vlan**
7. **configure terminal**
8. **interface** *interface-id*
9. **switchport mode trunk**
10. **end**
11. **show interfaces** *interface-id* **switchport**
12. Repeat the above steps on Switch A for a second port in the switch or switch stack.
13. Repeat the above steps on Switch B to configure the trunk ports that connect to the trunk ports configured on Switch A.
14. **show vlan**
15. **configure terminal**
16. **interface** *interface-id*
17. **spanning-tree vlan** *vlan-range* **port-priority** *priority-value*
18. **exit**
19. **interface** *interface-id*
20. **spanning-tree vlan** *vlan-range* **port-priority** *priority-value*
21. **end**
22. **show running-config**
23. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode on Switch A.
Step 2	**vtp domain** *domain-name*	Configures a VTP administrative domain.

	Command or Action	Purpose
	Example: `Switch(config)# vtp domain workdomain`	The domain name can be 1 to 32 characters.
Step 3	**vtp mode server** **Example:** `Switch(config)# vtp mode server`	Configures Switch A as the VTP server.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show vtp status** **Example:** `Switch# show vtp status`	Verifies the VTP configuration on both Switch A and Switch B. In the display, check the *VTP Operating Mode* and the *VTP Domain Name* fields.
Step 6	**show vlan** **Example:** `Switch# show vlan`	Verifies that the VLANs exist in the database on Switch A.
Step 7	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 8	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Defines the interface to be configured as a trunk, and enters interface configuration mode.
Step 9	**switchport mode trunk** **Example:** `Switch(config-if)# switchport mode trunk`	Configures the port as a trunk port.

	Command or Action	Purpose
Step 10	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 11	**show interfaces** *interface-id* **switchport** **Example:** `Switch# show interfaces gigabitethernet1/0/1 switchport`	Verifies the VLAN configuration.
Step 12	Repeat the above steps on Switch A for a second port in the switch or switch stack.	
Step 13	Repeat the above steps on Switch B to configure the trunk ports that connect to the trunk ports configured on Switch A.	
Step 14	**show vlan** **Example:** `Switch# show vlan`	When the trunk links come up, VTP passes the VTP and VLAN information to Switch B. This command verifies that Switch B has learned the VLAN configuration.
Step 15	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode on Switch A.
Step 16	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Defines the interface to set the STP port priority, and enters interface configuration mode.
Step 17	**spanning-tree vlan** *vlan-range* **port-priority** *priority-value* **Example:** `Switch(config-if)# spanning-tree vlan 8-10 port-priority 16`	Assigns the port priority for the VLAN range specified. Enter a port priority value from 0 to 240. Port priority values increment by 16.
Step 18	**exit** **Example:** `Switch(config-if)# exit`	Returns to global configuration mode.

	Command or Action	Purpose
Step 19	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/2`	Defines the interface to set the STP port priority, and enters interface configuration mode.
Step 20	**spanning-tree vlan** *vlan-range* **port-priority** *priority-value* **Example:** `Switch(config-if)# spanning-tree vlan 3-6 port-priority 16`	Assigns the port priority for the VLAN range specified. Enter a port priority value from 0 to 240. Port priority values increment by 16.
Step 21	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 22	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 23	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Network Load Sharing Using STP Priorities, on page 579

Configuring Load Sharing Using STP Path Cost

These steps describe how to configure a network with load sharing using STP path costs.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode trunk**
4. **exit**
5. Repeat Steps 2 through 4 on a second interface in Switch A or in Switch A stack.
6. **end**
7. **show running-config**
8. **show vlan**
9. **configure terminal**
10. **interface** *interface-id*
11. **spanning-tree vlan** *vlan-range* **cost** *cost-value*
12. **end**
13. Repeat Steps 9 through 13 on the other configured trunk interface on Switch A, and set the spanning-tree path cost to 30 for VLANs 8, 9, and 10.
14. **exit**
15. **show running-config**
16. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode on Switch A.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Defines the interface to be configured as a trunk, and enters interface configuration mode.
Step 3	**switchport mode trunk** **Example:** `Switch(config-if)# switchport mode trunk`	Configures the port as a trunk port.

	Command or Action	Purpose
Step 4	**exit** **Example:** `Switch(config-if)# exit`	Returns to global configuration mode.
Step 5	Repeat Steps 2 through 4 on a second interface in Switch A or in Switch A stack.	
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 7	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries. In the display, make sure that the interfaces are configured as trunk ports.
Step 8	**show vlan** **Example:** `Switch# show vlan`	When the trunk links come up, Switch A receives the VTP information from the other switches. This command verifies that Switch A has learned the VLAN configuration.
Step 9	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 10	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Defines the interface on which to set the STP cost, and enters interface configuration mode.
Step 11	**spanning-tree vlan** *vlan-range* **cost** *cost-value* **Example:** `Switch(config-if)# spanning-tree vlan 2-4 cost 30`	Sets the spanning-tree path cost to 30 for VLANs 2 through 4.

	Command or Action	Purpose
Step 12	**end** **Example:** `Switch(config-if)# end`	Returns to global configuration mode.
Step 13	Repeat Steps 9 through 13 on the other configured trunk interface on Switch A, and set the spanning-tree path cost to 30 for VLANs 8, 9, and 10.	
Step 14	**exit** **Example:** `Switch(config)# exit`	Returns to privileged EXEC mode.
Step 15	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries. In the display, verify that the path costs are set correctly for both trunk interfaces.
Step 16	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Network Load Sharing Using STP Path Cost, on page 579

Where to Go Next

After configuring VLAN trunks, you can configure the following:

- VLANs
- VLAN groups
- VLAN Membership Policy Server (VMPS)
- Voice VLANs

Additional References

Related Documents

Related Topic	Document Title
CLI commands	*Layer 2/3 Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)* *VLAN Command Reference (Catalyst 3850 Switches)*
Spanning Tree Protocol (STP)	*Network Management Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)* *Network Management Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*

Standards and RFCs

Standard/RFC	Title
RFC 1573	Evolution of the Interfaces Group of MIB-II
RFC 1757	Remote Network Monitoring Management
RFC 2021	SNMPv2 Management Information Base for the Transmission Control Protocol using SMIv2

MIBs

MIB	MIBs Link
All supported MIBs for this release. • BRIDGE-MIB (RFC1493) • CISCO-BRIDGE-EXT-MIB • CISCO-CDP-MIB • CISCO-PAGP-MIB • CISCO-PRIVATE-VLAN-MIB • CISCO-LAG-MIB • CISCO-L2L3-INTERFACE-CONFIG-MIB • CISCO-MAC-NOTIFICATION-MIB • CISCO-STP-EXTENSIONS-MIB • CISCO-VLAN-IFTABLE-RELATIONSHIP-MIB • CISCO-VLAN-MEMBERSHIP-MIB • CISCO-VTP-MIB • IEEE8023-LAG-MIB • IF-MIB (RFC 1573) • RMON-MIB (RFC 1757) • RMON2-MIB (RFC 2021)	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for VLAN Trunks

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 35

Configuring Voice VLANs

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for Voice VLANs

The following are the prerequisites for voice VLANs:

- Voice VLAN configuration is only supported on switch access ports; voice VLAN configuration is not supported on trunk ports.

> **Note** Trunk ports can carry any number of voice VLANs, similar to regular VLANs. The configuration of voice VLANs is not supported on trunk ports.

- Before you enable voice VLAN, we recommend that you enable QoS on the switch by entering the **mls qos** global configuration command and configure the port trust state to trust by entering the **mls qos trust cos** interface configuration command. If you use the auto-QoS feature, these settings are automatically configured.
- Before you enable voice VLAN, enable QoS on the switch by entering the **trust device cisco-phone** interface configuration command. If you use the auto QoS feature, these settings are automatically configured.
- You must enable CDP on the switch port connected to the Cisco IP Phone to send the configuration to the phone. (CDP is globally enabled by default on all switch interfaces.)

Restrictions for Voice VLANs

You cannot configure static secure MAC addresses in the voice VLAN.

Information About Voice VLAN

Voice VLANs

The voice VLAN feature enables access ports to carry IP voice traffic from an IP phone. When the switch is connected to a Cisco 7960 IP Phone, the phone sends voice traffic with Layer 3 IP precedence and Layer 2 class of service (CoS) values, which are both set to 5 by default. Because the sound quality of an IP phone call can deteriorate if the data is unevenly sent, the switch supports quality of service (QoS) based on IEEE 802.1p CoS. QoS uses classification and scheduling to send network traffic from the switch in a predictable manner.

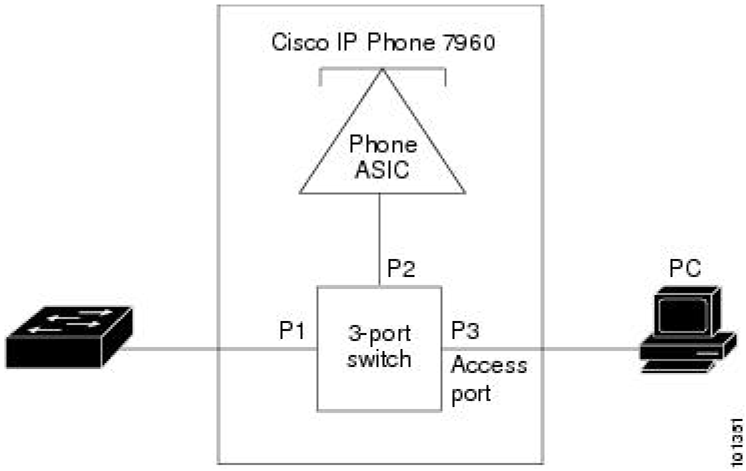
The Cisco 7960 IP Phone is a configurable device, and you can configure it to forward traffic with an IEEE 802.1p priority. You can configure the switch to trust or override the traffic priority assigned by a Cisco IP Phone.

This network configuration is one way to connect a Cisco 7960 IP Phone.

The Cisco IP Phone contains an integrated three-port 10/100 switch. The ports provide dedicated connections to these devices:

- Port 1 connects to the switch or other voice-over-IP (VoIP) device.
- Port 2 is an internal 10/100 interface that carries the IP phone traffic.
- Port 3 (access port) connects to a PC or other device.

Figure 17: Cisco 7960 IP Phone Connected to aSwitch



Cisco IP Phone Voice Traffic

You can configure an access port with an attached Cisco IP Phone to use one VLAN for voice traffic and another VLAN for data traffic from a device attached to the phone. You can configure access ports on the switch to send Cisco Discovery Protocol (CDP) packets that instruct an attached phone to send voice traffic to the switch in any of these ways:

- In the voice VLAN tagged with a Layer 2 CoS priority value
- In the access VLAN tagged with a Layer 2 CoS priority value
- In the access VLAN, untagged (no Layer 2 CoS priority value)

Note In all configurations, the voice traffic carries a Layer 3 IP precedence value (the default is 5 for voice traffic and 3 for voice control traffic).

Related Topics

Configuring Cisco IP Phone Voice Traffic , on page 603

Monitoring Voice VLAN, on page 607

Cisco IP Phone Data Traffic

The switch can also process tagged data traffic (traffic in IEEE 802.1Q or IEEE 802.1p frame types) from the device attached to the access port on the Cisco IP Phone. You can configure Layer 2 access ports on the switch to send CDP packets that instruct the attached phone to configure the phone access port in one of these modes:

- In trusted mode, all traffic received through the access port on the Cisco IP Phone passes through the phone unchanged.

- In untrusted mode, all traffic in IEEE 802.1Q or IEEE 802.1p frames received through the access port on the Cisco IP Phone receive a configured Layer 2 CoS value. The default Layer 2 CoS value is 0. Untrusted mode is the default.

Note Untagged traffic from the device attached to the Cisco IP Phone passes through the phone unchanged, regardless of the trust state of the access port on the phone.

Related Topics

Configuring the Priority of Incoming Data Frames , on page 606

Monitoring Voice VLAN, on page 607

Voice VLAN Configuration Guidelines

- Because a Cisco 7960 IP Phone also supports a connection to a PC or other device, a port connecting the switch to a Cisco IP Phone can carry mixed traffic. You can configure a port to decide how the Cisco IP Phone carries voice traffic and data traffic.
- The voice VLAN should be present and active on the switch for the IP phone to correctly communicate on the voice VLAN. Use the **show vlan** privileged EXEC command to see if the VLAN is present (listed in the display). If the VLAN is not listed, create the voice VLAN.
- The Power over Ethernet (PoE) switches are capable of automatically providing power to Cisco pre-standard and IEEE 802.3af-compliant powered devices if they are not being powered by an AC power source.
- The Port Fast feature is automatically enabled when voice VLAN is configured. When you disable voice VLAN, the Port Fast feature is not automatically disabled.
- If the Cisco IP Phone and a device attached to the phone are in the same VLAN, they must be in the same IP subnet. These conditions indicate that they are in the same VLAN:
    - They both use IEEE 802.1p or untagged frames.
    - The Cisco IP Phone uses IEEE 802.1p frames, and the device uses untagged frames.
    - The Cisco IP Phone uses untagged frames, and the device uses IEEE 802.1p frames.
    - The Cisco IP Phone uses IEEE 802.1Q frames, and the voice VLAN is the same as the access VLAN.
- The Cisco IP Phone and a device attached to the phone cannot communicate if they are in the same VLAN and subnet but use different frame types because traffic in the same subnet is not routed (routing would eliminate the frame type difference).
- Voice VLAN ports can also be these port types:
    - Dynamic access port.
    - IEEE 802.1x authenticated port.

Note If you enable IEEE 802.1x on an access port on which a voice VLAN is configured and to which a Cisco IP Phone is connected, the phone loses connectivity to the switch for up to 30 seconds.

◦ Protected port.

◦ A source or destination port for a SPAN or RSPAN session.

◦ Secure port.

Note When you enable port security on an interface that is also configured with a voice VLAN, you must set the maximum allowed secure addresses on the port to two plus the maximum number of secure addresses allowed on the access VLAN. When the port is connected to a Cisco IP Phone, the phone requires up to two MAC addresses. The phone address is learned on the voice VLAN and might also be learned on the access VLAN. Connecting a PC to the phone requires additional MAC addresses.

How to Configure Voice VLAN

Configuring Cisco IP Phone Voice Traffic

You can configure a port connected to the Cisco IP Phone to send CDP packets to the phone to configure the way in which the phone sends voice traffic. The phone can carry voice traffic in IEEE 802.1Q frames for a specified voice VLAN with a Layer 2 CoS value. It can use IEEE 802.1p priority tagging to give voice traffic a higher priority and forward all voice traffic through the native (access) VLAN. The Cisco IP Phone can also send untagged voice traffic or use its own configuration to send voice traffic in the access VLAN. In all configurations, the voice traffic carries a Layer 3 IP precedence value (the default is 5).

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **mls qos trust cos**
4. **trust device cisco-phone**
5. **switchport voice vlan** {*vlan-id* | **dot1p** | **none** | **untagged**}
6. **switchport voice** {**detect cisco-phone** [**full-duplex**] }
7. **end**
8. Use one of the following:
   - **show interfaces** *interface-id* **switchport**
   - **show running-config interface** *interface-id*
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Specifies the interface connected to the phone, and enters interface configuration mode.
Step 3	**mls qos trust cos** **Example:** `Switch(config-if)# mls qos trust cos`	Configures the interface to classify incoming traffic packets by using the packet CoS value. For untagged packets, the port default CoS value is used. **Note** Before configuring the port trust state, you must first globally enable QoS by using the **mls qos** global configuration command.
Step 4	**trust device cisco-phone** **Example:** `Switch(config-if)# trust-device cisco-phone`	Configures the interface to trust incoming traffic packets for the Cisco IP phone.
Step 5	**switchport voice vlan** {*vlan-id* | **dot1p** | **none** | **untagged**}	Configures the voice VLAN.

	Command or Action	Purpose
	Example: `Switch(config-if)# switchport voice vlan dot1p`	• *vlan-id*—Configures the phone to forward all voice traffic through the specified VLAN. By default, the Cisco IP Phone forwards the voice traffic with an IEEE 802.1Q priority of 5. Valid VLAN IDs are 1 to 4094. • **dot1p—**Configures the switch to accept voice and data IEEE 802.1p priority frames tagged with VLAN ID 0 (the native VLAN). By default, the switch drops all voice and data traffic tagged with VLAN 0. If configured for 802.1p the Cisco IP Phone forwards the traffic with an IEEE 802.1p priority of 5. • **none—**Allows the phone to use its own configuration to send untagged voice traffic. • **untagged—**Configures the phone to send untagged voice traffic.
Step 6	**switchport voice** {**detect cisco-phone** [**full-duplex**] } **Example:** `Switch(config-if)# switchport voice detect cisco-phone full-duplex`	Configures how the Cisco IP Phone carries voice traffic: • **detect—**Configures the interface to detect and recognize a Cisco IP Phone. • **cisco-phone—**When you initially enter the **switchport voice detect** command, this is the only allowed option. The default is **no switchport voice detect cisco-phone** [**full-duplex**]. • **full-duplex—**(Optional) Configures the switch to only accept a full-duplex Cisco IP Phone.
Step 7	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Step 8	Use one of the following:
	• **show interfaces** *interface-id* **switchport**
	• **show running-config interface** *interface-id*
	Example:
	`Switch# show interfaces gigabitethernet1/0/1 switchport`
	or
	`Switch# show running-config interface gigabitethernet1/0/1`
		Verifies your voice VLAN entries or your QoS and voice VLAN entries.

	Command or Action	Purpose
Step 9	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Cisco IP Phone Voice Traffic, on page 601
Monitoring Voice VLAN, on page 607

Configuring the Priority of Incoming Data Frames

You can connect a PC or other data device to a Cisco IP Phone port. To process tagged data traffic (in IEEE 802.1Q or IEEE 802.1p frames), you can configure the switch to send CDP packets to instruct the phone how to send data packets from the device attached to the access port on the Cisco IP Phone. The PC can generate packets with an assigned CoS value. You can configure the phone to not change (trust) or to override (not trust) the priority of frames arriving on the phone port from connected devices.

Follow these steps to set the priority of data traffic received from the non-voice port on the Cisco IP Phone:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport priority extend** {**cos** *value* | **trust**}
4. **end**
5. **show interfaces** *interface-id* **switchport**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Specifies the interface connected to the Cisco IP Phone, and enters interface configuration mode.
Step 3	**switchport priority extend** {**cos** *value* | **trust**} **Example:** `Switch(config-if)# switchport priority extend trust`	Sets the priority of data traffic received from the Cisco IP Phone access port: • **cos** *value*—Configures the phone to override the priority received from the PC or the attached device with the specified CoS value. The value is a number from 0 to 7, with 7 as the highest priority. The default priority is **cos** 0. • **trust**—Configures the phone access port to trust the priority received from the PC or the attached device.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show interfaces** *interface-id* **switchport** **Example:** `Switch# show interfaces gigabitethernet1/0/1 switchport`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Cisco IP Phone Data Traffic, on page 601
Monitoring Voice VLAN, on page 607

Monitoring Voice VLAN

To display voice VLAN configuration for an interface, use the **show interfaces** *interface-id* **switchport** privileged EXEC command.

Related Topics

Configuring Cisco IP Phone Voice Traffic , on page 603

Cisco IP Phone Voice Traffic, on page 601

Configuring the Priority of Incoming Data Frames , on page 606

Cisco IP Phone Data Traffic, on page 601

Troubleshooting Voice VLANs

To troubleshoot voice VLAN configuration for an interface, use the following privileged EXEC commands:

- **debug platform vlan**
- **debug switchport backup**
- **debug sw-vlan**
- **debug vlan configuration** *vlan*

Where to Go Next

After configuring voice VLANs, you can configure the following:

- VLANs
- VLAN groups
- VLAN Trunking
- VLAN Membership Policy Server (VMPS)
- VTP

Additional References

Related Documents

Related Topic	Document Title
CLI commands	*Layer 2/3 Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)* *VLAN Command Reference (Catalyst 3850 Switches)*

Standards and RFCs

Standard/RFC	Title
RFC 1573	Evolution of the Interfaces Group of MIB-II
RFC 1757	Remote Network Monitoring Management
RFC 2021	SNMPv2 Management Information Base for the Transmission Control Protocol using SMIv2

MIBs

MIB	MIBs Link
All supported MIBs for this release. • BRIDGE-MIB (RFC1493) • CISCO-BRIDGE-EXT-MIB • CISCO-CDP-MIB • CISCO-PAGP-MIB • CISCO-PRIVATE-VLAN-MIB • CISCO-LAG-MIB • CISCO-L2L3-INTERFACE-CONFIG-MIB • CISCO-MAC-NOTIFICATION-MIB • CISCO-STP-EXTENSIONS-MIB • CISCO-VLAN-IFTABLE-RELATIONSHIP-MIB • CISCO-VLAN-MEMBERSHIP-MIB • CISCO-VTP-MIB • IEEE8023-LAG-MIB • IF-MIB (RFC 1573) • RMON-MIB (RFC 1757) • RMON2-MIB (RFC 2021)	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Voice VLAN

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

PART V

Multicast Routing

CHAPTER 36

Configuring MSDP

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information About Configuring MSDP

This section describes how to configure the Multicast Source Discovery Protocol (MSDP on the switch. The MSDP connects multiple Protocol-Independent Multicast sparse-mode (PIM-SM) domains.

MSDP is not fully supported in this software release because of a lack of support for Multicast Border Gateway Protocol (MBGP), which works closely with MSDP. However, it is possible to create default peers that MSDP can operate with if MBGP is not running.

To use this feature, the active switch must be running the IP services feature set.

Understanding MSDP

MSDP allows multicast sources for a group to be known to all rendezvous points (RPs) in different domains. Each PIM-SM domain uses its own RPs and does not depend on RPs in other domains. An RP runs MSDP over the Transmission Control Protocol (TCP) to discover multicast sources in other domains.

An RP in a PIM-SM domain has an MSDP peering relationship with MSDP-enabled devices in another domain. The peering relationship occurs over a TCP connection, primarily exchanging a list of sources sending to multicast groups. The TCP connections between RPs are achieved by the underlying routing system. The receiving RP uses the source lists to establish a source path.

The purpose of this topology is to have domains discover multicast sources in other domains. If the multicast sources are of interest to a domain that has receivers, multicast data is delivered over the normal, source-tree building mechanism in PIM-SM. MSDP is also used to announce sources sending to a group. These announcements must originate at the domain's RP.

MSDP depends heavily on the Border Gateway Protocol (BGP) or MBGP for interdomain operation. We recommend that you run MSDP in RPs in your domain that are RPs for sources sending to global groups to be announced to the Internet.

MSDP Operation

When a source sends its first multicast packet, the first-hop router (*designated router* or RP) directly connected to the source sends a PIM register message to the RP. The RP uses the register message to register the active source and to forward the multicast packet down the shared tree in the local domain. With MSDP configured, the RP also forwards a source-active (SA) message to all MSDP peers. The SA message identifies the source, the group the source is sending to, and the address of the RP or the originator ID (the IP address of the interface used as the RP address), if configured.

Each MSDP peer receives and forwards the SA message away from the originating RP to achieve peer reverse-path flooding (RPF). The MSDP device examines the BGP or MBGP routing table to discover which peer is the next hop toward the originating RP of the SA message. Such a peer is called an *RPF peer* (reverse-path forwarding peer). The MSDP device forwards the message to all MSDP peers other than the RPF peer. For information on how to configure an MSDP peer when BGP and MBGP are not supported, see the Configuring a Default MSDP Peer, on page 615.

If the MSDP peer receives the same SA message from a non-RPF peer toward the originating RP, it drops the message. Otherwise, it forwards the message to all its MSDP peers.

The RP for a domain receives the SA message from an MSDP peer. If the RP has any join requests for the group the SA message describes and if the (*,G) entry exists with a nonempty outgoing interface list, the domain is interested in the group, and the RP triggers an (S,G) join toward the source. After the (S,G) join reaches the source's DR, a branch of the source tree has been built from the source to the RP in the remote domain. Multicast traffic can now flow from the source across the source tree to the RP and then down the shared tree in the remote domain to the receiver.

MSDP Benefits

MSDP has these benefits:

- It breaks up the shared multicast distribution tree. You can make the shared tree local to your domain. Your local members join the local tree, and join messages for the shared tree never need to leave your domain.

- PIM sparse-mode domains can rely only on their own RPs, decreasing reliance on RPs in another domain. This increases security because you can prevent your sources from being known outside your domain.
- Domains with only receivers can receive data without globally advertising group membership.
- Global source multicast routing table state is not required, saving memory.

How to Configure MSDP

Default MSDP Configuration

MSDP is not enabled, and no default MSDP peer exists.

Configuring a Default MSDP Peer

In this software release, because BGP and MBGP are not supported, you cannot configure an MSDP peer on the local switch by using the **ip msdp peer** global configuration command. Instead, you define a default MSDP peer (by using the **ip msdp default-peer** global configuration command) from which to accept all SA messages for the switch. The default MSDP peer must be a previously configured MSDP peer. Configure a default MSDP peer when the switch is not BGP- or MBGP-peering with an MSDP peer. If a single MSDP peer is configured, the switch always accepts all SA messages from that peer.

This figure shows a network in which default MSDP peers might be used. A customer who owns Switch B is connected to the Internet through two Internet service providers (ISPs), one owning Router A and the other owning Router C. They are not running BGP or MBGP between them. To learn about sources in the ISP's domain or in other domains, Switch B at the customer site identifies Router A as its default MSDP peer. Switch B advertises SA messages to both Router A and Router C but accepts SA messages only from Router A or only from Router C. If Router A is first in the configuration file, it is used if it is running. If Router A is not running, only then does Switch B accept SA messages from Router C. This is the default behavior without a prefix list.

If you specify a prefix list, the peer is a default peer only for the prefixes in the list. You can have multiple active default peers when you have a prefix list associated with each. When you do not have any prefix lists, you can configure multiple default peers, but only the first one is the active default peer as long as the router has connectivity to this peer and the peer is alive. If the first configured peer fails or the connectivity to this peer fails, the second configured peer becomes the active default, and so on.

The ISP probably uses a prefix list to define which prefixes it accepts from the customer's router.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**ip msdp default-peer** *ip-address* | *name* [**prefix-list** *list*] **Example:** `Router(config)# ip msdp default-peer 10.1.1.1 prefix-list site-a`	Defines a default peer from which to accept all MSDP SA messages. • For *ip-address* | *name*, enter the IP address or Domain Name System (DNS) server name of the MSDP default peer. • (Optional) For **prefix-list** *list*, enter the list name that specifies the peer to be the default peer only for the listed prefixes. You can have multiple active default peers when you have a prefix list associated with each. When you enter multiple **ip msdp default-peer** commands with the **prefix-list** keyword, you use all the default peers at the same time for different RP prefixes. This syntax is typically used in a service provider cloud that connects stub site clouds. When you enter multiple **ip msdp default-peer** commands without the **prefix-list** keyword, a single active peer accepts all SA messages. If that peer fails, the next configured default peer accepts all SA messages. This syntax is typically used at a stub site.
Step 3	**ip prefix-list** *name* [**description** *string*] | **seq** *number* {**permit** | **deny**} *network length* **Example:** `Router(config)# prefix-list site-a seq 3 permit 12 network length 128`	(Optional) Creates a prefix list using the name specified in Step 2. • (Optional) For **description** *string*, enter a description of up to 80 characters to describe this prefix list. • For **seq** *number*, enter the sequence number of the entry. The range is 1 to 4294967294. • The **deny** keyword denies access to matching conditions. • The **permit** keyword permits access to matching conditions. • For *network length*, specify the network number and length (in bits) of the network mask that is permitted or denied.
Step 4	**ip msdp description** {*peer-name* | *peer-address*} *text* **Example:** `Router(config)# ip msdp description peer-name site-b`	(Optional) Configures a description for the specified peer to make it easier to identify in a configuration or in **show** command output. By default, no description is associated with an MSDP peer.
Step 5	**end** **Example:** `Router(config)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 6	**show running-config** **Example:** `Router# show running-config`	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** `Router# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Caching Source-Active State

By default, the switch does not cache source/group pairs from received SA messages. When the switch forwards the MSDP SA information, it does not store it in memory. Therefore, if a member joins a group soon after a SA message is received by the local RP, that member needs to wait until the next SA message to hear about the source. This delay is known as join latency.

If you want to sacrifice some memory in exchange for reducing the latency of the source information, you can configure the switch to cache SA messages.

Beginning in privileged EXEC mode, follow these steps to enable the caching of source/group pairs. This procedure is optional.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip msdp cache-sa-state** [**list** *access-list-number*] **Example:** `Switch(config)# ip msdp cache-sa-state 100`	Enables the caching of source/group pairs (create an SA state). Those pairs that pass the access list are cached. For **list** *access-list-number*, the range is 100 to 199. **Note** An alternative to this command is the **ip msdp sa-reques** global configuration command, which causes the switch to send an SA request message to the MSDP peer when a new member for a group becomes active.

	Command or Action	Purpose
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* **Example:** `Switch(config)#` **`access-list 100 permit ip 171.69.0.0 0.0.255.255 224.2.0.0 0.0.255.255`**	Creates an IP extended access list, repeating the command as many times as necessary. • For *access-list-number*, the range is 100 to 199. Enter the same number created in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *protocol*, enter **ip** as the protocol name. • For *source*, enter the number of the network or host from which the packet is being sent. • For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. • For *destination*, enter the number of the network or host to which the packet is being sent. • For *destination-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the destination. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end** **Example:** `Switch(config)#` **`end`**	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** `Switch#` **`show running-config`**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch#` **`copy running-config startup-config`**	(Optional) Saves your entries in the configuration file.

Requesting Source Information from an MSDP Peer

Local RPs can send SA requests and get immediate responses for all active sources for a given group. By default, the switch does not send any SA request messages to its MSDP peers when a new member joins a group and wants to receive multicast traffic. The new member waits to receive the next periodic SA message.

If you want a new member of a group to learn the active multicast sources in a connected PIM sparse-mode domain that are sending to a group, configure the switch to send SA request messages to the specified MSDP peer when a new member joins a group. The peer replies with the information in its SA cache. If the peer does not have a cache configured, this command has no result. Configuring this feature reduces join latency but sacrifices memory.

This task is optional.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip msdp sa-request** {*ip-address* | *name*} **Example:** `Switch(config)# ip msdp sa-request 171.69.1.1`	Configure the switch to send SA request messages to the specified MSDP peer. For *ip-address* | *name*, enter the IP address or name of the MSDP peer from which the local switch requests SA messages when a new member for a group becomes active. Repeat the command for each MSDP peer that you want to supply with SA messages.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Controlling Source Information that Your Switch Originates

You can control the multicast source information that originates with your switch:

- Sources you advertise (based on your sources)
- Receivers of source information (based on knowing the requestor)

For more information, see the Redistributing Sources, on page 620 and the Filtering Source-Active Request Messages, on page 622.

Redistributing Sources

SA messages originate on RPs to which sources have registered. By default, any source that registers with an RP is advertised. The *A flag* is set in the RP when a source is registered, which means the source is advertised in an SA unless it is filtered.

This task is optional.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip msdp redistribute** [**list** *access-list-name*] [**asn** *aspath-access-list-number*] [**route-map** *map*] **Example:** `Switch(config)# ip msdp redistribute list 21`	Configures which (S,G) entries from the multicast routing table are advertised in SA messages. By default, only sources within the local domain are advertised. • (Optional) **list** *access-list-name*— Enters the name or number of an IP standard or extended access list. The range is 1 to 99 for standard access lists and 100 to 199 for extended lists. The access list controls which local sources are advertised and to which groups they send. • (Optional) **asn** *aspath-access-list-number*—Enters the IP standard or extended access list number in the range 1 to 199. This access list number must also be configured in the **ip as-path access-list** command. • (Optional) **route-map** *map*—Enters the IP standard or extended access list number in the range 1 to 199. This access list number must also be configured in the **ip as-path access-list** command. The switch advertises (S,G) pairs according to the access list or autonomous system path access list.
Step 3	Use one of the following: • **access-list***access-list-number* {**deny**| **permit**}	Creates an IP standard access list, repeating the command as many times as necessary. or

	Command or Action	Purpose

	source [*source-wildcard*]
	
	• **access-list***access-list-number* {**deny**| **permit**} *protocol source source-wildcard destination destination-wildcard*
	
	Example:
	`Switch(config)# access list 21 permit 194.1.22.0`
	
	or
	
	`Switch(config)# access list 21 permit ip 194.1.22.0 1.1.1.1 194.3.44.0 1.1.1.1`
		Creates an IP extended access list, repeating the command as many times as necessary.
		
		• *access-list-number*—Enters the same number created in Step 2. The range is 1 to 99 for standard access lists and 100 to 199 for extended lists.
		• **deny**—Denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched.
		• *protocol*—Enters **ip** as the protocol name.
		• *source*—Enters the number of the network or host from which the packet is being sent.
		• *source-wildcard*—Enters the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore.
		• *destination*—Enters the number of the network or host to which the packet is being sent.
		• *destination-wildcard*—Enters the wildcard bits in dotted decimal notation to be applied to the destination. Place ones in the bit positions that you want to ignore.
		
		Recall that the access list is always terminated by an implicit deny statement for everything.

Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Filtering Source-Active Request Messages

By default, only switches that are caching SA information can respond to SA requests. By default, such a switch honors all SA request messages from its MSDP peers and supplies the IP addresses of the active sources.

However, you can configure the switch to ignore all SA requests from an MSDP peer. You can also honor only those SA request messages from a peer for groups described by a standard access list. If the groups in the access list pass, SA request messages are accepted. All other such messages from the peer for other groups are ignored.

This task is optional.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	Use one of the following: • **ip msdp filter-sa-request** {*ip-address*| *name*} • **ip msdp filter-sa-request** {*ip-address*| *name*} **list** *access-list-number* **Example:** `Switch(config)# ip msdp filter sa-request 171.69.2.2`	Filters all SA request messages from the specified MSDP peer. or Filters SA request messages from the specified MSDP peer for groups that pass the standard access list. The access list describes a multicast group address. The range for the access-list-number is 1 to 99.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] **Example:** `Switch(config)# access-list 1 permit 192.4.22.0 0.0.0.255`	Creates an IP standard access list, repeating the command as many times as necessary. • For *access-list-number*, the range is 1 to 99. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, enter the number of the network or host from which the packet is being sent. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.

	Command or Action	Purpose
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Controlling Source Information that Your Switch Forwards

By default, the switch forwards all SA messages it receives to all its MSDP peers. However, you can prevent outgoing messages from being forwarded to a peer by using a filter or by setting a time-to-live (TTL) value.

Using a Filter

By creating a filter, you can perform one of these actions:

- Filter all source/group pairs
- Specify an IP extended access list to pass only certain source/group pairs
- Filter based on match criteria in a route map

This task is optional.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose

Step 2	Use one of the following:
	• **ip msdp sa-filter out** {*ip-address* | *name*}
	• **ip msdp sa-filter out** {*ip-address* | *name*} **list** *access-list-number*
	• **ip msdp sa-filter out** {*ip-address* | *name*} **route-map** *map-tag*
	
	Example:
	`Switch(config)# ip msdp sa-filter out switch.cisco.com`
	or
	`Switch(config)# ip msdp sa-filter out list 100`
	or
	`Switch(config)# ip msdp sa-filter out switch.cisco.com route-map 22`
		• Filters all SA messages to the specified MSDP peer.
		• Passes only those SA messages that pass the IP extended access list to the specified peer. The range for the extended *access-list-number* is 100 to 199.
		If both the **list** and the **route-map** keywords are used, all conditions must be true to pass any (S,G) pair in outgoing SA messages.
		• Passes only those SA messages that meet the match criteria in the route map *map-tag* to the specified MSDP peer.
		If all match criteria are true, a **permit** from the route map passes routes through the filter. A **deny** filters routes.

Step 3	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* **Example:** `Switch(config)# access list 100 permit ip 194.1.22.0 1.1.1.1 194.3.44.0 1.1.1.1`	(Optional) Creates an IP extended access list, repeating the command as many times as necessary. • For *access-list-number*, enter the number specified in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *protocol*, enter **ip** as the protocol name. • For *source*, enter the number of the network or host from which the packet is being sent. • For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. • For *destination*, enter the number of the network or host to which the packet is being sent. • For *destination-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the destination. Place ones in the bit positions that you want to ignore.

	Command or Action	Purpose
		Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Using TTL to Limit the Multicast Data Sent in SA Messages

You can use a TTL value to control what data is encapsulated in the first SA message for every source. Only multicast packets with an IP-header TTL greater than or equal to the *ttl* argument are sent to the specified MSDP peer. For example, you can limit internal traffic to a TTL of 8. If you want other groups to go to external locations, you must send those packets with a TTL greater than 8.

This task is optional.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip msdp ttl-threshold** {*ip-address* | *name*} *ttl* **Example:** `Switch(config)# ip msdp ttl-threshold switch.cisco.com 0`	Limits which multicast data is encapsulated in the first SA message to the specified MSDP peer. • For *ip-address* | *name*, enter the IP address or name of the MSDP peer to which the TTL limitation applies.

	Command or Action	Purpose
		• For *ttl*, enter the TTL value. The default is 0, which means all multicast data packets are forwarded to the peer until the TTL is exhausted. The range is 0 to 255.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Controlling Source Information that Your Switch Receives

By default, the switch receives all SA messages that its MSDP RPF peers send to it. However, you can control the source information that you receive from MSDP peers by filtering incoming SA messages. In other words, you can configure the switch to not accept them.

You can perform one of these actions:

- Filter all incoming SA messages from an MSDP peer
- Specify an IP extended access list to pass certain source/group pairs
- Filter based on match criteria in a route map

This task is optional.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose

Step 2	Use one of the following:
	• **ip msdp sa-filter in** {*ip-address* | *name*}
	• **ip msdp sa-filter in** {*ip-address* | *name*} **list** *access-list-number*
	• **ip msdp sa-filter in** {*ip-address* | *name*} **route-map** *map-tag*
	
	Example:
	`Switch(config)# ip msdp sa-filter in switch.cisco.com`
	or
	`Switch(config)# ip msdp sa-filter in list 100`
	or
	`Switch(config)# ip msdp sa-filter in switch.cisco.com route-map 22`
		• Filters all SA messages to the specified MSDP peer.
		• Passes only those SA messages from the specified peer that pass the IP extended access list. The range for the extended *access-list-number* is 100 to 199.
		If both the **list** and the **route-map** keywords are used, all conditions must be true to pass any (S,G) pair in outgoing SA messages.
		• Passes only those SA messages from the specified MSDP peer that meet the match criteria in the route map *map-tag*.
		If all match criteria are true, a **permit** from the route map passes routes through the filter. A **deny** filters routes.

Step 3	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* **Example:** `Switch(config)# access list 100 permit ip 194.1.22.0 1.1.1.1 194.3.44.0 1.1.1.1`	(Optional) Creates an IP extended access list, repeating the command as many times as necessary. • *access-list-number*, enter the number specified in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *protocol*, enter **ip** as the protocol name. • For *source*, enter the number of the network or host from which the packet is being sent. • For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. • For *destination*, enter the number of the network or host to which the packet is being sent. • For *destination-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the destination. Place ones in the bit positions that you want to ignore.

	Command or Action	Purpose
		Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring an MSDP Mesh Group

An MSDP mesh group is a group of MSDP speakers that have fully meshed MSDP connectivity among one another. Any SA messages received from a peer in a mesh group are not forwarded to other peers in the same mesh group. Thus, you reduce SA message flooding and simplify peer-RPF flooding. Use the **ip msdp mesh-group** global configuration command when there are multiple RPs within a domain. It is especially used to send SA messages across a domain. You can configure multiple mesh groups (with different names) in a single switch.

This task is optional.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**ip msdp mesh-group** *name* {*ip-address* | *name*} **Example:** `Switch(config)# ip msdp mesh-group 2 switch.cisco.com`	Configures an MSDP mesh group, and specifies the MSDP peer belonging to that mesh group. By default, the MSDP peers do not belong to a mesh group. • For *name*, enter the name of the mesh group. • For *ip-address* | *name*, enter the IP address or name of the MSDP peer to be a member of the mesh group. Repeat this procedure on each MSDP peer in the group.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Shutting Down an MSDP Peer

If you want to configure many MSDP commands for the same peer and you do not want the peer to become active, you can shut down the peer, configure it, and later bring it up. When a peer is shut down, the TCP connection is terminated and is not restarted. You can also shut down an MSDP session without losing configuration information for the peer.

This task is optional.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip msdp shutdown** {*peer-name* | *peer address*} **Example:** `Switch(config)# ip msdp shutdown switch.cisco.com`	Shuts down the specified MSDP peer without losing configuration information. For *peer-name* | *peer address*, enter the IP address or name of the MSDP peer to shut down.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Including a Bordering PIM Dense-Mode Region in MSDP

You can configure MSDP on a switch that borders a PIM sparse-mode region with a dense-mode region. By default, active sources in the dense-mode region do not participate in MSDP.

Note We do not recommend using the **ip msdp border sa-address** global configuration command. It is better to configure the border router in the sparse-mode domain to proxy-register sources in the dense-mode domain to the RP of the sparse-mode domain and have the sparse-mode domain use standard MSDP procedures to advertise these sources.

The **ip msdp originator-id** global configuration command also identifies an interface to be used as the RP address. If both the **ip msdp border sa-address** and the **ip msdp originator-id** global configuration commands are configured, the address derived from the **ip msdp originator-id** command specifies the RP address.

This task is optional.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip msdp border sa-address** *interface-id* **Example:** `Switch(config)# ip msdp border sa-address 0/1`	Configures the switch on the border between a dense-mode and sparse-mode region to send SA messages about active sources in the dense-mode region. For *interface-id*, specifies the interface from which the IP address is derived and used as the RP address in SA messages. The IP address of the interface is used as the Originator-ID, which is the RP field in the SA message.
Step 3	**ip msdp redistribute** [**list** *access-list-name*] [**asn** *aspath-access-list-number*] [**route-map** *map*] **Example:** `Switch(config)# ip msdp redistribute list 100`	Configures which (S,G) entries from the multicast routing table are advertised in SA messages. For more information, see the Redistributing Sources, on page 620.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring an Originating Address other than the RP Address

You can allow an MSDP speaker that originates an SA message to use the IP address of the interface as the RP address in the SA message by changing the Originator ID. You might change the Originator ID in one of these cases:

- If you configure a logical RP on multiple switches in an MSDP mesh group.
- If you have a switch that borders a PIM sparse-mode domain and a dense-mode domain. If a switch borders a dense-mode domain for a site, and sparse-mode is being used externally, you might want dense-mode sources to be known to the outside world. Because this switch is not an RP, it would not have an RP address to use in an SA message. Therefore, this command provides the RP address by specifying the address of the interface.

If both the **ip msdp border sa-address** and the **ip msdp originator-id** global configuration commands are configured, the address derived from the **ip msdp originator-id** command specifies the address of the RP.

This task is optional.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip msdp originator-id** *interface-id* **Example:** `Switch(config)# ip msdp originator-id 0/1`	Configures the RP address in SA messages to be the address of the originating device interface. For *interface-id*, specify the interface on the local switch.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring and Maintaining MSDP

Commands that monitor MSDP SA messages, peers, state, and peer status:

Table 75: Commands for Monitoring and Maintaining MSDP

Command	Purpose
debug ip msdp [*peer-address* | *name*] [**detail**] [**routes**]	Debugs an MSDP activity.
debug ip msdp resets	Debugs MSDP peer reset reasons.
show ip msdp count [*autonomous-system-number*]	Displays the number of sources and groups originated in SA messages from each autonomous system. The **ip msdp cache-sa-state** command must be configured for this command to produce any output.
show ip msdp peer [*peer-address* | *name*]	Displays detailed information about an MSDP peer.
show ip msdp sa-cache [*group-address* | *source-address* | *group-name* | *source-name*] [*autonomous-system-number*]	Displays (S,G) state learned from MSDP peers.
show ip msdp summary	Displays MSDP peer status and SA message counts.

Commands that clear MSDP connections, statistics, and SA cache entries:

Table 76: Commands for Clearing MSDP Connections, Statistics, or SA Cache Entries

Command	Purpose
clear ip msdp peer *peer-address* | *name*	Clears the TCP connection to the specified MSDP peer, resetting all MSDP message counters.
clear ip msdp statistics [*peer-address* | *name*]	Clears statistics counters for one or all the MSDP peers without resetting the sessions.
clear ip msdp sa-cache [*group-address* | *name*]	Clears the SA cache entries for all entries, all sources for a specific group, or all entries for a specific source/group pair.

Configuration Examples for Configuring MSDP

Configuring a Default MSDP Peer: Example

This example shows a partial configuration of Router A and Router C in . Each of these ISPs have more than one customer (like the customer in) who use default peering (no BGP or MBGP). In that case, they might have similar configurations. That is, they accept SAs only from a default peer if the SA is permitted by the corresponding prefix list.

Router A

```
Router(config)# ip msdp default-peer 10.1.1.1
Router(config)# ip msdp default-peer 10.1.1.1 prefix-list site-a
Router(config)# ip prefix-list site-b permit 10.0.0.0/1
```

Router C

```
Router(config)# ip msdp default-peer 10.1.1.1 prefix-list site-a
Router(config)# ip prefix-list site-b permit 10.0.0.0/1
```

Caching Source-Active State: Example

This example shows how to enable the cache state for all sources in 171.69.0.0/16 sending to groups 224.2.0.0/16:

```
Switch(config)# ip msdp cache-sa-state 100
Switch(config)# access-list 100 permit ip 171.69.0.0 0.0.255.255 224.2.0.0 0.0.255.255
```

Requesting Source Information from an MSDP Peer: Example

This example shows how to configure the switch to send SA request messages to the MSDP peer at 171.69.1.1:

```
Switch(config)# ip msdp sa-request 171.69.1.1
```

Controlling Source Information that Your Switch Originates: Example

This example shows how to configure the switch to filter SA request messages from the MSDP peer at 171.69.2.2. SA request messages from sources on network 192.4.22.0 pass access list 1 and are accepted; all others are ignored.

```
Switch(config)# ip msdp filter sa-request 171.69.2.2 list 1
Switch(config)# access-list 1 permit 192.4.22.0 0.0.0.255
```

Controlling Source Information that Your Switch Forwards: Example

This example shows how to allow only (S,G) pairs that pass access list 100 to be forwarded in an SA message to the peer named *switch.cisco.com*:

```
Switch(config)# ip msdp peer switch.cisco.com connect-source gigabitethernet1/0/1
Switch(config)# ip msdp sa-filter out switch.cisco.com list 100
Switch(config)# access-list 100 permit ip 171.69.0.0 0.0.255.255 224.20 0 0.0.255.255
```

Controlling Source Information that Your Switch Receives: Example

This example shows how to filter all SA messages from the peer named *switch.cisco.com*:

```
Switch(config)# ip msdp peer switch.cisco.com connect-source gigabitethernet1/0/1
Switch(config)# ip msdp sa-filter in switch.cisco.com
```

CHAPTER 37

Configuring IP Unicast Routing

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information About Configuring IP Unicast Routing

This module describes how to configure IP Version 4 (IPv4) unicast routing on the switch.

Note On switches running the LAN base feature, static routing on VLANs is supported only with this release.

A switch stack operates and appears as a single router to the rest of the routers in the network. To use advanced routing features and other routing protocols, you must have the IP services feature set enabled on the standalone switch or on the active switch.

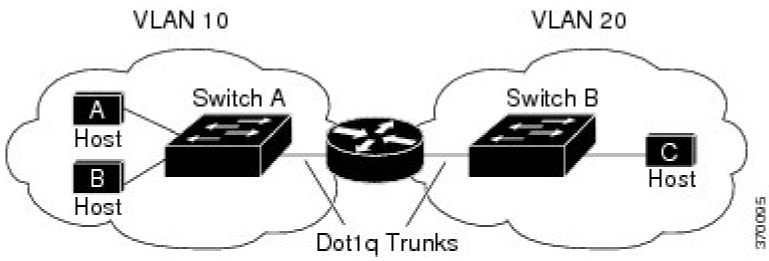
Note In addition to IPv4 traffic, you can also enable IP Version 6 (IPv6) unicast routing and configure interfaces to forward IPv6 traffic if the switch or switch stack is running the IP base or IP services feature set.

Information About IP Routing

In some network environments, VLANs are associated with individual networks or subnetworks. In an IP network, each subnetwork is mapped to an individual VLAN. Configuring VLANs helps control the size of the broadcast domain and keeps local traffic local. However, network devices in different VLANs cannot communicate with one another without a Layer 3 device (router) to route traffic between the VLAN, referred to as inter-VLAN routing. You configure one or more routers to route traffic to the appropriate destination VLAN.

This figure shows a basic routing topology. Switch A is in VLAN 10, and Switch B is in VLAN 20. The router has an interface in each VLAN.

Figure 18: Routing Topology Example



When Host A in VLAN 10 needs to communicate with Host B in VLAN 10, it sends a packet addressed to that host. Switch A forwards the packet directly to Host B, without sending it to the router.

When Host A sends a packet to Host C in VLAN 20, Switch A forwards the packet to the router, which receives the traffic on the VLAN 10 interface. The router checks the routing table, finds the correct outgoing interface, and forwards the packet on the VLAN 20 interface to Switch B. Switch B receives the packet and forwards it to Host C.

Types of Routing

Routers and Layer 3 switches can route packets in these ways:

- By using default routing
- By using preprogrammed static routes for the traffic
- By dynamically calculating routes by using a routing protocol

IP Routing and Switch Stacks

A switch stack appears to the network as a single switch, regardless of which switch in the stack is connected to a routing peer.

The active switch performs these functions:

- The MAC address of the active switch is used as the router MAC address for the whole stack, and all outside devices use this address to send IP packets to the stack.
- All IP packets that require software forwarding or processing go through the CPU of the active switch.

Stack members perform these functions:

- They act as routing standby switches, ready to take over in case they are elected as the new active switch if the active switch fails.
- They program the routes into hardware.

If a active switch fails, the stack detects that the active switch is down and elects one of the stack members to be the new active switch. During this period, except for a momentary interruption, the hardware continues to forward packets with no active protocols.

Upon election, the new active switch performs these functions:

- It builds routing tables, generates the CEF database, and distributes it to stack members.
- It uses its MAC address as the router MAC address. To notify its network peers of the new MAC address, it periodically (every few seconds for 5 minutes) sends a gratuitous ARP reply with the new router MAC address.

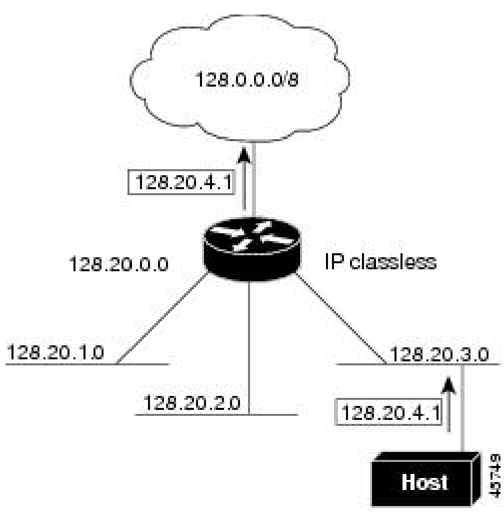
Note If you configure the persistent MAC address feature on the stack and the active switch changes, the stack MAC address does not change for the configured time period. If the previous active switch rejoins the stack as a member switch during that time period, the stack MAC address remains the MAC address of the previous active switch.

Classless Routing

By default, classless routing behavior is enabled on the switch when it is configured to route. With classless routing, if a router receives packets for a subnet of a network with no default route, the router forwards the packet to the best supernet route. A supernet consists of contiguous blocks of Class C address spaces used to simulate a single, larger address space and is designed to relieve the pressure on the rapidly depleting Class B address space.

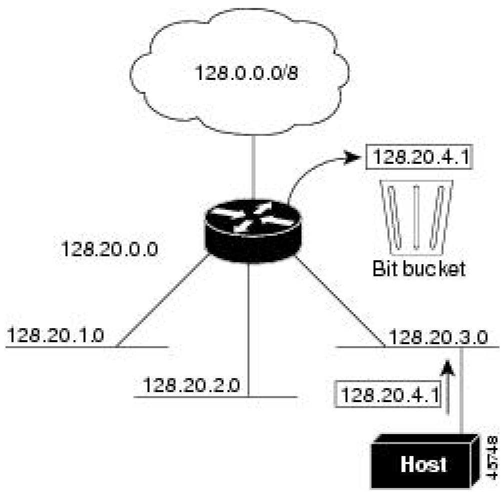
In Figure 41-2, classless routing is enabled. When the host sends a packet to 120.20.4.1, instead of discarding the packet, the router forwards it to the best supernet route. If you disable classless routing and a router receives packets destined for a subnet of a network with no network default route, the router discards the packet.

Figure 19: IP Classless Routing



In Figure 41-3, the router in network 128.20.0.0 is connected to subnets 128.20.1.0, 128.20.2.0, and 128.20.3.0. If the host sends a packet to 120.20.4.1, because there is no network default route, the router discards the packet.

Figure 20: No IP Classless Routing



Address Resolution

You can control interface-specific handling of IP by using address resolution. A device using IP can have both a local address or MAC address, which uniquely defines the device on its local segment or LAN, and a network address, which identifies the network to which the device belongs.

Note In a switch stack, network communication uses a single MAC address and the IP address of the stack.

The local address or MAC address is known as a data link address because it is contained in the data link layer (Layer 2) section of the packet header and is read by data link (Layer 2) devices. To communicate with a device on Ethernet, the software must learn the MAC address of the device. The process of learning the MAC address from an IP address is called address resolution. The process of learning the IP address from the MAC address is called reverse address resolution.

The switch can use these forms of address resolution:

- Address Resolution Protocol (ARP) is used to associate IP address with MAC addresses. Taking an IP address as input, ARP learns the associated MAC address and then stores the IP address/MAC address association in an ARP cache for rapid retrieval. Then the IP datagram is encapsulated in a link-layer frame and sent over the network. Encapsulation of IP datagrams and ARP requests or replies on IEEE 802 networks other than Ethernet is specified by the Subnetwork Access Protocol (SNAP).
- Proxy ARP helps hosts with no routing tables learn the MAC addresses of hosts on other networks or subnets. If the switch (router) receives an ARP request for a host that is not on the same interface as the ARP request sender, and if the router has all of its routes to the host through other interfaces, it generates a proxy ARP packet giving its own local data link address. The host that sent the ARP request then sends its packets to the router, which forwards them to the intended host.

The switch also uses the Reverse Address Resolution Protocol (RARP), which functions the same as ARP does, except that the RARP packets request an IP address instead of a local MAC address. Using RARP requires a RARP server on the same network segment as the router interface. Use the **ip rarp-server** *address* interface configuration command to identify the server.

For more information on RARP, see the *Cisco IOS Configuration Fundamentals Configuration Guide, Release 12.4*.

Proxy ARP

Proxy ARP, the most common method for learning about other routes, enables an Ethernet host with no routing information to communicate with hosts on other networks or subnets. The host assumes that all hosts are on the same local Ethernet and that they can use ARP to learn their MAC addresses. If a switch receives an ARP request for a host that is not on the same network as the sender, the switch evaluates whether it has the best route to that host. If it does, it sends an ARP reply packet with its own Ethernet MAC address, and the host that sent the request sends the packet to the switch, which forwards it to the intended host. Proxy ARP treats all networks as if they are local and performs ARP requests for every IP address.

ICMP Router Discovery Protocol

Router discovery allows the switch to dynamically learn about routes to other networks using ICMP router discovery protocol (IRDP). IRDP allows hosts to locate routers. When operating as a client, the switch generates router discovery packets. When operating as a host, the switch receives router discovery packets. The switch can also listen to Routing Information Protocol (RIP) routing updates and use this information to infer locations of routers. The switch does not actually store the routing tables sent by routing devices; it merely keeps track of which systems are sending the data. The advantage of using IRDP is that it allows each router to specify both a priority and the time after which a device is assumed to be down if no further packets are received.

Each device discovered becomes a candidate for the default router, and a new highest-priority router is selected when a higher priority router is discovered, when the current default router is declared down, or when a TCP connection is about to time out because of excessive retransmissions.

UDP Broadcast Packets and Protocols

User Datagram Protocol (UDP) is an IP host-to-host layer protocol, as is TCP. UDP provides a low-overhead, connectionless session between two end systems and does not provide for acknowledgment of received datagrams. Network hosts occasionally use UDP broadcasts to find address, configuration, and name information. If such a host is on a network segment that does not include a server, UDP broadcasts are normally not forwarded. You can remedy this situation by configuring an interface on a router to forward certain classes of broadcasts to a helper address. You can use more than one helper address per interface.

You can specify a UDP destination port to control which UDP services are forwarded. You can specify multiple UDP protocols. You can also specify the Network Disk (ND) protocol, which is used by older diskless Sun workstations and the network security protocol SDNS.

By default, both UDP and ND forwarding are enabled if a helper address has been defined for an interface. The description for the **ip forward-protocol** interface configuration command in the *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services, Release 12.4* lists the ports that are forwarded by default if you do not specify any UDP ports.

Broadcast Packet Handling

After configuring an IP interface address, you can enable routing and configure one or more routing protocols, or you can configure the way the switch responds to network broadcasts. A broadcast is a data packet destined for all hosts on a physical network. The switch supports two kinds of broadcasting:

- A directed broadcast packet is sent to a specific network or series of networks. A directed broadcast address includes the network or subnet fields.
- A flooded broadcast packet is sent to every network.

> **Note** You can also limit broadcast, unicast, and multicast traffic on Layer 2 interfaces by using the **storm-control** interface configuration command to set traffic suppression levels.

Routers provide some protection from broadcast storms by limiting their extent to the local cable. Bridges (including intelligent bridges), because they are Layer 2 devices, forward broadcasts to all network segments, thus propagating broadcast storms. The best solution to the broadcast storm problem is to use a single broadcast address scheme on a network. In most modern IP implementations, you can set the address to be used as the broadcast address. Many implementations, including the one in the switch, support several addressing schemes for forwarding broadcast messages.

IP Broadcast Flooding

You can allow IP broadcasts to be flooded throughout your internetwork in a controlled fashion by using the database created by the bridging STP. Using this feature also prevents loops. To support this capability, bridging must be configured on each interface that is to participate in the flooding. If bridging is not configured on an interface, it still can receive broadcasts. However, the interface never forwards broadcasts it receives, and the router never uses that interface to send broadcasts received on a different interface.

Packets that are forwarded to a single network address using the IP helper-address mechanism can be flooded. Only one copy of the packet is sent on each network segment.

To be considered for flooding, packets must meet these criteria. (Note that these are the same conditions used to consider packet forwarding using IP helper addresses.)

- The packet must be a MAC-level broadcast.
- The packet must be an IP-level broadcast.
- The packet must be a TFTP, DNS, Time, NetBIOS, ND, or BOOTP packet, or a UDP specified by the **ip forward-protocol udp** global configuration command.
- The time-to-live (TTL) value of the packet must be at least two.

A flooded UDP datagram is given the destination address specified with the **ip broadcast-address** interface configuration command on the output interface. The destination address can be set to any address. Thus, the destination address might change as the datagram propagates through the network. The source address is never changed. The TTL value is decremented.

When a flooded UDP datagram is sent out an interface (and the destination address possibly changed), the datagram is handed to the normal IP output routines and is, therefore, subject to access lists, if they are present on the output interface.

In the switch, the majority of packets are forwarded in hardware; most packets do not go through the switch CPU. For those packets that do go to the CPU, you can speed up spanning tree-based UDP flooding by a factor of about four to five times by using turbo-flooding. This feature is supported over Ethernet interfaces configured for ARP encapsulation.

How to Configure IP Routing

By default, IP routing is disabled on the switch, and you must enable it before routing can take place. For detailed IP routing configuration information, see the *Cisco IOS IP Configuration Guide, Release 12.4*

In the following procedures, the specified interface must be one of these Layer 3 interfaces:

- A routed port: a physical port configured as a Layer 3 port by using the **no switchport** interface configuration command.
- A switch virtual interface (SVI): a VLAN interface created by using the **interface vlan** *vlan_id* global configuration command and by default a Layer 3 interface.
- An EtherChannel port channel in Layer 3 mode: a port-channel logical interface created by using the **interface port-channel** *port-channel-number* global configuration command and binding the Ethernet interface into the channel group. For more information, see the "Configuring Layer 3 EtherChannels" section on page 39-15.

Note The switch does not support tunnel interfaces for unicast routed traffic.

All Layer 3 interfaces on which routing will occur must have IP addresses assigned to them. See the "Assigning IP Addresses to Network Interfaces" section on page 41-7.

Note A Layer 3 switch can have an IP address assigned to each routed port and SVI. The number of routed ports and SVIs that you can configure is limited to 128, exceeding the recommended number and volume of features being implemented might impact CPU utilization because of hardware limitations.

Configuring routing consists of several main procedures:

- To support VLAN interfaces, create and configure VLANs on the switch or switch stack, and assign VLAN membership to Layer 2 interfaces. For more information, see Chapter 14, "Configuring VLANs."
- Configure Layer 3 interfaces.
- Enable IP routing on the switch.
- Assign IP addresses to the Layer 3 interfaces.
- Enable selected routing protocols on the switch.
- Configure routing protocol parameters (optional).

How to Configure IP Addressing

A required task for configuring IP routing is to assign IP addresses to Layer 3 network interfaces to enable the interfaces and allow communication with the hosts on those interfaces that use IP. The following sections describe how to configure various IP addressing features. Assigning IP addresses to the interface is required; the other procedures are optional.

Default IP Addressing Configuration

Table 77: Default Addressing Configuration

Feature	Default Setting
IP address	None defined.
ARP	No permanent entries in the Address Resolution Protocol (ARP) cache. Encapsulation: Standard Ethernet-style ARP. Timeout: 14400 seconds (4 hours).
IP broadcast address	255.255.255.255 (all ones).
IP classless routing	Enabled.
IP default gateway	Disabled.
IP directed broadcast	Disabled (all IP directed broadcasts are dropped).
IP domain	Domain list: No domain names defined. Domain lookup: Enabled. Domain name: Enabled.
IP forward-protocol	If a helper address is defined or User Datagram Protocol (UDP) flooding is configured, UDP forwarding is enabled on default ports. Any-local-broadcast: Disabled. Spanning Tree Protocol (STP): Disabled. Turbo-flood: Disabled.
IP helper address	Disabled.
IP host	Disabled.

Feature	Default Setting
IRDP	Disabled. Defaults when enabled: • Broadcast IRDP advertisements. • Maximum interval between advertisements: 600 seconds. • Minimum interval between advertisements: 0.75 times max interval • Preference: 0.
IP proxy ARP	Enabled.
IP routing	Disabled.
IP subnet-zero	Disabled.

Assigning IP Addresses to Network Interfaces

An IP address identifies a location to which IP packets can be sent. Some IP addresses are reserved for special uses and cannot be used for host, subnet, or network addresses. RFC 1166, "Internet Numbers," contains the official description of IP addresses.

An interface can have one primary IP address. A mask identifies the bits that denote the network number in an IP address. When you use the mask to subnet a network, the mask is referred to as a subnet mask. To receive an assigned network number, contact your Internet service provider.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 3	**no switchport** **Example:** `Switch(config-if)# no switchport`	Removes the interface from Layer 2 configuration mode (if it is a physical interface).

	Command or Action	Purpose
Step 4	**ip address** *ip-address subnet-mask* **Example:** `Switch(config-if)# ip address 10.1.5.1 255.255.255.0`	Configures the IP address and IP subnet mask.
Step 5	**no shutdown** **Example:** `Switch(config-if)# no shutdown`	Enables the physical interface.
Step 6	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 7	**show ip route** **Example:** `Switch# show ip route`	Verifies your entries.
Step 8	**show ip interface** [*interface-id*] **Example:** `Switch# show ip interface gigabitethernet 1/0/1`	Verifies your entries.
Step 9	**show running-config interface** [*interface-id*] **Example:** `Switch# show running-config interface gigabitethernet 1/0/1`	Verifies your entries.
Step 10	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Using Subnet Zero

Subnetting with a subnet address of zero is strongly discouraged because of the problems that can arise if a network and a subnet have the same addresses. For example, if network 131.108.0.0 is subnetted as 255.255.255.0, subnet zero would be written as 131.108.0.0, which is the same as the network address.

You can use the all ones subnet (131.108.255.0) and even though it is discouraged, you can enable the use of subnet zero if you need the entire subnet space for your IP address.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip subnet-zero** **Example:** `Switch(config)# ip subnet-zero`	Enables the use of subnet zero for interface addresses and routing updates.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show running-config** **Example:** `Switch# show running-config`	Verifies your entry.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entry in the configuration file.

Enabling Classless Routing

To prevent the switch from forwarding packets destined for unrecognized subnets to the best supernet route possible, you can disable classless routing behavior.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**no ip classless** **Example:** `Switch(config)#no ip classless`	Disables classless routing behavior.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show running-config** **Example:** `Switch# show running-config`	Verifies your entry.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entry in the configuration file.

Configuring Address Resolution Methods

You can perform the following tasks to configure address resolution.

Defining a Static ARP Cache

ARP and other address resolution protocols provide dynamic mapping between IP addresses and MAC addresses. Because most hosts support dynamic address resolution, you usually do not need to specify static ARP cache entries. If you must define a static ARP cache entry, you can do so globally, which installs a permanent entry in the ARP cache that the switch uses to translate IP addresses into MAC addresses. Optionally, you can also specify that the switch respond to ARP requests as if it were the owner of the specified IP address. If you do not want the ARP entry to be permanent, you can specify a timeout period for the ARP entry.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**arp** *ip-address hardware-address type* **Example:** `Switch(config)# ip 10.1.5.1 c2f3.220a.12f4 arpa`	Associates an IP address with a MAC (hardware) address in the ARP cache, and specifies encapsulation type as one of these: • **arpa—**ARP encapsulation for Ethernet interfaces • **snap—**Subnetwork Address Protocol encapsulation for Token Ring and FDDI interfaces • **sap—**HP's ARP type
Step 3	**arp** *ip-address hardware-address type* [**alias**] **Example:** `Switch(config)# ip 10.1.5.3 d7f3.220d.12f5 arpa alias`	(Optional) Specifies that the switch respond to ARP requests as if it were the owner of the specified IP address.
Step 4	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the interface to configure.
Step 5	**arp** *timeout seconds* **Example:** `Switch(config-if)# arp 20000`	(Optional) Sets the length of time an ARP cache entry will stay in the cache. The default is 14400 seconds (4 hours). The range is 0 to 2147483 seconds.
Step 6	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 7	**show interfaces** [*interface-id*] **Example:** `Switch# show interfaces gigabitethernet 1/0/1`	Verifies the type of ARP and the timeout value used on all interfaces or a specific interface.
Step 8	**show arp** **Example:** `Switch# show arp`	Views the contents of the ARP cache.
Step 9	**show ip arp** **Example:** `Switch# show ip arp`	Views the contents of the ARP cache.

	Command or Action	Purpose
Step 10	**copy running-config startup-config** **Example:** `Switch# copy running-config start-config`	(Optional) Saves your entries in the configuration file.

Setting ARP Encapsulation

By default, Ethernet ARP encapsulation (represented by the **arpa** keyword) is enabled on an IP interface. You can change the encapsulation methods to SNAP if required by your network.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/2`	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 3	**arp** {**arpa** | **snap**} **Example:** `Switch(config-if)# arp arpa`	Specifies the ARP encapsulation method: • **arpa—**Address Resolution Protocol • **snap—**Subnetwork Address Protocol
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show interfaces** [*interface-id*] **Example:** `Switch# show interfaces`	Verifies ARP encapsulation configuration on all interfaces or the specified interface.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Enabling Proxy ARP

By default, the switch uses proxy ARP to help hosts learn MAC addresses of hosts on other networks or subnets.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/2`	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 3	**ip proxy-arp** **Example:** `Switch(config-if)# ip proxy-arp`	Enables proxy ARP on the interface.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show ip interface** [*interface-id*] **Example:** `Switch# show ip interface gigabitethernet 1/0/2`	Verifies the configuration on the interface or all interfaces.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Routing Assistance When IP Routing is Disabled

These mechanisms allow the switch to learn about routes to other networks when it does not have IP routing enabled:

- Proxy ARP
- Default Gateway
- ICMP Router Discovery Protocol (IRDP)

Proxy ARP

Proxy ARP is enabled by default. To enable it after it has been disabled, see the “Enabling Proxy ARP” section. Proxy ARP works as long as other routers support it.

Default Gateway

Another method for locating routes is to define a default router or default gateway. All non-local packets are sent to this router, which either routes them appropriately or sends an IP Control Message Protocol (ICMP) redirect message back, defining which local router the host should use. The switch caches the redirect messages and forwards each packet as efficiently as possible. A limitation of this method is that there is no means of detecting when the default router has gone down or is unavailable.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip default-gateway** *ip-address* **Example:** `Switch(config)# ip default gateway 10.1.5.1`	Sets up a default gateway (router).
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show ip redirects** **Example:** `Switch# show ip redirects`	Displays the address of the default gateway router to verify the setting.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

ICMP Router Discovery Protocol (IRDP)

The only required task for IRDP routing on an interface is to enable IRDP processing on that interface. When enabled, the default parameters apply.

You can optionally change any of these parameters. If you change the **maxadvertinterval** value, the **holdtime** and **minadvertinterval** values also change, so it is important to first change the **maxadvertinterval** value, before manually changing either the **holdtime** or **minadvertinterval** values.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 3	**ip irdp** **Example:** `Switch(config-if)# ip irdp`	Enables IRDP processing on the interface.
Step 4	**ip irdp multicast** **Example:** `Switch(config-if)# ip irdp multicast`	(Optional) Sends IRDP advertisements to the multicast address (224.0.0.1) instead of IP broadcasts. **Note** This command allows for compatibility with Sun Microsystems Solaris, which requires IRDP packets to be sent out as multicasts. Many implementations cannot receive these multicasts; ensure end-host ability before using this command.
Step 5	**ip irdp holdtime** *seconds* **Example:** `Switch(config-if)# ip irdp holdtime 1000`	(Optional) Sets the IRDP period for which advertisements are valid. The default is three times the **maxadvertinterval** value. It must be greater than **maxadvertinterval** and cannot be greater than 9000 seconds. If you change the **maxadvertinterval** value, this value also changes.
Step 6	**ip irdp maxadvertinterval** *seconds* **Example:** `Switch(config-if)# ip irdp maxadvertinterval 650`	(Optional) Sets the IRDP maximum interval between advertisements. The default is 600 seconds.

	Command or Action	Purpose
Step 7	**ip irdp minadvertinterval** *seconds* **Example:** `Switch(config-if)# ip irdp minadvertinterval 500`	(Optional) Sets the IRDP minimum interval between advertisements. The default is 0.75 times the **maxadvertinterval**. If you change the **maxadvertinterval**, this value changes to the new default (0.75 of **maxadvertinterval**).
Step 8	**ip irdp preference** *number* **Example:** `Switch(config-if)# ip irdp preference 2`	(Optional) Sets a device IRDP preference level. The allowed range is –231 to 231. The default is 0. A higher value increases the router preference level.
Step 9	**ip irdp address** *address* [*number*] **Example:** `Switch(config-if)# ip irdp address 10.1.10.10`	(Optional) Specifies an IRDP address and preference to proxy-advertise.
Step 10	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 11	**show ip irdp** **Example:** `Switch# show ip irdp`	Verifies settings by displaying IRDP values.
Step 12	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Broadcast Packet Handling

Perform the tasks in these sections to enable these schemes:

- Enabling Directed Broadcast-to-Physical Broadcast Translation, page 41-19
- Forwarding UDP Broadcast Packets and Protocols, page 41-20
- Establishing an IP Broadcast Address, page 41-22
- Flooding IP Broadcasts, page 41-23

Enabling Directed Broadcast-to-Physical Broadcast Translation

By default, IP directed broadcasts are dropped; they are not forwarded. Dropping IP-directed broadcasts makes routers less susceptible to denial-of-service attacks.

You can enable forwarding of IP-directed broadcasts on an interface where the broadcast becomes a physical (MAC-layer) broadcast. Only those protocols configured by using the **ip forward-protocol** global configuration command are forwarded.

You can specify an access list to control which broadcasts are forwarded. When an access list is specified, only those IP packets permitted by the access list are eligible to be translated from directed broadcasts to physical broadcasts. For more information on access lists, see Chapter 36, "Configuring Network Security with ACLs."

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/2`	Enters interface configuration mode, and specifies the interface to configure.
Step 3	**ip directed-broadcast** [*access-list-number*] **Example:** `Switch(config-if)# ip directed-broadcast 103`	Enables directed broadcast-to-physical broadcast translation on the interface. You can include an access list to control which broadcasts are forwarded. When an access list, only IP packets permitted by the access list can be translated. **Note** The **ip directed-broadcast** interface configuration command can be configured on a VPN routing/forwarding(VRF) interface and is VRF aware. Directed broadcast traffic is routed only within the VRF.
Step 4	**exit** **Example:** `Switch(config-if)# exit`	Returns to global configuration mode.
Step 5	**ip forward-protocol** {**udp** [*port*] | **nd** | **sdns**} **Example:** `Switch(config)# ip forward-protocol nd`	Specifies which protocols and ports the router forwards when forwarding broadcast packets. • **udp—**Forward UPD datagrams. port: (Optional) Destination port that controls which UDP services are forwarded. • **nd—**Forward ND datagrams. • **sdns—**Forward SDNS datagrams

	Command or Action	Purpose
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 7	**show ip interface** [*interface-id*] **Example:** `Switch# show ip interface`	Verifies the configuration on the interface or all interfaces
Step 8	**show running-config** **Example:** `Switch# show running-config`	Verifies the configuration on the interface or all interfaces
Step 9	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Forwarding UDP Broadcast Packets and Protocols

If you do not specify any UDP ports when you configure the forwarding of UDP broadcasts, you are configuring the router to act as a BOOTP forwarding agent. BOOTP packets carry DHCP information.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 3	**ip helper-address** *address* **Example:** `Switch(config-if)# ip helper address 10.1.10.1`	Enables forwarding and specifies the destination address for forwarding UDP broadcast packets, including BOOTP.

	Command or Action	Purpose
Step 4	**exit** **Example:** `Switch(config-if)# exit`	Returns to global configuration mode.
Step 5	**ip forward-protocol** {**udp** [*port*] | **nd** | **sdns**} **Example:** `Switch(config)# ip forward-protocol sdns`	Specifies which protocols the router forwards when forwarding broadcast packets.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 7	**show ip interface** [*interface-id*] **Example:** `Switch# show ip interface gigabitethernet 1/0/1`	Verifies the configuration on the interface or all interfaces.
Step 8	**show running-config** **Example:** `Switch# show running-config`	Verifies the configuration on the interface or all interfaces.
Step 9	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Establishing an IP Broadcast Address

The most popular IP broadcast address (and the default) is an address consisting of all ones (255.255.255.255). However, the switch can be configured to generate any form of IP broadcast address.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the interface to configure.
Step 3	**ip broadcast-address** *ip-address* **Example:** `Switch(config-if)# ip broadcast-address 128.1.255.255`	Enters a broadcast address different from the default, for example 128.1.255.255.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show ip interface** [*interface-id*] **Example:** `Switch# show ip interface`	Verifies the broadcast address on the interface or all interfaces.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Flooding IP Broadcasts

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip forward-protocol spanning-tree** **Example:** `Switch(config)# ip forward-protocol spanning-tree`	Uses the bridging spanning-tree database to flood UDP datagrams.

	Command or Action	Purpose
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show running-config** **Example:** `Switch# show running-config`	Verifies your entry.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entry in the configuration file.
Step 6	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode
Step 7	**ip forward-protocol turbo-flood** **Example:** `Switch(config)# ip forward-protocol turbo-flood`	Uses the spanning-tree database to speed up flooding of UDP datagrams.
Step 8	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 9	**show running-config** **Example:** `Switch# show running-config`	Verifies your entry.
Step 10	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entry in the configuration file.

Monitoring and Maintaining IP Addressing

When the contents of a particular cache, table, or database have become or are suspected to be invalid, you can remove all its contents by using the **clear** privileged EXEC commands. Table 41-2 lists the commands for clearing contents.

Table 78: Commands to Clear Caches, Tables, and Databases

clear arp-cache	Clears the IP ARP cache and the fast-switching cache.
clear host {*name* | *****}	Removes one or all entries from the hostname and the address cache.
clear ip route {**network** [*mask*] | *****}	Removes one or more routes from the IP routing table.

You can display specific statistics, such as the contents of IP routing tables, caches, and databases; the reachability of nodes; and the routing path that packets are taking through the network. Table 41-3 lists the privileged EXEC commands for displaying IP statistics.

Table 79: Commands to Display Caches, Tables, and Databases

show arp	Displays the entries in the ARP table.
show hosts	Displays the default domain name, style of lookup service, name server hosts, and the cached list of hostnames and addresses.
show ip aliases	Displays IP addresses mapped to TCP ports (aliases).
show ip arp	Displays the IP ARP cache.
show ip interface [*interface-id*]	Displays the IP status of interfaces.
show ip irdp	Displays IRDP values.
show ip masks *address*	Displays the masks used for network addresses and the number of subnets using each mask.
show ip redirects	Displays the address of a default gateway.
show ip route [*address* [*mask*]] | [*protocol*]	Displays the current state of the routing table.
show ip route summary	Displays the current state of the routing table in summary form.

How to Configure IP Unicast Routing

Enabling IP Unicast Routing

By default, the switch is in Layer 2 switching mode and IP routing is disabled. To use the Layer 3 capabilities of the switch, you must enable IP routing.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip routing** **Example:** `Switch(config)# ip routing`	Enables IP routing.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Example of Enabling IP Routing

This example shows how to enable IP routing :

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip routing

Switch(config-router)# end
```

What to Do Next

You can now set up parameters for the selected routing protocols as described in these sections:

- RIP
- OSPF,
- EIGRP
- BGP
- Unicast Reverse Path Forwarding
- Protocol-Independent Features (optional)

Information About RIP

The Routing Information Protocol (RIP) is an interior gateway protocol (IGP) created for use in small, homogeneous networks. It is a distance-vector routing protocol that uses broadcast User Datagram Protocol (UDP) data packets to exchange routing information. The protocol is documented in RFC 1058. You can find detailed information about RIP in *IP Routing Fundamentals,* published by Cisco Press.

Note RIP is supported in the IP Base.

Using RIP, the switch sends routing information updates (advertisements) every 30 seconds. If a router does not receive an update from another router for 180 seconds or more, it marks the routes served by that router as unusable. If there is still no update after 240 seconds, the router removes all routing table entries for the non-updating router.

RIP uses hop counts to rate the value of different routes. The hop count is the number of routers that can be traversed in a route. A directly connected network has a hop count of zero; a network with a hop count of 16 is unreachable. This small range (0 to 15) makes RIP unsuitable for large networks.

If the router has a default network path, RIP advertises a route that links the router to the pseudonetwork 0.0.0.0. The 0.0.0.0 network does not exist; it is treated by RIP as a network to implement the default routing feature. The switch advertises the default network if a default was learned by RIP or if the router has a gateway of last resort and RIP is configured with a default metric. RIP sends updates to the interfaces in specified networks. If an interface's network is not specified, it is not advertised in any RIP update.

Summary Addresses and Split Horizon

Routers connected to broadcast-type IP networks and using distance-vector routing protocols normally use the split-horizon mechanism to reduce the possibility of routing loops. Split horizon blocks information about routes from being advertised by a router on any interface from which that information originated. This feature usually optimizes communication among multiple routers, especially when links are broken.

How to Configure RIP

Default RIP Configuration

Table 80: Default RIP Configuration

Feature	Default Setting
Auto summary	Enabled.
Default-information originate	Disabled.
Default metric	Built-in; automatic metric translations.
IP RIP authentication key-chain	No authentication. Authentication mode: clear text.
IP RIP triggered	Disabled
IP split horizon	Varies with media.
Neighbor	None defined.
Network	None specified.
Offset list	Disabled.
Output delay	0 milliseconds.
Timers basic	• Update: 30 seconds. • Invalid: 180 seconds. • Hold-down: 180 seconds. • Flush: 240 seconds.
Validate-update-source	Enabled.
Version	Receives RIP Version 1 and 2 packets; sends Version 1 packets.

Configuring Basic RIP Parameters

To configure RIP, you enable RIP routing for a network and optionally configure other parameters. On the switches, RIP configuration commands are ignored until you configure the network number.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip routing** **Example:** `Switch(config)# ip routing`	Enables IP routing. (Required only if IP routing is disabled.)
Step 3	**router rip** **Example:** `Switch(config)# router rip`	Enables a RIP routing process, and enter router configuration mode.
Step 4	**network** *network number* **Example:** `Switch(config)# network 12`	Associates a network with a RIP routing process. You can specify multiple **network** commands. RIP routing updates are sent and received through interfaces only on these networks. **Note** You must configure a network number for the RIP commands to take effect.
Step 5	**neighbor** *ip-address* **Example:** `Switch(config)# neighbor 10.2.5.1`	(Optional) Defines a neighboring router with which to exchange routing information. This step allows routing updates from RIP (normally a broadcast protocol) to reach nonbroadcast networks.
Step 6	**offset-list** [*access-list number* | *name*] {**in** | **out**} *offset* [*type number*] **Example:** `Switch(config)# offset-list 103 in 10`	(Optional) Applies an offset list to routing metrics to increase incoming and outgoing metrics to routes learned through RIP. You can limit the offset list with an access list or an interface.
Step 7	**timers basic** *update invalid holddown flush* **Example:** `Switch(config)# timers basic 45 360 400 300`	(Optional) Adjusts routing protocol timers. Valid ranges for all timers are 0 to 4294967295 seconds. • *update*—The time between sending routing updates. The default is 30 seconds. • *invalid*—The timer after which a route is declared invalid. The default is 180 seconds. • *holddown*—The time before a route is removed from the routing table. The default is 180 seconds. • *flush*—The amount of time for which routing updates are postponed. The default is 240 seconds.

	Command or Action	Purpose
Step 8	**version** {**1** | **2**} **Example:** `Switch(config)# version 2`	(Optional) Configures the switch to receive and send only RIP Version 1 or RIP Version 2 packets. By default, the switch receives Version 1 and 2 but sends only Version 1. You can also use the interface commands **ip rip** {**send** | **receive**} **version 1** | **2** | **1 2**} to control what versions are used for sending and receiving on interfaces.
Step 9	**no auto summary** **Example:** `Switch(config)# no auto summary`	(Optional) Disables automatic summarization. By default, the switch summarizes subprefixes when crossing classful network boundaries. Disable summarization (RIP Version 2 only) to advertise subnet and host routing information to classful network boundaries.
Step 10	**no validate-update-source** **Example:** `Switch(config)# no validdate-update-source`	(Optional) Disables validation of the source IP address of incoming RIP routing updates. By default, the switch validates the source IP address of incoming RIP routing updates and discards the update if the source address is not valid. Under normal circumstances, disabling this feature is not recommended. However, if you have a router that is off-network and you want to receive its updates, you can use this command.
Step 11	**output-delay** *delay* **Example:** `Switch(config)# output-delay 8`	(Optional) Adds interpacket delay for RIP updates sent. By default, packets in a multiple-packet RIP update have no delay added between packets. If you are sending packets to a lower-speed device, you can add an interpacket delay in the range of 8 to 50 milliseconds.
Step 12	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 13	**show ip protocols** **Example:** `Switch# show ip protocols`	Verifies your entries.
Step 14	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring RIP Authentication

RIP Version 1 does not support authentication. If you are sending and receiving RIP Version 2 packets, you can enable RIP authentication on an interface. The key chain specifies the set of keys that can be used on the interface. If a key chain is not configured, no authentication is performed.

The switch supports two modes of authentication on interfaces for which RIP authentication is enabled: plain text and MD5. The default is plain text.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the interface to configure.
Step 3	**ip rip authentication key-chain** *name-of-chain* **Example:** `Switch(config-if)# ip rip authentication key-chain trees`	Enables RIP authentication.
Step 4	**ip rip authentication mode** {**text** | **md5**} **Example:** `Switch(config-if)# ip rip authentication mode md5`	Configures the interface to use plain text authentication (the default) or MD5 digest authentication.
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**show running-config interface** [*interface-id*] **Example:** `Switch# show running-config`	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Summary Addresses and Split Horizon

Note In general, disabling split horizon is not recommended unless you are certain that your application requires it to properly advertise routes.

If you want to configure an interface running RIP to advertise a summarized local IP address pool on a network access server for dial-up clients, use the **ip summary-address rip** interface configuration command.

Note If split horizon is enabled, neither autosummary nor interface IP summary addresses are advertised.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 3	**ip address** *ip-address subnet-mask* **Example:** `Switch(config-if)# ip address 10.1.1.10 255.255.255.0`	Configures the IP address and IP subnet.
Step 4	**ip summary-address rip ip address** *ip-network mask* **Example:** `Switch(config-if)# ip summary-address rip ip address 10.1.1.30 255.255.255.0`	Configures the IP address to be summarized and the IP network mask.
Step 5	**no ip split horizon** **Example:** `Switch(config-if)# no ip split horizon`	Disables split horizon on the interface.
Step 6	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 7	**show ip interface** *interface-id* **Example:** `Switch# show ip interface gigabitethernet 1/0/1`	Verifies your entries.
Step 8	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Split Horizon

Routers connected to broadcast-type IP networks and using distance-vector routing protocols normally use the split-horizon mechanism to reduce the possibility of routing loops. Split horizon blocks information about routes from being advertised by a router on any interface from which that information originated. This feature can optimize communication among multiple routers, especially when links are broken.

Note In general, we do not recommend disabling split horizon unless you are certain that your application requires it to properly advertise routes.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the interface to configure.
Step 3	**ip address** *ip-address subnet-mask* **Example:** `Switch(config-if)# ip address 10.1.1.10 255.255.255.0`	Configures the IP address and IP subnet.

	Command or Action	Purpose
Step 4	**no ip split-horizon** **Example:** `Switch(config-if)# no ip split-horizon`	Disables split horizon on the interface.
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**show ip interface** *interface-id* **Example:** `Switch# show ip interface gigabitethernet 1/0/1`	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuration Example for Summary Addresses and Split Horizon

In this example, the major net is 10.0.0.0. The summary address 10.2.0.0 overrides the autosummary address of 10.0.0.0 so that 10.2.0.0 is advertised out interface Gigabit Ethernet port 2, and 10.0.0.0 is not advertised. In the example, if the interface is still in Layer 2 mode (the default), you must enter a **no switchport** interface configuration command before entering the **ip address** interface configuration command.

Note If split horizon is enabled, neither autosummary nor interface summary addresses (those configured with the **ip summary-address rip** router configuration command) are advertised.

```
Switch(config)# router rip
Switch(config-router)# interface gigabitethernet1/0/2
Switch(config-if)# ip address 10.1.5.1 255.255.255.0
Switch(config-if)# ip summary-address rip 10.2.0.0 255.255.0.0
Switch(config-if)# no ip split-horizon
Switch(config-if)# exit
Switch(config)# router rip
Switch(config-router)# network 10.0.0.0
Switch(config-router)# neighbor 2.2.2.2 peer-group mygroup
Switch(config-router)# end
```

Information About OSPF

OSPF is an Interior Gateway Protocol (IGP) designed expressly for IP networks, supporting IP subnetting and tagging of externally derived routing information. OSPF also allows packet authentication and uses IP multicast when sending and receiving packets. The Cisco implementation supports RFC 1253, OSPF management information base (MIB).

Note OSPF is supported in IP Base.

The Cisco implementation conforms to the OSPF Version 2 specifications with these key features:

- Definition of stub areas is supported.
- Routes learned through any IP routing protocol can be redistributed into another IP routing protocol. At the intradomain level, this means that OSPF can import routes learned through EIGRP and RIP. OSPF routes can also be exported into RIP.
- Plain text and MD5 authentication among neighboring routers within an area is supported.
- Configurable routing interface parameters include interface output cost, retransmission interval, interface transmit delay, router priority, router dead and hello intervals, and authentication key.
- Virtual links are supported.
- Not-so-stubby-areas (NSSAs) per RFC 1587are supported.

OSPF typically requires coordination among many internal routers, area border routers (ABRs) connected to multiple areas, and autonomous system boundary routers (ASBRs). The minimum configuration would use all default parameter values, no authentication, and interfaces assigned to areas. If you customize your environment, you must ensure coordinated configuration of all routers.

OSPF Nonstop Forwarding

The switch or switch stack supports two levels of nonstop forwarding (NSF):

- OSPF NSF Awareness, on page 671
- OSPF NSF Capability, on page 672

OSPF NSF Awareness

The IP-services feature set supports OSPF NSF Awareness supported for IPv4. When the neighboring router is NSF-capable, the Layer 3 switch continues to forward packets from the neighboring router during the interval between the primary Route Processor (RP) in a router crashing and the backup RP taking over, or while the primary RP is manually reloaded for a non-disruptive software upgrade.

This feature cannot be disabled.

OSPF NSF Capability

The IP services feature set supports the OSPFv2 NSF IETF format in addition to the OSPFv2 NSF Cisco format that is supported in earlier releases. For information about this feature, see *NSF—OSPF (RFC 3623 OSPF Graceful Restart):* http://www.cisco.com/en/US/docs/ios/ha/configuration/guide/ha-ospf_grrs.html#wp1055692.

The IP-services feature set also supports OSPF NSF-capable routing for IPv4 for better convergence and lower traffic loss following a stack master change. When a stack master change occurs in an OSPF NSF-capable stack, the new stack master must do two things to resynchronize its link-state database with its OSFP neighbors:

- Release the available OSPF neighbors on the network without resetting the neighbor relationship.
- Reacquire the contents of the link-state database for the network.

After a stack master change, the new master sends an OSPF NSF signal to neighboring NSF-aware devices. A device recognizes this signal to mean that it should not reset the neighbor relationship with the stack. As the NSF-capable stack master receives signals from other routes on the network, it begins to rebuild its neighbor list.

When the neighbor relationships are reestablished, the NSF-capable stack master resynchronizes its database with its NSF-aware neighbors, and routing information is exchanged between the OSPF neighbors. The new stack master uses this routing information to remove stale routes, to update the routing information database (RIB), and to update the forwarding information base (FIB) with the new information. The OSPF protocols then fully converge.

Note OSPF NSF requires that all neighbor networking devices be NSF-aware. If an NSF-capable router discovers non-NSF aware neighbors on a network segment, it disables NSF capabilities for that segment. Other network segments where all devices are NSF-aware or NSF-capable continue to provide NSF capabilities.

Use the **nsf** OSPF routing configuration command to enable OSPF NSF routing. Use the **show ip ospf** privileged EXEC command to verify that it is enabled.

For more information, see *Cisco Nonstop Forwarding*: http://www.cisco.com/en/US/docs/ios/ha/configuration/guide/ha-nonstp_fwdg.html

OSPF Area Parameters

You can optionally configure several OSPF area parameters. These parameters include authentication for password-based protection against unauthorized access to an area, stub areas, and not-so-stubby-areas (NSSAs). Stub areas are areas into which information on external routes is not sent. Instead, the area border router (ABR) generates a default external route into the stub area for destinations outside the autonomous system (AS). An NSSA does not flood all LSAs from the core into the area, but can import AS external routes within the area by redistribution.

Route summarization is the consolidation of advertised addresses into a single summary route to be advertised by other areas. If network numbers are contiguous, you can use the **area range** router configuration command to configure the ABR to advertise a summary route that covers all networks in the range.

Other OSPF Parameters

You can optionally configure other OSPF parameters in router configuration mode.

- Route summarization: When redistributing routes from other protocols as described in the "Using Route Maps to Redistribute Routing Information" section on page 41-124, each route is advertised individually in an external LSA. To help decrease the size of the OSPF link state database, you can use the **summary-address** router configuration command to advertise a single router for all the redistributed routes included in a specified network address and mask.
- Virtual links: In OSPF, all areas must be connected to a backbone area. You can establish a virtual link in case of a backbone-continuity break by configuring two Area Border Routers as endpoints of a virtual link. Configuration information includes the identity of the other virtual endpoint (the other ABR) and the nonbackbone link that the two routers have in common (the transit area). Virtual links cannot be configured through a stub area.
- Default route: When you specifically configure redistribution of routes into an OSPF routing domain, the route automatically becomes an autonomous system boundary router (ASBR). You can force the ASBR to generate a default route into the OSPF routing domain.
- Domain Name Server (DNS) names for use in all OSPF **show** privileged EXEC command displays makes it easier to identify a router than displaying it by router ID or neighbor ID.
- Default Metrics: OSPF calculates the OSPF metric for an interface according to the bandwidth of the interface. The metric is calculated as *ref-bw* divided by bandwidth, where *ref* is 10 by default, and bandwidth (*bw*) is specified by the **bandwidth** interface configuration command. For multiple links with high bandwidth, you can specify a larger number to differentiate the cost on those links.
- Administrative distance is a rating of the trustworthiness of a routing information source, an integer between 0 and 255, with a higher value meaning a lower trust rating. An administrative distance of 255 means the routing information source cannot be trusted at all and should be ignored. OSPF uses three different administrative distances: routes within an area (interarea), routes to another area (interarea), and routes from another routing domain learned through redistribution (external). You can change any of the distance values.
- Passive interfaces: Because interfaces between two devices on an Ethernet represent only one network segment, to prevent OSPF from sending hello packets for the sending interface, you must configure the sending device to be a passive interface. Both devices can identify each other through the hello packet for the receiving interface.
- Route calculation timers: You can configure the delay time between when OSPF receives a topology change and when it starts the shortest path first (SPF) calculation and the hold time between two SPF calculations.
- Log neighbor changes: You can configure the router to send a syslog message when an OSPF neighbor state changes, providing a high-level view of changes in the router.

LSA Group Pacing

The OSPF LSA group pacing feature allows the router to group OSPF LSAs and pace the refreshing, check-summing, and aging functions for more efficient router use. This feature is enabled by default with a 4-minute default pacing interval, and you will not usually need to modify this parameter. The optimum group pacing interval is inversely proportional to the number of LSAs the router is refreshing, check-summing, and aging. For example, if you have approximately 10,000 LSAs in the database, decreasing the pacing interval would benefit you. If you have a very small database (40 to 100 LSAs), increasing the pacing interval to 10 to 20 minutes might benefit you slightly.

Loopback Interfaces

OSPF uses the highest IP address configured on the interfaces as its router ID. If this interface is down or removed, the OSPF process must recalculate a new router ID and resend all its routing information out its interfaces. If a loopback interface is configured with an IP address, OSPF uses this IP address as its router ID, even if other interfaces have higher IP addresses. Because loopback interfaces never fail, this provides greater stability. OSPF automatically prefers a loopback interface over other interfaces, and it chooses the highest IP address among all loopback interfaces.

How to Configure OSPF

Default OSPF Configuration

Table 81: Default OSPF Configuration

Feature	Default Setting
Interface parameters	Cost: 1. Retransmit interval: 5 seconds. Transmit delay: 1 second. Priority: 1. Hello interval: 10 seconds. Dead interval: 4 times the hello interval. No authentication. No password specified. MD5 authentication disabled.
Area	Authentication type: 0 (no authentication). Default cost: 1. Range: Disabled. Stub: No stub area defined. NSSA: No NSSA area defined.
Auto cost	100 Mb/s.
Default-information originate	Disabled. When enabled, the default metric setting is 10, and the external route type default is Type 2.
Default metric	Built-in, automatic metric translation, as appropriate for each routing protocol.

Feature	Default Setting
Distance OSPF	dist1 (all routes within an area): 110. dist2 (all routes from one area to another): 110. and dist3 (routes from other routing domains): 110.
OSPF database filter	Disabled. All outgoing link-state advertisements (LSAs) are flooded to the interface.
IP OSPF name lookup	Disabled.
Log adjacency changes	Enabled.
Neighbor	None specified.
Neighbor database filter	Disabled. All outgoing LSAs are flooded to the neighbor.
Network area	Disabled.
Nonstop Forwarding (NSF) awareness	Enabled. Allows Layer 3 switches to continue forwarding packets from a neighboring NSF-capable router during hardware or software changes.
NSF capability	Disabled. **Note** The switch stack supports OSPF NSF-capable routing for IPv4.
Router ID	No OSPF routing process defined.
Summary address	Disabled.
Timers LSA group pacing	240 seconds.
Timers shortest path first (spf)	spf delay: 5 seconds.; spf-holdtime: 10 seconds.
Virtual link	No area ID or router ID defined. Hello interval: 10 seconds. Retransmit interval: 5 seconds. Transmit delay: 1 second. Dead interval: 40 seconds. Authentication key: no key predefined. Message-digest key (MD5): no key predefined.

Configuring Basic OSPF Parameters

To enable OSPF, create an OSPF routing process, specify the range of IP addresses to associate with the routing process, and assign area IDs to be associated with that range. For switches running the IP services image, you can configure either the Cisco OSPFv2 NSF format or the IETF OSPFv2 NSF format.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router ospf** process-id **Example:** `Switch(config)# router ospf 15`	Enables OSPF routing, and enter router configuration mode. The process ID is an internally used identification parameter that is locally assigned and can be any positive integer. Each OSPF routing process has a unique value. **Note** OSPF for Routed Access supports only one OSPFv2 and one OSPFv3 instance with a maximum number of 200 dynamically learned routes.
Step 3	**nsf cisco** [**enforce global**] **Example:** `Switch(config)# nsf cisco enforce global`	(Optional) Enables Cisco NSF operations for OSPF. The **enforce global** keyword cancels NSF restart when non-NSF-aware neighboring networking devices are detected. **Note** Enter the command in Step 3 or Step 4, and go to Step 5.
Step 4	**nsf ietf** [**restart-interval** *seconds*] **Example:** `Switch(config)# nsf ietf restart-interval 60`	(Optional) Enables IETF NSF operations for OSPF. The **restart-interval** keyword specifies the length of the graceful restart interval, in seconds. The range is from 1 to 1800. The default is 120. **Note** Enter the command in Step 3 or Step 4, and go to Step 5.
Step 5	**network** address wildcard-mask **area** area-id **Example:** `Switch(config)# network 10.1.1.1 255.240.0.0 area 20`	Define an interface on which OSPF runs and the area ID for that interface. You can use the wildcard-mask to use a single command to define one or more multiple interfaces to be associated with a specific OSPF area. The area ID can be a decimal value or an IP address.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 7	**show ip protocols** **Example:** `Switch# show ip protocols`	Verifies your entries.
Step 8	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring OSPF Interfaces

You can use the **ip ospf** interface configuration commands to modify interface-specific OSPF parameters. You are not required to modify any of these parameters, but some interface parameters (hello interval, dead interval, and authentication key) must be consistent across all routers in an attached network. If you modify these parameters, be sure all routers in the network have compatible values.

Note The **ip ospf** interface configuration commands are all optional.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** interface-id **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 3	**ip ospf** cost **Example:** `Switch(config-if)# ip ospf 8`	(Optional) Explicitly specifies the cost of sending a packet on the interface.
Step 4	**ip ospf retransmit-interval** seconds **Example:** `Switch(config-if)# ip ospf transmit-interval 10`	(Optional) Specifies the number of seconds between link state advertisement transmissions. The range is 1 to 65535 seconds. The default is 5 seconds.
Step 5	**ip ospf transmit-delay** seconds **Example:** `Switch(config-if)# ip ospf transmit-delay 2`	(Optional) Sets the estimated number of seconds to wait before sending a link state update packet. The range is 1 to 65535 seconds. The default is 1 second.
Step 6	**ip ospf priority** number **Example:** `Switch(config-if)# ip ospf priority 5`	(Optional) Sets priority to help find the OSPF designated router for a network. The range is from 0 to 255. The default is 1.

	Command or Action	Purpose
Step 7	**ip ospf hello-interval** seconds **Example:** `Switch(config-if)# ip ospf hello-interval 12`	(Optional) Sets the number of seconds between hello packets sent on an OSPF interface. The value must be the same for all nodes on a network. The range is 1 to 65535 seconds. The default is 10 seconds.
Step 8	**ip ospf dead-interval** seconds **Example:** `Switch(config-if)# ip ospf dead-interval 8`	(Optional) Sets the number of seconds after the last device hello packet was seen before its neighbors declare the OSPF router to be down. The value must be the same for all nodes on a network. The range is 1 to 65535 seconds. The default is 4 times the hello interval.
Step 9	**ip ospf authentication-key** key **Example:** `Switch(config-if)# ip ospf authentication-key password`	(Optional) Assign a password to be used by neighboring OSPF routers. The password can be any string of keyboard-entered characters up to 8 bytes in length. All neighboring routers on the same network must have the same password to exchange OSPF information.
Step 10	**ip ospf message digest-key** keyid **md5** key **Example:** `Switch(config-if)# ip ospf message digest-key 16 md5 your1pass`	(Optional) Enables MDS authentication. • *keyid*—An identifier from 1 to 255. • *key*—An alphanumeric password of up to 16 bytes.
Step 11	**ip ospf database-filter all out** **Example:** `Switch(config-if)# ip ospf database-filter all out`	(Optional) Block flooding of OSPF LSA packets to the interface. By default, OSPF floods new LSAs over all interfaces in the same area, except the interface on which the LSA arrives.
Step 12	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 13	**show ip ospf interface** [*interface-name*] **Example:** `Switch# show ip ospf interface`	Displays OSPF-related interface information.
Step 14	**show ip ospf neighbor detail** **Example:** `Switch# show ip ospf neighbor detail`	Displays NSF awareness status of neighbor switch. The output matches one of these examples: • *Options is 0x52* *LLS Options is 0x1 (LR)* When both of these lines appear, the neighbor switch is NSF aware.

	Command or Action	Purpose
		• *Options is 0x42*—This means the neighbor switch is not NSF aware.
Step 15	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring OSPF Area Parameters

Before You Begin

Note The OSPF **area** router configuration commands are all optional.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router ospf** process-id **Example:** `Switch(config)# router ospf 109`	Enables OSPF routing, and enter router configuration mode.
Step 3	**area** *area-id* **authentication** **Example:** `Switch(config-router)# area 1 authentication`	(Optional) Allow password-based protection against unauthorized access to the identified area. The identifier can be either a decimal value or an IP address.
Step 4	**area** *area-id* **authentication message-digest** **Example:** `Switch(config-router)# area 1 authentication message-digest`	(Optional) Enables MD5 authentication on the area.

	Command or Action	Purpose
Step 5	**area** *area-id* **stub** [**no-summary**] **Example:** `Switch(config-router)# area 1 stub`	(Optional) Define an area as a stub area. The **no-summary** keyword prevents an ABR from sending summary link advertisements into the stub area.
Step 6	**area** *area-id* **nssa** [**no-redistribution**] [**default-information-originate**] [**no-summary**] **Example:** `Switch(config-router)# area 1 nssa default-information-originate`	(Optional) Defines an area as a not-so-stubby-area. Every router within the same area must agree that the area is NSSA. Select one of these keywords: • **no-redistribution—**Select when the router is an NSSA ABR and you want the **redistribute** command to import routes into normal areas, but not into the NSSA. • **default-information-originate—**Select on an ABR to allow importing type 7 LSAs into the NSSA. • **no-redistribution—**Select to not send summary LSAs into the NSSA.
Step 7	**area** *area-id* **range** *address mask* **Example:** `Switch(config-router)# area 1 range 255.240.0.0`	(Optional) Specifies an address range for which a single route is advertised. Use this command only with area border routers.
Step 8	**end** **Example:** `Switch(config-router)# end`	Returns to privileged EXEC mode.
Step 9	**show ip ospf** [*process-id*] **Example:** `Switch# show ip ospf`	Displays information about the OSPF routing process in general or for a specific process ID to verify configuration.
Step 10	**show ip ospf** [*process-id* [*area-id*]] **database** **Example:** `Switch# show ip osfp database`	Displays lists of information related to the OSPF database for a specific router.
Step 11	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Other OSPF Parameters

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router ospf** *process-id* **Example:** `Switch(config)# router ospf 10`	Enables OSPF routing, and enter router configuration mode.
Step 3	**summary-address** *address mask* **Example:** `Switch(config)# summary-address 10.1.1.1 255.255.255.0`	(Optional) Specifies an address and IP subnet mask for redistributed routes so that only one summary route is advertised.
Step 4	**area** *area-id* **virtual-link** *router-id* [**hello-interval** *seconds*] [**retransmit-interval** *seconds*] [**trans**] [[**authentication-key** *key*] | **message-digest-key** *keyid* **md5** *key*]] **Example:** `Switch(config)# area 2 virtual-link 192.168.255.1 hello-interval 5`	(Optional) Establishes a virtual link and set its parameters. See the "Configuring OSPF Interfaces" section on page 41-39 for parameter definitions and Table 41-5 on page 41-35 for virtual link defaults.
Step 5	**default-information originate** [**always**] [**metric** *metric-value*] [**metric-type** *type-value*] [**route-map** *map-name*] **Example:** `Switch(config)# default-information originate metric 100 metric-type 1`	(Optional) Forces the ASBR to generate a default route into the OSPF routing domain. Parameters are all optional.
Step 6	**ip ospf name-lookup** **Example:** `Switch(config)# ip ospf name-lookup`	(Optional) Configures DNS name lookup. The default is disabled.
Step 7	**ip auto-cost reference-bandwidth** *ref-bw* **Example:** `Switch(config)# ip auto-cost reference-bandwidth 5`	(Optional) Specifies an address range for which a single route will be advertised. Use this command only with area border routers.

	Command or Action	Purpose
Step 8	**distance ospf** {[**inter-area** *dist1*] [**inter-area** *dist2*] [**external** *dist3*]} **Example:** `Switch(config)# distance ospf inter-area 150`	(Optional) Changes the OSPF distance values. The default distance for each type of route is 110. The range is 1 to 255.
Step 9	**passive-interface** *type number* **Example:** `Switch(config)# passive-interface gigabitethernet 1/0/6`	(Optional) Suppresses the sending of hello packets through the specified interface.
Step 10	**timers throttle spf** *spf-delay spf-holdtime spf-wait* **Example:** `Switch(config)# timers throttle spf 200 100 100`	(Optional) Configures route calculation timers. • *spf-delay*—Delay between receiving a change to SPF calculation. The range is from 1 to 600000 miliseconds. • *spf-holdtime*—Delay between first and second SPF calculation. The range is form 1 to 600000 in milliseconds. • *spf-wait*—Maximum wait time in milliseconds for SPF calculations. The range is from 1 to 600000 in milliseconds.
Step 11	**ospf log-adj-changes** **Example:** `Switch(config)# ospf log-adj-changes`	(Optional) Sends syslog message when a neighbor state changes.
Step 12	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 13	**show ip ospf** [*process-id* [*area-id*]] **database** **Example:** `Switch# show ip ospf database`	Displays lists of information related to the OSPF database for a specific router. For some of the keyword options, see the "Monitoring OSPF" section on page 41-47.
Step 14	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Changing LSA Group Pacing

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router ospf** *process-id* **Example:** `Switch(config)# router ospf 25`	Enables OSPF routing, and enter router configuration mode.
Step 3	**timers lsa-group-pacing** *seconds* **Example:** `Switch(config-router)# timers lsa-group-pacing 15`	Changes the group pacing of LSAs.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring a Loopback Interface

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface loopback 0** **Example:** `Switch(config)# interface loopback 0`	Creates a loopback interface, and enter interface configuration mode.
Step 3	**ip address** address mask **Example:** `Switch(config-if)# ip address 10.1.1.5 255.255.240.0`	Assign an IP address to this interface.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show ip interface** **Example:** `Switch# show ip interface`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring OSPF

You can display specific statistics such as the contents of IP routing tables, caches, and databases.

Table 82: Show IP OSPF Statistics Commands

show ip ospf [*process-id*]	Displays general information about OSPF routing processes.

show ip ospf [*process-id*] **database** [**router**] [*link-state-id*] **show ip ospf** [*process-id*] **database** [**router**] [**self-originate**] **show ip ospf** [*process-id*] **database** [**router**] [**adv-router** [*ip-address*]] **show ip ospf** [*process-id*] **database** [**network**] [*link-state-id*] **show ip ospf** [*process-id*] **database** [**summary**] [*link-state-id*] **show ip ospf** [*process-id*] **database** [**asbr-summary**] [*link-state-id*] **show ip ospf** [*process-id*] **database** [**external**] [*link-state-id*] **show ip ospf** [*process-id area-id*] **database** [**database-summary**]	Displays lists of information related to the OSPF database.
show ip ospf border-routes	Displays the internal OSPF routing ABR and ASBR table entries.
show ip ospf interface [*interface-name*]	Displays OSPF-related interface information.
show ip ospf neighbor [*interface-name*] [*neighbor-id*] **detail**	Displays OSPF interface neighbor information.
show ip ospf virtual-links	Displays OSPF-related virtual links information.

Configuration Examples for OSPF

Example: Configuring Basic OSPF Parameters

This example shows how to configure an OSPF routing process and assign it a process number of 109:

```
Switch(config)# router ospf 109
Switch(config-router)# network 131.108.0.0 255.255.255.0 area 24
```

Information About EIGRP

Enhanced IGRP (EIGRP) is a Cisco proprietary enhanced version of the IGRP. EIGRP uses the same distance vector algorithm and distance information as IGRP; however, the convergence properties and the operating efficiency of EIGRP are significantly improved.

The convergence technology employs an algorithm referred to as the Diffusing Update Algorithm (DUAL), which guarantees loop-free operation at every instant throughout a route computation and allows all devices involved in a topology change to synchronize at the same time. Routers that are not affected by topology changes are not involved in recomputations.

IP EIGRP provides increased network width. With RIP, the largest possible width of your network is 15 hops. Because the EIGRP metric is large enough to support thousands of hops, the only barrier to expanding the network is the transport-layer hop counter. EIGRP increments the transport control field only when an IP packet has traversed 15 routers and the next hop to the destination was learned through EIGRP. When a RIP route is used as the next hop to the destination, the transport control field is incremented as usual.

EIGRP Features

EIGRP offers these features:

- Fast convergence.
- Incremental updates when the state of a destination changes, instead of sending the entire contents of the routing table, minimizing the bandwidth required for EIGRP packets.
- Less CPU usage because full update packets need not be processed each time they are received.
- Protocol-independent neighbor discovery mechanism to learn about neighboring routers.
- Variable-length subnet masks (VLSMs).
- Arbitrary route summarization.
- EIGRP scales to large networks.

EIGRP Components

EIGRP has these four basic components:

- Neighbor discovery and recovery is the process that routers use to dynamically learn of other routers on their directly attached networks. Routers must also discover when their neighbors become unreachable or inoperative. Neighbor discovery and recovery is achieved with low overhead by periodically sending small hello packets. As long as hello packets are received, the Cisco IOS software can learn that a neighbor is alive and functioning. When this status is determined, the neighboring routers can exchange routing information.
- The reliable transport protocol is responsible for guaranteed, ordered delivery of EIGRP packets to all neighbors. It supports intermixed transmission of multicast and unicast packets. Some EIGRP packets must be sent reliably, and others need not be. For efficiency, reliability is provided only when necessary. For example, on a multiaccess network that has multicast capabilities (such as Ethernet), it is not necessary to send hellos reliably to all neighbors individually. Therefore, EIGRP sends a single multicast hello with an indication in the packet informing the receivers that the packet need not be acknowledged. Other types of packets (such as updates) require acknowledgment, which is shown in the packet. The reliable transport has a provision to send multicast packets quickly when there are unacknowledged packets pending. Doing so helps ensure that convergence time remains low in the presence of varying speed links.
- The DUAL finite state machine embodies the decision process for all route computations. It tracks all routes advertised by all neighbors. DUAL uses the distance information (known as a metric) to select efficient, loop-free paths. DUAL selects routes to be inserted into a routing table based on feasible successors. A successor is a neighboring router used for packet forwarding that has a least-cost path to a destination that is guaranteed not to be part of a routing loop. When there are no feasible successors, but there are neighbors advertising the destination, a recomputation must occur. This is the process whereby a new successor is determined. The amount of time it takes to recompute the route affects the

convergence time. Recomputation is processor-intensive; it is advantageous to avoid recomputation if it is not necessary. When a topology change occurs, DUAL tests for feasible successors. If there are feasible successors, it uses any it finds to avoid unnecessary recomputation.

- The protocol-dependent modules are responsible for network layer protocol-specific tasks. An example is the IP EIGRP module, which is responsible for sending and receiving EIGRP packets that are encapsulated in IP. It is also responsible for parsing EIGRP packets and informing DUAL of the new information received. EIGRP asks DUAL to make routing decisions, but the results are stored in the IP routing table. EIGRP is also responsible for redistributing routes learned by other IP routing protocols.

Note To enable EIGRP, the switch or stack master must be running the IP services feature set.

EIGRP Nonstop Forwarding

The switch stack supports two levels of EIGRP nonstop forwarding:

- EIGRP NSF Awareness
- EIGRP NSF Capability

EIGRP NSF Awareness

The IP-services feature set supports EIGRP NSF Awareness for IPv4. When the neighboring router is NSF-capable, the Layer 3 switch continues to forward packets from the neighboring router during the interval between the primary Route Processor (RP) in a router failing and the backup RP taking over, or while the primary RP is manually reloaded for a nondisruptive software upgrade.

This feature cannot be disabled. For more information on this feature, see the “EIGRP Nonstop Forwarding (NSF) Awareness” section of the *Cisco IOS IP Routing Protocols Configuration Guide, Release 12.4*.

EIGRP NSF Capability

The IP services feature set supports EIGRP Cisco NSF routing to speed up convergence and to eliminate traffic loss after a stack master change. For details about this NSF capability, see the “Configuring Nonstop Forwarding” chapter in the *High Availability Configuration Guide, Cisco IOS XE Release 3S*: http://www.cisco.com/en/US/docs/ios/ios_xe/ha/configuration/guide/ha-nonstp_fwdg_xe.html.

The IP-services feature set also supports EIGRP NSF-capable routing for IPv4 for better convergence and lower traffic loss following a stack master change. When an EIGRP NSF-capable stack master restarts or a new stack master starts up and NSF restarts, the switch has no neighbors, and the topology table is empty. The switch must bring up the interfaces, reacquire neighbors, and rebuild the topology and routing tables without interrupting the traffic directed toward the switch stack. EIGRP peer routers maintain the routes learned from the new stack master and continue forwarding traffic through the NSF restart process.

To prevent an adjacency reset by the neighbors, the new stack master uses a new Restart (RS) bit in the EIGRP packet header to show the restart. When the neighbor receives this, it synchronizes the stack in its peer list and maintains the adjacency with the stack. The neighbor then sends its topology table to the stack master with the RS bit set to show that it is NSF-aware and is aiding the new stack master.

If at least one of the stack peer neighbors is NSF-aware, the stack master receives updates and rebuilds its database. Each NSF-aware neighbor sends an end of table (EOT) marker in the last update packet to mark the end of the table content. The stack master recognizes the convergence when it receives the EOT marker, and it then begins sending updates. When the stack master has received all EOT markers from its neighbors or when the NSF converge timer expires, EIGRP notifies the routing information database (RIB) of convergence and floods its topology table to all NSF-aware peers.

EIGRP Stub Routing

The EIGRP stub routing feature, available in all feature sets, reduces resource utilization by moving routed traffic closer to the end user.

Note The IP base feature set contains EIGRP stub routing capability, which only advertises connected or summary routes from the routing tables to other switches in the network. The switch uses EIGRP stub routing at the access layer to eliminate the need for other types of routing advertisements. For enhanced capability and complete EIGRP routing, the switch must be running the IP services feature set. On a switch running the IP base feature set, if you try to configure multi-VRF-CE and EIGRP stub routing at the same time, the configuration is not allowed. IPv6 EIGRP stub routing is not supported with the IP base feature set.

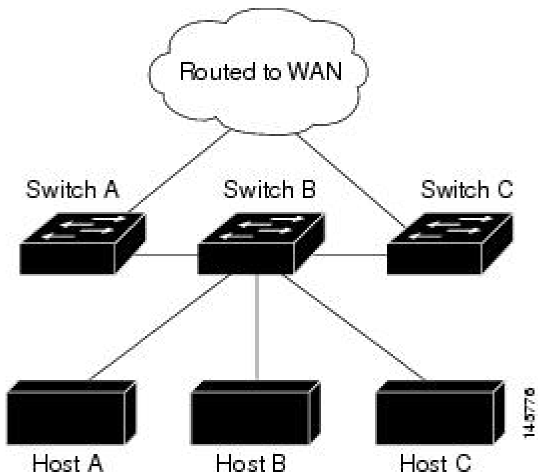
In a network using EIGRP stub routing, the only allowable route for IP traffic to the user is through a switch that is configured with EIGRP stub routing. The switch sends the routed traffic to interfaces that are configured as user interfaces or are connected to other devices.

When using EIGRP stub routing, you need to configure the distribution and remote routers to use EIGRP and to configure only the switch as a stub. Only specified routes are propagated from the switch. The switch responds to all queries for summaries, connected routes, and routing updates.

Any neighbor that receives a packet informing it of the stub status does not query the stub router for any routes, and a router that has a stub peer does not query that peer. The stub router depends on the distribution router to send the proper updates to all peers.

In Figure 41-4, switch B is configured as an EIGRP stub router. Switches A and C are connected to the rest of the WAN. Switch B advertises connected, static, redistribution, and summary routes to switch A and C. Switch B does not advertise any routes learned from switch A (and the reverse).

Figure 21: EIGRP Stub Router Configuration



For more information about EIGRP stub routing, see "Configuring EIGRP Stub Routing" section of the *Cisco IOS IP Configuration Guide, Volume 2 of 3: Routing Protocols, Release 12.4*.

How to Configure EIGRP

To create an EIGRP routing process, you must enable EIGRP and associate networks. EIGRP sends updates to the interfaces in the specified networks. If you do not specify an interface network, it is not advertised in any EIGRP update.

Note If you have routers on your network that are configured for IGRP, and you want to change to EIGRP, you must designate transition routers that have both IGRP and EIGRP configured. In these cases, perform Steps 1 through 3 in the next section and also see the "Configuring Split Horizon" section. You must use the same AS number for routes to be automatically redistributed.

Default EIGRP Configuration

Table 83: Default EIGRP Configuration

Feature	Default Setting
Auto summary	Disabled.
Default-information	Exterior routes are accepted and default information is passed between EIGRP processes when doing redistribution.

Feature	Default Setting
Default metric	Only connected routes and interface static routes can be redistributed without a default metric. The metric includes: • Bandwidth: 0 or greater kb/s. • Delay (tens of microseconds): 0 or any positive number that is a multiple of 39.1 nanoseconds. • Reliability: any number between 0 and 255 (255 means 100 percent reliability). • Loading: effective bandwidth as a number between 0 and 255 (255 is 100 percent loading). • MTU: maximum transmission unit size of the route in bytes. 0 or any positive integer.
Distance	Internal distance: 90. External distance: 170.
EIGRP log-neighbor changes	Disabled. No adjacency changes logged.
IP authentication key-chain	No authentication provided.
IP authentication mode	No authentication provided.
IP bandwidth-percent	50 percent.
IP hello interval	For low-speed nonbroadcast multiaccess (NBMA) networks: 60 seconds; all other networks: 5 seconds.
IP hold-time	For low-speed NBMA networks: 180 seconds; all other networks: 15 seconds.
IP split-horizon	Enabled.
IP summary address	No summary aggregate addresses are predefined.
Metric weights	tos: 0; k1 and k3: 1; k2, k4, and k5: 0
Network	None specified.
Nonstop Forwarding (NSF) Awareness	Enabled for IPv4 on switches running the IP services feature set. Allows Layer 3 switches to continue forwarding packets from a neighboring NSF-capable router during hardware or software changes.
NSF capability	Disabled. **Note** The switch supports EIGRP NSF-capable routing for IPv4.

Feature	Default Setting
Offset-list	Disabled.
Router EIGRP	Disabled.
Set metric	No metric set in the route map.
Traffic-share	Distributed proportionately to the ratios of the metrics.
Variance	1 (equal-cost load-balancing).

Configuring Basic EIGRP Parameters

Before You Begin

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router eigrp** autonomous-system **Example:** `Switch(config)# router eigrp 10`	Enables an EIGRP routing process, and enter router configuration mode. The AS number identifies the routes to other EIGRP routers and is used to tag routing information.
Step 3	**nsf** **Example:** `Switch(config)# nsf`	(Optional) Enables EIGRP NSF. Enter this command on the stack master and on all of its peers.
Step 4	**network** network-number **Example:** `Switch(config)# network 192.168.0.0`	Associate networks with an EIGRP routing process. EIGRP sends updates to the interfaces in the specified networks.
Step 5	**eigrp log-neighbor-changes** **Example:** `Switch(config)# eigrp log-neighbor-changes`	(Optional) Enables logging of EIGRP neighbor changes to monitor routing system stability.

	Command or Action	Purpose
Step 6	**metric weights** *tos k1 k2 k3 k4 k5* **Example:** `Switch(config)# metric weights 0 2 0 2 0 0`	(Optional) Adjust the EIGRP metric. Although the defaults have been carefully set to provide excellent operation in most networks, you can adjust them. Setting metrics is complex and is not recommended without guidance from an experienced network designer.
Step 7	**offset-list** [*access-list number* | *name*] {**in** | **out**} *offset* [*type number*] **Example:** `Switch(config)# offset-list 21 out 10`	(Optional) Applies an offset list to routing metrics to increase incoming and outgoing metrics to routes learned through EIGRP. You can limit the offset list with an access list or an interface.
Step 8	**auto-summary** **Example:** `Switch(config)# auto-summary`	(Optional) Enables automatic summarization of subnet routes into network-level routes.
Step 9	**ip summary-address eigrp** *autonomous-system-number address mask* **Example:** `Switch(config)# ip summary-address eigrp 1 192.168.0.0 255.255.0.0`	(Optional) Configures a summary aggregate.
Step 10	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 11	**show ip protocols** **Example:** `Switch# show ip protocols`	Verifies your entries. For NSF awareness, the output shows: *** IP Routing is NSF aware *** EIGRP NSF enabled
Step 12	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring EIGRP Interfaces

Other optional EIGRP parameters can be configured on an interface basis.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 3	**ip bandwidth-percent eigrp** *percent* **Example:** `Switch(config-if)# ip bandwidth-percent eigrp 60`	(Optional) Configures the percentage of bandwidth that can be used by EIGRP on an interface. The default is 50 percent.
Step 4	**ip summary-address eigrp** *autonomous-system-number address mask* **Example:** `Switch(config-if)# ip summary-address eigrp 109 192.161.0.0 255.255.0.0`	(Optional) Configures a summary aggregate address for a specified interface (not usually necessary if auto-summary is enabled).
Step 5	**ip hello-interval eigrp** *autonomous-system-number seconds* **Example:** `Switch(config-if)# ip hello-interval eigrp 109 10`	(Optional) Change the hello time interval for an EIGRP routing process. The range is 1 to 65535 seconds. The default is 60 seconds for low-speed NBMA networks and 5 seconds for all other networks.
Step 6	**ip hold-time eigrp** *autonomous-system-number seconds* **Example:** `Switch(config-if)# ip hold-time eigrp 109 40`	(Optional) Change the hold time interval for an EIGRP routing process. The range is 1 to 65535 seconds. The default is 180 seconds for low-speed NBMA networks and 15 seconds for all other networks. Do not adjust the hold time without consulting Cisco technical support.
Step 7	**no ip split-horizon eigrp** *autonomous-system-number* **Example:** `Switch(config-if)# no ip split-horizon eigrp 109`	(Optional) Disables split horizon to allow route information to be advertised by a router out any interface from which that information originated.

	Command or Action	Purpose
Step 8	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 9	**show ip eigrp interface** **Example:** `Switch# show ip eigrp interface`	Displays which interfaces EIGRP is active on and information about EIGRP relating to those interfaces.
Step 10	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring EIGRP Route Authentication

EIGRP route authentication provides MD5 authentication of routing updates from the EIGRP routing protocol to prevent the introduction of unauthorized or false routing messages from unapproved sources.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** interface-id **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 3	**ip authentication mode eigrp** *autonomous-system* **md5** **Example:** `Switch(config-if)# ip authentication mode eigrp 104 md5`	Enables MD5 authentication in IP EIGRP packets.
Step 4	**ip authentication key-chain eigrp** *autonomous-system key-chain*	Enables authentication of IP EIGRP packets.

	Command or Action	Purpose
	Example: `Switch(config-if)# ip authentication key-chain eigrp 105 chain1`	
Step 5	**exit** **Example:** `Switch(config-if)# exit`	Returns to global configuration mode.
Step 6	**key chain** *name-of-chain* **Example:** `Switch(config)# key chain chain1`	Identify a key chain and enter key-chain configuration mode. Match the name configured in Step 4.
Step 7	**key** *number* **Example:** `Switch(config-keychain)# key 1`	In key-chain configuration mode, identify the key number.
Step 8	**key-string** *text* **Example:** `Switch(config-keychain-key)# key-string key1`	In key-chain key configuration mode, identify the key string.
Step 9	**accept-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*} **Example:** `Switch(config-keychain-key)# accept-lifetime 13:30:00 Jan 25 2011 duration 7200`	(Optional) Specifies the time period during which the key can be received. The *start-time* and *end-time* syntax can be either *hh:mm:ss Month date year* or *hh:mm:ss date Month year*. The default is forever with the default *start-time* and the earliest acceptable date as January 1, 1993. The default *end-time* and **duration** is **infinite**.
Step 10	**send-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*} **Example:** `Switch(config-keychain-key)# send-lifetime 14:00:00 Jan 25 2011 duration 3600`	(Optional) Specifies the time period during which the key can be sent. The *start-time* and *end-time* syntax can be either *hh:mm:ss Month date year* or *hh:mm:ss date Month year*. The default is forever with the default *start-time* and the earliest acceptable date as January 1, 1993. The default *end-time* and **duration** is **infinite**.
Step 11	**end** **Example:** `Switch(config-keychain-key)# exit`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 12	**show key chain** **Example:** `Switch# show key chain`	Displays authentication key information.
Step 13	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring and Maintaining EIGRP

You can delete neighbors from the neighbor table. You can also display various EIGRP routing statistics. Table 41-8 lists the privileged EXEC commands for deleting neighbors and displaying statistics. For explanations of fields in the resulting display, see the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.4*.

Table 84: IP EIGRP Clear and Show Commands

clear ip eigrp neighbors [*if-address* | *interface*]	Deletes neighbors from the neighbor table.
show ip eigrp interface [*interface*] [*as number*]	Displays information about interfaces configured for EIGRP.
show ip eigrp neighbors [*type-number*]	Displays EIGRP discovered neighbors.
show ip eigrp topology [*autonomous-system-number*] | [[*ip-address*] *mask*]]	Displays the EIGRP topology table for a given process.
show ip eigrp traffic [*autonomous-system-number*]	Displays the number of packets sent and received for all or a specified EIGRP process.

Information About BGP

The Border Gateway Protocol (BGP) is an exterior gateway protocol used to set up an interdomain routing system that guarantees the loop-free exchange of routing information between autonomous systems. Autonomous systems are made up of routers that operate under the same administration and that run Interior Gateway Protocols (IGPs), such as RIP or OSPF, within their boundaries and that interconnect by using an Exterior Gateway Protocol (EGP). BGP Version 4 is the standard EGP for interdomain routing in the Internet. The protocol is defined in RFCs 1163, 1267, and 1771. You can find detailed information about BGP in

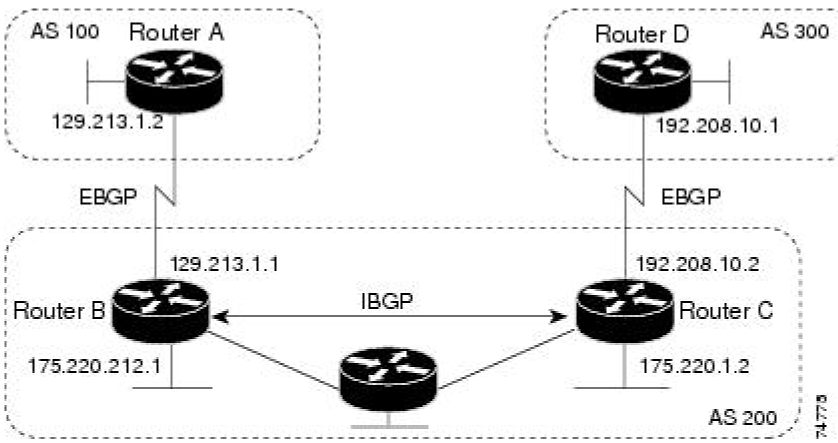
Internet Routing Architectures, published by Cisco Press, and in the "Configuring BGP" chapter in the *Cisco IP and IP Routing Configuration Guide*.

For details about BGP commands and keywords, see the "IP Routing Protocols" part of the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols* .

BGP Network Topology

Routers that belong to the same autonomous system (AS) and that exchange BGP updates run internal BGP (IBGP), and routers that belong to different autonomous systems and that exchange BGP updates run external BGP (EBGP). Most configuration commands are the same for configuring EBGP and IBGP. The difference is that the routing updates are exchanged either between autonomous systems (EBGP) or within an AS (IBGP). Figure 41-5 shows a network that is running both EBGP and IBGP.

Figure 22: EBGP, IBGP, and Multiple Autonomous Systems



Before exchanging information with an external AS, BGP ensures that networks within the AS can be reached by defining internal BGP peering among routers within the AS and by redistributing BGP routing information to IGPs that run within the AS, such as IGRP and OSPF.

Routers that run a BGP routing process are often referred to as BGP speakers. BGP uses the Transmission Control Protocol (TCP) as its transport protocol (specifically port 179). Two BGP speakers that have a TCP connection to each other for exchanging routing information are known as peers or neighbors. In Figure 41-5, Routers A and B are BGP peers, as are Routers B and C and Routers C and D. The routing information is a series of AS numbers that describe the full path to the destination network. BGP uses this information to construct a loop-free map of autonomous systems.

The network has these characteristics:

- Routers A and B are running EBGP, and Routers B and C are running IBGP. Note that the EBGP peers are directly connected and that the IBGP peers are not. As long as there is an IGP running that allows the two neighbors to reach one another, IBGP peers do not have to be directly connected.
- All BGP speakers within an AS must establish a peer relationship with each other. That is, the BGP speakers within an AS must be fully meshed logically. BGP4 provides two techniques that reduce the requirement for a logical full mesh: confederations and route reflectors.
- AS 200 is a transit AS for AS 100 and AS 300—that is, AS 200 is used to transfer packets between AS 100 and AS 300.

BGP peers initially exchange their full BGP routing tables and then send only incremental updates. BGP peers also exchange keepalive messages (to ensure that the connection is up) and notification messages (in response to errors or special conditions).

In BGP, each route consists of a network number, a list of autonomous systems that information has passed through (the autonomous system path), and a list of other path attributes. The primary function of a BGP system is to exchange network reachability information, including information about the list of AS paths, with other BGP systems. This information can be used to determine AS connectivity, to prune routing loops, and to enforce AS-level policy decisions.

A router or switch running Cisco IOS does not select or use an IBGP route unless it has a route available to the next-hop router and it has received synchronization from an IGP (unless IGP synchronization is disabled). When multiple routes are available, BGP bases its path selection on attribute values. See the “Configuring BGP Decision Attributes” section for information about BGP attributes.

BGP Version 4 supports classless interdomain routing (CIDR) so you can reduce the size of your routing tables by creating aggregate routes, resulting in supernets. CIDR eliminates the concept of network classes within BGP and supports the advertising of IP prefixes.

Nonstop Forwarding Awareness

The BGP NSF Awareness feature is supported for IPv4 in the IP services feature set. To enable this feature with BGP routing, you need to enable Graceful Restart. When the neighboring router is NSF-capable, and this feature is enabled, the Layer 3 switch continues to forward packets from the neighboring router during the interval between the primary Route Processor (RP) in a router failing and the backup RP taking over, or while the primary RP is manually reloaded for a nondisruptive software upgrade.

For more information, see the “BGP Nonstop Forwarding (NSF) Awareness” section of the *Cisco IOS IP Routing Protocols Configuration Guide, Release 12.4*.

Information About BGP Routing

To enable BGP routing, you establish a BGP routing process and define the local network. Because BGP must completely recognize the relationships with its neighbors, you must also specify a BGP neighbor.

BGP supports two kinds of neighbors: internal and external. Internal neighbors are in the same AS; external neighbors are in different autonomous systems. External neighbors are usually adjacent to each other and share a subnet, but internal neighbors can be anywhere in the same AS.

The switch supports the use of private AS numbers, usually assigned by service providers and given to systems whose routes are not advertised to external neighbors. The private AS numbers are from 64512 to 65535. You can configure external neighbors to remove private AS numbers from the AS path by using the **neighbor remove-private-as** router configuration command. Then when an update is passed to an external neighbor, if the AS path includes private AS numbers, these numbers are dropped.

If your AS will be passing traffic through it from another AS to a third AS, it is important to be consistent about the routes it advertises. If BGP advertised a route before all routers in the network had learned about the route through the IGP, the AS might receive traffic that some routers could not yet route. To prevent this from happening, BGP must wait until the IGP has propagated information across the AS so that BGP is synchronized with the IGP. Synchronization is enabled by default. If your AS does not pass traffic from one AS to another AS, or if all routers in your autonomous systems are running BGP, you can disable synchronization, which allows your network to carry fewer routes in the IGP and allows BGP to converge more quickly.

Routing Policy Changes

Routing policies for a peer include all the configurations that might affect inbound or outbound routing table updates. When you have defined two routers as BGP neighbors, they form a BGP connection and exchange routing information. If you later change a BGP filter, weight, distance, version, or timer, or make a similar configuration change, you must reset the BGP sessions so that the configuration changes take effect.

There are two types of reset, hard reset and soft reset. Cisco IOS Releases 12.1 and later support a soft reset without any prior configuration. To use a soft reset without preconfiguration, both BGP peers must support the soft route refresh capability, which is advertised in the OPEN message sent when the peers establish a TCP session. A soft reset allows the dynamic exchange of route refresh requests and routing information between BGP routers and the subsequent re-advertisement of the respective outbound routing table.

- When soft reset generates inbound updates from a neighbor, it is called dynamic inbound soft reset.
- When soft reset sends a set of updates to a neighbor, it is called outbound soft reset.

A soft inbound reset causes the new inbound policy to take effect. A soft outbound reset causes the new local outbound policy to take effect without resetting the BGP session. As a new set of updates is sent during outbound policy reset, a new inbound policy can also take effect.

Table 41-10 lists the advantages and disadvantages hard reset and soft reset.

Table 85: Advantages and Disadvantages of Hard and Soft Resets

Type of Reset	Advantages	Disadvantages
Hard reset	No memory overhead	The prefixes in the BGP, IP, and FIB tables provided by the neighbor are lost. Not recommended.
Outbound soft reset	No configuration, no storing of routing table updates	Does not reset inbound routing table updates.
Dynamic inbound soft reset	Does not clear the BGP session and cache Does not require storing of routing table updates and has no memory overhead	Both BGP routers must support the route refresh capability (in Cisco IOS Release 12.1 and later).

BGP Decision Attributes

When a BGP speaker receives updates from multiple autonomous systems that describe different paths to the same destination, it must choose the single best path for reaching that destination. When chosen, the selected path is entered into the BGP routing table and propagated to its neighbors. The decision is based on the value of attributes that the update contains and other BGP-configurable factors.

When a BGP peer learns two EBGP paths for a prefix from a neighboring AS, it chooses the best path and inserts that path in the IP routing table. If BGP multipath support is enabled and the EBGP paths are learned from the same neighboring autonomous systems, instead of a single best path, multiple paths are installed in the IP routing table. Then, during packet switching, per-packet or per-destination load-balancing is performed among the multiple paths. The **maximum-paths** router configuration command controls the number of paths allowed.

These factors summarize the order in which BGP evaluates the attributes for choosing the best path:

If the path specifies a next hop that is inaccessible, drop the update. The BGP next-hop attribute, automatically determined by the software, is the IP address of the next hop that is going to be used to reach a destination. For EBGP, this is usually the IP address of the neighbor specified by the **neighbor remote-as router** configuration command. You can disable next-hop processing by using route maps or the **neighbor next-hop-self** router configuration command.

1. Prefer the path with the largest weight (a Cisco proprietary parameter). The weight attribute is local to the router and not propagated in routing updates. By default, the weight attribute is 32768 for paths that the router originates and zero for other paths. Routes with the largest weight are preferred. You can use access lists, route maps, or the **neighbor weight** router configuration command to set weights.
2. Prefer the route with the highest local preference. Local preference is part of the routing update and exchanged among routers in the same AS. The default value of the local preference attribute is 100. You can set local preference by using the **bgp default local-preference** router configuration command or by using a route map.
3. Prefer the route that was originated by BGP running on the local router.
4. Prefer the route with the shortest AS path.
5. Prefer the route with the lowest origin type. An interior route or IGP is lower than a route learned by EGP, and an EGP-learned route is lower than one of unknown origin or learned in another way.
6. Prefer the route with the lowest multi -exit discriminator (MED) metric attribute if the neighboring AS is the same for all routes considered. You can configure the MED by using route maps or by using the **default-metric** router configuration command. When an update is sent to an IBGP peer, the MED is included.
7. Prefer the external (EBGP) path over the internal (IBGP) path.
8. Prefer the route that can be reached through the closest IGP neighbor (the lowest IGP metric). This means that the router will prefer the shortest internal path within the AS to reach the destination (the shortest path to the BGP next-hop).
9. If the following conditions are all true, insert the route for this path into the IP routing table:

   Both the best route and this route are external.

   Both the best route and this route are from the same neighboring autonomous system.

   maximum-paths is enabled.
10. If multipath is not enabled, prefer the route with the lowest IP address value for the BGP router ID. The router ID is usually the highest IP address on the router or the loopback (virtual) address, but might be implementation-specific.

Route Maps

Within BGP, route maps can be used to control and to modify routing information and to define the conditions by which routes are redistributed between routing domains. See the “Using Route Maps to Redistribute Routing Information” section on page 41-124 for more information about route maps. Each route map has a name that identifies the route map (*map tag*) and an optional sequence number.

BGP Filtering

You can filter BGP advertisements by using AS-path filters, such as the **as-path access-list** global configuration command and the **neighbor filter-list** router configuration command. You can also use access lists with the **neighbor distribute-list** router configuration command. Distribute-list filters are applied to network numbers. See the "Controlling Advertising and Processing in Routing Updates" section on page 41-135 for information about the **distribute-list** command.

You can use route maps on a per-neighbor basis to filter updates and to modify various attributes. A route map can be applied to either inbound or outbound updates. Only the routes that pass the route map are sent or accepted in updates. On both inbound and outbound updates, matching is supported based on AS path, community, and network numbers. Autonomous system path matching requires the **match as-path access-list** route-map command, community based matching requires the **match community-list** route-map command, and network-based matching requires the **ip access-list** global configuration command.

Prefix List for BGP Filtering

You can use prefix lists as an alternative to access lists in many BGP route filtering commands, including the **neighbor distribute-list** router configuration command. The advantages of using prefix lists include performance improvements in loading and lookup of large lists, incremental update support, easier CLI configuration, and greater flexibility.

Filtering by a prefix list involves matching the prefixes of routes with those listed in the prefix list, as when matching access lists. When there is a match, the route is used. Whether a prefix is permitted or denied is based upon these rules:

- An empty prefix list permits all prefixes.
- An implicit deny is assumed if a given prefix does not match any entries in a prefix list.
- When multiple entries of a prefix list match a given prefix, the sequence number of a prefix list entry identifies the entry with the lowest sequence number.

By default, sequence numbers are generated automatically and incremented in units of five. If you disable the automatic generation of sequence numbers, you must specify the sequence number for each entry. You can specify sequence values in any increment. If you specify increments of one, you cannot insert additional entries into the list; if you choose very large increments, you might run out of values.

BGP Community Filtering

One way that BGP controls the distribution of routing information based on the value of the COMMUNITIES attribute. The attribute is a way to groups destinations into communities and to apply routing decisions based on the communities. This method simplifies configuration of a BGP speaker to control distribution of routing information.

A community is a group of destinations that share some common attribute. Each destination can belong to multiple communities. AS administrators can define to which communities a destination belongs. By default, all destinations belong to the general Internet community. The community is identified by the COMMUNITIES attribute, an optional, transitive, global attribute in the numerical range from 1 to 4294967200. These are some predefined, well-known communities:

- **internet—**Advertise this route to the Internet community. All routers belong to it.

- **no-export—**Do not advertise this route to EBGP peers.
- **no-advertise—**Do not advertise this route to any peer (internal or external).
- **local-as—**Do not advertise this route to peers outside the local autonomous system.

Based on the community, you can control which routing information to accept, prefer, or distribute to other neighbors. A BGP speaker can set, append, or modify the community of a route when learning, advertising, or redistributing routes. When routes are aggregated, the resulting aggregate has a COMMUNITIES attribute that contains all communities from all the initial routes.

You can use community lists to create groups of communities to use in a match clause of a route map. As with an access list, a series of community lists can be created. Statements are checked until a match is found. As soon as one statement is satisfied, the test is concluded.

To set the COMMUNITIES attribute and match clauses based on communities, see the **match community-list** and **set community** route-map configuration commands in the "Using Route Maps to Redistribute Routing Information" section.

BGP Neighbors and Peer Groups

Often many BGP neighbors are configured with the same update policies (that is, the same outbound route maps, distribute lists, filter lists, update source, and so on). Neighbors with the same update policies can be grouped into peer groups to simplify configuration and to make updating more efficient. When you have configured many peers, we recommend this approach.

To configure a BGP peer group, you create the peer group, assign options to the peer group, and add neighbors as peer group members. You configure the peer group by using the **neighbor** router configuration commands. By default, peer group members inherit all the configuration options of the peer group, including the remote-as (if configured), version, update-source, out-route-map, out-filter-list, out-dist-list, minimum-advertisement-interval, and next-hop-self. All peer group members also inherit changes made to the peer group. Members can also be configured to override the options that do not affect outbound updates.

Aggregate Routes

Classless interdomain routing (CIDR) enables you to create aggregate routes (or supernets) to minimize the size of routing tables. You can configure aggregate routes in BGP either by redistributing an aggregate route into BGP or by creating an aggregate entry in the BGP routing table. An aggregate address is added to the BGP table when there is at least one more specific entry in the BGP table.

Routing Domain Confederations

One way to reduce the IBGP mesh is to divide an autonomous system into multiple subautonomous systems and to group them into a single confederation that appears as a single autonomous system. Each autonomous system is fully meshed within itself and has a few connections to other autonomous systems in the same confederation. Even though the peers in different autonomous systems have EBGP sessions, they exchange routing information as if they were IBGP peers. Specifically, the next hop, MED, and local preference information is preserved. You can then use a single IGP for all of the autonomous systems.

BGP Route Reflectors

BGP requires that all of the IBGP speakers be fully meshed. When a router receives a route from an external neighbor, it must advertise it to all internal neighbors. To prevent a routing information loop, all IBPG speakers must be connected. The internal neighbors do not send routes learned from internal neighbors to other internal neighbors.

With route reflectors, all IBGP speakers need not be fully meshed because another method is used to pass learned routes to neighbors. When you configure an internal BGP peer to be a route reflector, it is responsible for passing IBGP learned routes to a set of IBGP neighbors. The internal peers of the route reflector are divided into two groups: client peers and nonclient peers (all the other routers in the autonomous system). A route reflector reflects routes between these two groups. The route reflector and its client peers form a cluster. The nonclient peers must be fully meshed with each other, but the client peers need not be fully meshed. The clients in the cluster do not communicate with IBGP speakers outside their cluster.

When the route reflector receives an advertised route, it takes one of these actions, depending on the neighbor:

- A route from an external BGP speaker is advertised to all clients and nonclient peers.
- A route from a nonclient peer is advertised to all clients.
- A route from a client is advertised to all clients and nonclient peers. Hence, the clients need not be fully meshed.

Usually a cluster of clients have a single route reflector, and the cluster is identified by the route reflector router ID. To increase redundancy and to avoid a single point of failure, a cluster might have more than one route reflector. In this case, all route reflectors in the cluster must be configured with the same 4-byte cluster ID so that a route reflector can recognize updates from route reflectors in the same cluster. All the route reflectors serving a cluster should be fully meshed and should have identical sets of client and nonclient peers.

Route Dampening

Route flap dampening is a BGP feature designed to minimize the propagation of flapping routes across an internetwork. A route is considered to be flapping when it is repeatedly available, then unavailable, then available, then unavailable, and so on. When route dampening is enabled, a numeric penalty value is assigned to a route when it flaps. When a route's accumulated penalties reach a configurable limit, BGP suppresses advertisements of the route, even if the route is running. The reuse limit is a configurable value that is compared with the penalty. If the penalty is less than the reuse limit, a suppressed route that is up is advertised again.

Dampening is not applied to routes that are learned by IBGP. This policy prevents the IBGP peers from having a higher penalty for routes external to the AS.

More BGP Information

For detailed descriptions of BGP configuration, see the "Configuring BGP" chapter in the "IP Routing Protocols" part of the *Cisco IOS IP Configuration Guide, Release 12.4*. For details about specific commands, see the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.4*.

How to Configure BGP

Default BGP Configuration

Table 41-9 shows the basic default BGP configuration. For the defaults for all characteristics, see the specific commands in the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.4*.

Table 86: Default BGP Configuration

Feature	**Default Setting**
Aggregate address	Disabled: None defined.
AS path access list	None defined.
Auto summary	Disabled.
Best path	• The router considers *as-path* in choosing a route and does not compare similar routes from external BGP peers. • Compare router ID: Disabled.
BGP community list	• Number: None defined. When you permit a value for the community number, the list defaults to an implicit deny for everything else that has not been permitted. • Format: Cisco default format (32-bit number).
BGP confederation identifier/peers	• Identifier: None configured. • Peers: None identified.
BGP Fast external fallover	Enabled.
BGP local preference	100. The range is 0 to 4294967295 with the higher value preferred.
BGP network	None specified; no backdoor route advertised.
BGP route dampening	Disabled by default. When enabled: • Half-life is 15 minutes. • Re-use is 750 (10-second increments). • Suppress is 2000 (10-second increments). • Max-suppress-time is 4 times half-life; 60 minutes.

Feature	Default Setting
BGP router ID	The IP address of a loopback interface if one is configured or the highest IP address configured for a physical interface on the router.
Default information originate (protocol or network redistribution)	Disabled.
Default metric	Built-in, automatic metric translations.
Distance	• External route administrative distance: 20 (acceptable values are from 1 to 255). • Internal route administrative distance: 200 (acceptable values are from 1 to 255). • Local route administrative distance: 200 (acceptable values are from 1 to 255).
Distribute list	• In (filter networks received in updates): Disabled. • Out (suppress networks from being advertised in updates): Disabled.
Internal route redistribution	Disabled.
IP prefix list	None defined.
Multi exit discriminator (MED)	• Always compare: Disabled. Does not compare MEDs for paths from neighbors in different autonomous systems. • Best path compare: Disabled. • MED missing as worst path: Disabled. • Deterministic MED comparison is disabled.

Feature	Default Setting
Neighbor	• Advertisement interval: 30 seconds for external peers; 5 seconds for internal peers. • Change logging: Enabled. • Conditional advertisement: Disabled. • Default originate: No default route is sent to the neighbor. • Description: None. • Distribute list: None defined. • External BGP multihop: Only directly connected neighbors are allowed. • Filter list: None used. • Maximum number of prefixes received: No limit. • Next hop (router as next hop for BGP neighbor): Disabled. • Password: Disabled. • Peer group: None defined; no members assigned. • Prefix list: None specified. • Remote AS (add entry to neighbor BGP table): No peers defined. • Private AS number removal: Disabled. • Route maps: None applied to a peer. • Send community attributes: None sent to neighbors. • Shutdown or soft reconfiguration: Not enabled. • Timers: keepalive: 60 seconds; holdtime: 180 seconds. • Update source: Best local address. • Version: BGP Version 4. • Weight: Routes learned through BGP peer: 0; routes sourced by the local router: 32768.
NSF[9] Awareness	Disabled[10]. If enabled, allows Layer 3 switches to continue forwarding packets from a neighboring NSF-capable router during hardware or software changes.
Route reflector	None configured.
Synchronization (BGP and IGP)	Disabled.
Table map update	Disabled.

Feature	Default Setting
Timers	Keepalive: 60 seconds; holdtime: 180 seconds.

9 Nonstop Forwarding

10 NSF Awareness can be enabled for IPv4 on switches with the IP services feature set by enabling Graceful Restart.

Enabling BGP Routing

Before You Begin

Note To enable BGP, the switch or stack master must be running the IP services feature set.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip routing** **Example:** `Switch(config)# ip routing`	Enables IP routing.
Step 3	**router bgp** *autonomous-system* **Example:** `Switch(config)# router bgp 45000`	Enables a BGP routing process, assign it an AS number, and enter router configuration mode. The AS number can be from 1 to 65535, with 64512 to 65535 designated as private autonomous numbers.
Step 4	**network** *network-number* [**mask** *network-mask*] [**route-map** *route-map-name*] **Example:** `Switch(config)# network 10.108.0.0`	Configures a network as local to this AS, and enter it in the BGP table.
Step 5	**neighbor** {*ip-address* | *peer-group-name*} **remote-as** *number* **Example:** `Switch(config)# neighbor 10.108.1.2 remote-as 65200`	Adds an entry to the BGP neighbor table specifying that the neighbor identified by the IP address belongs to the specified AS. For EBGP, neighbors are usually directly connected, and the IP address is the address of the interface at the other end of the connection. For IBGP, the IP address can be the address of any of the router interfaces.

	Command or Action	Purpose
Step 6	**neighbor** {*ip-address* | *peer-group-name*} **remove-private-as** **Example:** `Switch(config)# neighbor 172.16.2.33 remove-private-as`	(Optional) Removes private AS numbers from the AS-path in outbound routing updates.
Step 7	**synchronization** **Example:** `Switch(config)# synchronization`	(Optional) Enables synchronization between BGP and an IGP.
Step 8	**auto-summary** **Example:** `Switch(config)# auto-summary`	(Optional) Enables automatic network summarization. When a subnet is redistributed from an IGP into BGP, only the network route is inserted into the BGP table.
Step 9	**bgp graceful-restart** **Example:** `Switch(config)# bgp graceful-start`	(Optional) Enables NSF awareness on switch. By default, NSF awareness is disabled.
Step 10	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 11	**show ip bgp network** *network-number* **Example:** `Switch# show ip bgp network 10.108.0.0`	Verifies the configuration.
Step 12	**show ip bgp neighbor** **Example:** `Switch# show ip bgp neighbor`	Verifies that NSF awareness (Graceful Restart) is enabled on the neighbor. If NSF awareness is enabled on the switch and the neighbor, this message appears: *Graceful Restart Capability: advertised and received* If NSF awareness is enabled on the switch, but not on the neighbor, this message appears: *Graceful Restart Capability: advertised*
Step 13	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Managing Routing Policy Changes

To learn if a BGP peer supports the route refresh capability and to reset the BGP session:

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show ip bgp neighbors** **Example:** `Switch# show ip bgp neighbors`	Displays whether a neighbor supports the route refresh capability. When supported, this message appears for the router: *Received route refresh capability from peer.*
Step 2	**clear ip bgp** {***** | *address* | *peer-group-name*} **Example:** `Switch# clear ip bgp *`	Resets the routing table on the specified connection. • Enter an asterisk (*) to specify that all connections be reset. • Enter an IP address to specify the connection to be reset. • Enter a peer group name to reset the peer group.
Step 3	**clear ip bgp** {***** | *address* | *peer-group-name*} **soft out** **Example:** `Switch# clear ip bgp * soft out`	(Optional) Performs an outbound soft reset to reset the inbound routing table on the specified connection. Use this command if route refresh is supported. • Enter an asterisk (*) to specify that all connections be reset. • Enter an IP address to specify the connection to be reset. • Enter a peer group name to reset the peer group.
Step 4	**show ip bgp** **Example:** `Switch# show ip bgp`	Verifies the reset by checking information about the routing table and about BGP neighbors.
Step 5	**show ip bgp neighbors** **Example:** `Switch# show ip bgp neighbors`	Verifies the reset by checking information about the routing table and about BGP neighbors.

Configuring BGP Decision Attributes

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router bgp** *autonomous-system* **Example:** `Switch(config)# router bgp 4500`	Enables a BGP routing process, assign it an AS number, and enter router configuration mode.
Step 3	**bgp best-path as-path ignore** **Example:** `Switch(config-router)# bgp bestpath as-path ignore`	(Optional) Configures the router to ignore AS path length in selecting a route.
Step 4	**neighbor** {*ip-address* | *peer-group-name*} **next-hop-self** **Example:** `Switch(config-router)# neighbor 10.108.1.1 next-hop-self`	(Optional) Disables next-hop processing on BGP updates to a neighbor by entering a specific IP address to be used instead of the next-hop address.
Step 5	**neighbor** {*ip-address* | *peer-group-name*} **weight** *weight* **Example:** `Switch(config-router)# neighbor 172.16.12.1 weight 50`	(Optional) Assign a weight to a neighbor connection. Acceptable values are from 0 to 65535; the largest weight is the preferred route. Routes learned through another BGP peer have a default weight of 0; routes sourced by the local router have a default weight of 32768.
Step 6	**default-metric** *number* **Example:** `Switch(config-router)# default-metric 300`	(Optional) Sets a MED metric to set preferred paths to external neighbors. All routes without a MED will also be set to this value. The range is 1 to 4294967295. The lowest value is the most desirable.
Step 7	**bgp bestpath med missing-as-worst** **Example:** `Switch(config-router)# bgp bestpath med missing-as-worst`	(Optional) Configures the switch to consider a missing MED as having a value of infinity, making the path without a MED value the least desirable path.

	Command or Action	Purpose
Step 8	**bgp always-compare med** **Example:** `Switch(config-router)# bgp always-compare-med`	(Optional) Configures the switch to compare MEDs for paths from neighbors in different autonomous systems. By default, MED comparison is only done among paths in the same AS.
Step 9	**bgp bestpath med confed** **Example:** `Switch(config-router)# bgp bestpath med confed`	(Optional) Configures the switch to consider the MED in choosing a path from among those advertised by different subautonomous systems within a confederation.
Step 10	**bgp deterministic med** **Example:** `Switch(config-router)# bgp deterministic med`	(Optional) Configures the switch to consider the MED variable when choosing among routes advertised by different peers in the same AS.
Step 11	**bgp default local-preference** *value* **Example:** `Switch(config-router)# bgp default local-preference 200`	(Optional) Change the default local preference value. The range is 0 to 4294967295; the default value is 100. The highest local preference value is preferred.
Step 12	**maximum-paths** *number* **Example:** `Switch(config-router)# maximum-paths 8`	(Optional) Configures the number of paths to be added to the IP routing table. The default is to only enter the best path in the routing table. The range is from 1 to 16. Having multiple paths allows load-balancing among the paths. (Although the switch software allows a maximum of 32 equal-cost routes, the switch hardware will never use more than 16 paths per route.)
Step 13	**end** **Example:** `Switch(config-router)# end`	Returns to privileged EXEC mode.
Step 14	**show ip bgp** **Example:** `Switch# show ip bgp`	Verifies the reset by checking information about the routing table and about BGP neighbors.
Step 15	**show ip bgp neighbors** **Example:** `Switch# show ip bgp neighbors`	Verifies the reset by checking information about the routing table and about BGP neighbors.

	Command or Action	Purpose
Step 16	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring BGP Filtering with Route Maps

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**route-map** *map-tag* [**permit** | **deny**] [*sequence-number*] **Example:** `Switch(config)# route-map set-peer-address permit 10`	Creates a route map, and enter route-map configuration mode.
Step 3	**set ip next-hop** *ip-address* [*...ip-address*] [**peer-address**] **Example:** `Switch(config)# set ip next-hop 10.1.1.3`	(Optional) Sets a route map to disable next-hop processing • In an inbound route map, set the next hop of matching routes to be the neighbor peering address, overriding third-party next hops. • In an outbound route map of a BGP peer, set the next hop to the peering address of the local router, disabling the next-hop calculation.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show route-map** [*map-name*] **Example:** `Switch# show route-map`	Displays all route maps configured or only the one specified to verify configuration.

	Command or Action	Purpose
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring BGP Filtering by Neighbor

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router bgp** *autonomous-system* **Example:** `Switch(config)# router bgp 109`	Enables a BGP routing process, assign it an AS number, and enter router configuration mode.
Step 3	**neighbor** {*ip-address* | *peer-group name*} **distribute-list** {**access-list-number** | *name*} {**in** | **out**} **Example:** `Switch(config-router)# neighbor 172.16.4.1 distribute-list 39 in`	(Optional) Filters BGP routing updates to or from neighbors as specified in an access list. **Note** You can also use the **neighbor prefix-list** router configuration command to filter updates, but you cannot use both commands to configure the same BGP peer.
Step 4	**neighbor** {*ip-address* | *peer-group name*} **route-map** *map-tag* {**in** | **out**} **Example:** `Switch(config-router)# neighbor 172.16.70.24 route-map internal-map in`	(Optional) Applies a route map to filter an incoming or outgoing route.
Step 5	**end** **Example:** `Switch(config-router)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 6	**show ip bgp neighbors** **Example:** `Switch# show ip bgp neighbors`	Verifies the configuration.
Step 7	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring BGP Filtering by Access Lists and Neighbors

Another method of filtering is to specify an access list filter on both incoming and outbound updates, based on the BGP autonomous system paths. Each filter is an access list based on regular expressions. (See the "Regular Expressions" appendix in the *Cisco IOS Dial Technologies Command Reference, Release 12.4* for more information on forming regular expressions.) To use this method, define an autonomous system path access list, and apply it to updates to and from particular neighbors.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip as-path access-list** *access-list-number* {**permit** | **deny**} *as-regular-expressions* **Example:** `Switch(config)# ip as-path access-list 1 deny _65535_`	Defines a BGP-related access list.
Step 3	**router bgp** *autonomous-system* **Example:** `Switch(config)# router bgp 110`	Enters BGP router configuration mode.
Step 4	**neighbor** {*ip-address* | *peer-group name*} **filter-list** {*access-list-number* | *name*} {**in** | **out** | **weight** *weight*} **Example:** `Switch(config-router)# neighbor 172.16.1.1 filter-list 1 out`	Establishes a BGP filter based on an access list.

	Command or Action	Purpose
Step 5	**end** **Example:** `Switch(config-router)# end`	Returns to privileged EXEC mode.
Step 6	**show ip bgp neighbors** [**paths** *regular-expression*] **Example:** `Switch# show ip bgp neighbors`	Verifies the configuration.
Step 7	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Prefix Lists for BGP Filtering

You do not need to specify a sequence number when removing a configuration entry. **Show** commands include the sequence numbers in their output.

Before using a prefix list in a command, you must set up the prefix list.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip prefix-list** *list-name* [**seq** *seq-value*] **deny** | **permit** *network/len* [**ge** *ge-value*] [**le** *le-value*] **Example:** `Switch(config)# ip prefix-list BLUE permit 172.16.1.0/24`	Creates a prefix list with an optional sequence number to **deny** or **permit** access for matching conditions. You must enter at least one **permit** or **deny** clause. • *network/len* is the network number and length (in bits) of the network mask. • (Optional) **ge** and **le** values specify the range of the prefix length to be matched.The specified *ge-value* and *le-value* must satisfy this condition: *len* < *ge-value* < *le-value* < 32

	Command or Action	Purpose
Step 3	**ip prefix-list** *list-name* **seq** *seq-value* **deny** | **permit** *network/len* [**ge** *ge-value*] [**le** *le-value*] **Example:** `Switch(config)# ip prefix-list BLUE seq 10 permit 172.24.1.0/24`	(Optional) Adds an entry to a prefix list, and assign a sequence number to the entry.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show ip prefix list** [**detail** | **summary**] *name* [*network/len*] [**seq** *seq-num*] [**longer**] [**first-match**] **Example:** `Switch# show ip prefix list summary test`	Verifies the configuration by displaying information about a prefix list or prefix list entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring BGP Community Filtering

By default, no COMMUNITIES attribute is sent to a neighbor. You can specify that the COMMUNITIES attribute be sent to the neighbor at an IP address by using the **neighbor send-community** router configuration command.

SUMMARY STEPS

1. **configure terminal**
2. **ip community-list** *community-list-number* {**permit** | **deny**} *community-number*
3. **router bgp** *autonomous-system*
4. **neighbor** {*ip-address* | *peer-group name*} **send-community**
5. **set comm-list** *list-num* **delete**
6. **exit**
7. **ip bgp-community new-format**
8. **end**
9. **show ip bgp community**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip community-list** *community-list-number* {**permit** | **deny**} *community-number* **Example:** `Switch(config)# ip community-list 1 permit 50000:10`	Creates a community list, and assign it a number. • The *community-list-number* is an integer from 1 to 99 that identifies one or more permit or deny groups of communities. • The *community-number* is the number configured by a **set community** route-map configuration command.
Step 3	**router bgp** *autonomous-system* **Example:** `Switch(config)# router bgp 108`	Enters BGP router configuration mode.
Step 4	**neighbor** {*ip-address* | *peer-group name*} **send-community** **Example:** `Switch(config-router)# neighbor 172.16.70.23 send-community`	Specifies that the COMMUNITIES attribute be sent to the neighbor at this IP address.
Step 5	**set comm-list** *list-num* **delete** **Example:** `Switch(config-router)# set comm-list 500 delete`	(Optional) Removes communities from the community attribute of an inbound or outbound update that match a standard or extended community list specified by a route map.
Step 6	**exit** **Example:** `Switch(config-router)# end`	Returns to global configuration mode.
Step 7	**ip bgp-community new-format** **Example:** `Switch(config)# ip bgp-community new format`	(Optional) Displays and parses BGP communities in the format AA:NN. A BGP community is displayed in a two-part format 2 bytes long. The Cisco default community format is in the format NNAA. In the most recent RFC for BGP, a community takes the form AA:NN, where the first part is the AS number and the second part is a 2-byte number.

	Command or Action	Purpose
Step 8	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 9	**show ip bgp community** **Example:** `Switch# show ip bgp community`	Verifies the configuration.
Step 10	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring BGP Neighbors and Peer Groups

To assign configuration options to an individual neighbor, specify any of these router configuration commands by using the neighbor IP address. To assign the options to a peer group, specify any of the commands by using the peer group name. You can disable a BGP peer or peer group without removing all the configuration information by using the **neighbor shutdown** router configuration command.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**router bgp** *autonomous-system*	Enters BGP router configuration mode.
Step 3	**neighbor** *peer-group-name* **peer-group**	Creates a BGP peer group.
Step 4	**neighbor** *ip-address* **peer-group** *peer-group-name*	Makes a BGP neighbor a member of the peer group.
Step 5	**neighbor** {*ip-address* | *peer-group-name*} **remote-as** *number*	Specifies a BGP neighbor. If a peer group is not configured with a **remote-as** *number*, use this command to create peer groups containing EBGP neighbors. The range is 1 to 65535.
Step 6	**neighbor** {*ip-address* | *peer-group-name*} **description** *text*	(Optional) Associates a description with a neighbor.
Step 7	**neighbor** {*ip-address* | *peer-group-name*} **default-originate** [**route-map** *map-name*]	(Optional) Allows a BGP speaker (the local router) to send the default route 0.0.0.0 to a neighbor for use as a default route.

	Command or Action	Purpose
Step 8	**neighbor** {*ip-address* | *peer-group-name*} **send-community**	(Optional) Specifies that the COMMUNITIES attribute be sent to the neighbor at this IP address.
Step 9	**neighbor** {*ip-address* | *peer-group-name*} **update-source** *interface*	(Optional) Allows internal BGP sessions to use any operational interface for TCP connections.
Step 10	**neighbor** {*ip-address* | *peer-group-name*} **ebgp-multihop**	(Optional) Allows BGP sessions, even when the neighbor is not on a directly connected segment. The multihop session is not established if the only route to the multihop peer's address is the default route (0.0.0.0).
Step 11	**neighbor** {*ip-address* | *peer-group-name*} **local-as** *number*	(Optional) Specifies an AS number to use as the local AS. The range is 1 to 65535.
Step 12	**neighbor** {*ip-address* | *peer-group-name*} **advertisement-interval** *seconds*	(Optional) Sets the minimum interval between sending BGP routing updates.
Step 13	**neighbor** {*ip-address* | *peer-group-name*} **maximum-prefix** *maximum* [*threshold*]	(Optional) Controls how many prefixes can be received from a neighbor. The range is 1 to 4294967295. The *threshold* (optional) is the percentage of maximum at which a warning message is generated. The default is 75 percent.
Step 14	**neighbor** {*ip-address* | *peer-group-name*} **next-hop-self**	(Optional) Disables next-hop processing on the BGP updates to a neighbor.
Step 15	neighbor {*ip-address* | *peer-group-name*} password *string*	(Optional) Sets MD5 authentication on a TCP connection to a BGP peer. The same password must be configured on both BGP peers, or the connection between them is not made.
Step 16	**neighbor** {*ip-address* | *peer-group-name*} **route-map** *map-name* {**in** | **out**}	(Optional) Applies a route map to incoming or outgoing routes.
Step 17	**neighbor** {*ip-address* | *peer-group-name*} **send-community**	(Optional) Specifies that the COMMUNITIES attribute be sent to the neighbor at this IP address.
Step 18	**neighbor** {*ip-address* | *peer-group-name*} **timers** *keepalive holdtime*	(Optional) Sets timers for the neighbor or peer group. • The *keepalive* interval is the time within which keepalive messages are sent to peers. The range is 1 to 4294967295 seconds; the default is 60. • The *holdtime* is the interval after which a peer is declared inactive after not receiving a keepalive message from it. The range is 1 to 4294967295 seconds; the default is 180.
Step 19	**neighbor** {*ip-address* | *peer-group-name*} **weight** *weight*	(Optional) Specifies a weight for all routes from a neighbor.
Step 20	**neighbor** {*ip-address* | *peer-group-name*} **distribute-list** {*access-list-number* | *name*} {**in** | **out**}	(Optional) Filter BGP routing updates to or from neighbors, as specified in an access list.

	Command or Action	Purpose
Step 21	**neighbor** {*ip-address* | *peer-group-name*} **filter-list** *access-list-number* {**in** | **out** | **weight** *weight*}	(Optional) Establish a BGP filter.
Step 22	**neighbor** {*ip-address* | *peer-group-name*} **version** *value*	(Optional) Specifies the BGP version to use when communicating with a neighbor.
Step 23	**neighbor** {*ip-address* | *peer-group-name*} **soft-reconfiguration inbound**	(Optional) Configures the software to start storing received updates.
Step 24	**end**	Returns to privileged EXEC mode.
Step 25	**show ip bgp neighbors**	Verifies the configuration.
Step 26	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Aggregate Addresses in a Routing Table

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router bgp** *autonomous-system* **Example:** `Switch(config)# router bgp 106`	Enters BGP router configuration mode.
Step 3	**aggregate-address** *address mask* **Example:** `Switch(config-router)# aggregate-address 10.0.0.0 255.0.0.0`	Creates an aggregate entry in the BGP routing table. The aggregate route is advertised as coming from the AS, and the atomic aggregate attribute is set to indicate that information might be missing.
Step 4	**aggregate-address** *address mask* **as-set** **Example:** `Switch(config-router)# aggregate-address 10.0.0.0 255.0.0.0 as-set`	(Optional) Generates AS set path information. This command creates an aggregate entry following the same rules as the previous command, but the advertised path will be an AS_SET consisting of all elements contained in all paths. Do not use this keyword when aggregating many paths because this route must be continually withdrawn and updated.

	Command or Action	Purpose
Step 5	**aggregate-address** *address-mask* **summary-only** **Example:** `Switch(config-router)# aggregate-address 10.0.0.0 255.0.0.0 summary-only`	(Optional) Advertises summary addresses only.
Step 6	**aggregate-address** *address mask* **suppress-map** *map-name* **Example:** `Switch(config-router)# aggregate-address 10.0.0.0 255.0.0.0 suppress-map map1`	(Optional) Suppresses selected, more specific routes.
Step 7	**aggregate-address** *address mask* **advertise-map** *map-name* **Example:** `Switch(config-router)# aggregate-address 10.0.0.0 255.0.0.0 advertise-map map2`	(Optional) Generates an aggregate based on conditions specified by the route map.
Step 8	**aggregate-address** *address mask* **attribute-map** *map-name* **Example:** `Switch(config-router)# aggregate-address 10.0.0.0 255.0.0.0 attribute-map map3`	(Optional) Generates an aggregate with attributes specified in the route map.
Step 9	**end** **Example:** `Switch(config-router)# end`	Returns to privileged EXEC mode.
Step 10	**show ip bgp neighbors** [**advertised-routes**] **Example:** `Switch# show ip bgp neighbors`	Verifies the configuration.
Step 11	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Routing Domain Confederations

You must specify a confederation identifier that acts as the autonomous system number for the group of autonomous systems.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router bgp** *autonomous-system* **Example:** `Switch(config)# router bgp 100`	Enters BGP router configuration mode.
Step 3	**bgp confederation identifier** *autonomous-system* **Example:** `Switch(config)# bgp confederation identifier 50007`	Configures a BGP confederation identifier.
Step 4	**bgp confederation peers** *autonomous-system* [*autonomous-system* ...] **Example:** `Switch(config)# bgp confederation peers 51000 51001 51002`	Specifies the autonomous systems that belong to the confederation and that will be treated as special EBGP peers.
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 6	**show ip bgp neighbor** **Example:** `Switch# show ip bgp neighbor`	Verifies the configuration.
Step 7	**show ip bgp network** **Example:** `Switch# show ip bgp network`	Verifies the configuration.
Step 8	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring BGP Route Reflectors

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router bgp** *autonomous-system* **Example:** `Switch(config)# router bgp 101`	Enters BGP router configuration mode.
Step 3	**neighbor** {*ip-address* | *peer-group-name*} **route-reflector-client** **Example:** `Switch(config-router)# neighbor 172.16.70.24 route-reflector-client`	Configures the local router as a BGP route reflector and the specified neighbor as a client.
Step 4	**bgp cluster-id** *cluster-id* **Example:** `Switch(config-router)# bgp cluster-id 10.0.1.2`	(Optional) Configures the cluster ID if the cluster has more than one route reflector.
Step 5	**no bgp client-to-client reflection** **Example:** `Switch(config-router)# no bgp client-to-client reflection`	(Optional) Disables client-to-client route reflection. By default, the routes from a route reflector client are reflected to other clients. However, if the clients are fully meshed, the route reflector does not need to reflect routes to clients.
Step 6	**end** **Example:** `Switch(config-router)# end`	Returns to privileged EXEC mode.
Step 7	**show ip bgp** **Example:** `Switch# show ip bgp`	Verifies the configuration. Displays the originator ID and the cluster-list attributes.
Step 8	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Route Dampening

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router bgp** *autonomous-system* **Example:** `Switch(config)# router bgp 100`	Enters BGP router configuration mode.
Step 3	**bgp dampening** **Example:** `Switch(config-router)# bgp dampening`	Enables BGP route dampening.
Step 4	**bgp dampening** *half-life reuse suppress max-suppress* [**route-map** *map*] **Example:** `Switch(config-router)# bgp dampening 30 1500 10000 120`	(Optional) Changes the default values of route dampening factors.
Step 5	**end** **Example:** `Switch(config-router)# end`	Returns to privileged EXEC mode.
Step 6	**show ip bgp flap-statistics** [{**regexp** *regexp*} | {**filter-list** *list*} | {*address mask* [**longer-prefix**]}] **Example:** `Switch# show ip bgp flap-statistics`	(Optional) Monitors the flaps of all paths that are flapping. The statistics are deleted when the route is not suppressed and is stable.
Step 7	**show ip bgp dampened-paths** **Example:** `Switch# show pi bgp dampened-paths`	(Optional) Displays the dampened routes, including the time remaining before they are suppressed.

	Command or Action	Purpose
Step 8	**clear ip bgp flap-statistics** [{**regexp** *regexp*} | {**filter-list** *list*} | {*address mask* [**longer-prefix**]} **Example:** `Switch# clear ip bgp flap-statistics`	(Optional) Clears BGP flap statistics to make it less likely that a route will be dampened.
Step 9	**clear ip bgp dampening** **Example:** `Switch# clear ip bgp dampening`	(Optional) Clears route dampening information, and unsuppress the suppressed routes.
Step 10	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring and Maintaining BGP

You can remove all contents of a particular cache, table, or database. This might be necessary when the contents of the particular structure have become or are suspected to be invalid.

You can display specific statistics, such as the contents of BGP routing tables, caches, and databases. You can use the information to get resource utilization and solve network problems. You can also display information about node reachability and discover the routing path your device's packets are taking through the network.

Table 41-8 lists the privileged EXEC commands for clearing and displaying BGP. For explanations of the display fields, see the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.4*.

Table 87: IP BGP Clear and Show Commands

clear ip bgp *address*	Resets a particular BGP connection.
clear ip bgp *	Resets all BGP connections.
clear ip bgp peer-group *tag*	Removes all members of a BGP peer group.
show ip bgp *prefix*	Displays peer groups and peers not in peer groups to which the prefix has been advertised. Also display prefix attributes such as the next hop and the local prefix.
show ip bgp cidr-only	Displays all BGP routes that contain subnet and supernet network masks.

show ip bgp community [*community-number*] [**exact**]	Displays routes that belong to the specified communities.
show ip bgp community-list *community-list-number* [**exact-match**]	Displays routes that are permitted by the community list.
show ip bgp filter-list *access-list-number*	Displays routes that are matched by the specified AS path access list.
show ip bgp inconsistent-as	Displays the routes with inconsistent originating autonomous systems.
show ip bgp regexp *regular-expression*	Displays the routes that have an AS path that matches the specified regular expression entered on the command line.
show ip bgp	Displays the contents of the BGP routing table.
show ip bgp neighbors [*address*]	Displays detailed information on the BGP and TCP connections to individual neighbors.
show ip bgp neighbors [*address*] [**advertised-routes** | **dampened-routes** | **flap-statistics** | **paths** *regular-expression* | **received-routes** | **routes**]	Displays routes learned from a particular BGP neighbor.
show ip bgp paths	Displays all BGP paths in the database.
show ip bgp peer-group [*tag*] [**summary**]	Displays information about BGP peer groups.
show ip bgp summary	Displays the status of all BGP connections.

The **bgp log-neighbor changes** command is enabled by default. It allows to log messages that are generated when a BGP neighbor resets, comes up, or goes down.

Configuration Examples for BGP

Example: Configuring BGP on Routers

In Figure 41-5

Router A:

```
Switch(config)# router bgp 100
Switch(config-router)# neighbor 129.213.1.1 remote-as 200
```

Router B:

```
Switch(config)# router bgp 200
```

```
Switch(config-router)# neighbor 129.213.1.2 remote-as 100
Switch(config-router)# neighbor 175.220.1.2 remote-as 200
```

Router C:

```
Switch(config)# router bgp 200
Switch(config-router)# neighbor 175.220.212.1 remote-as 200
Switch(config-router)# neighbor 192.208.10.1 remote-as 300
```

Router D:

```
Switch(config)# router bgp 300
Switch(config-router)# neighbor 192.208.10.2 remote-as 200
```

To verify that BGP peers are running, use the show ip bgp neighbors privileged EXEC command. This is the output of this command on Router A:

```
Switch# show ip bgp neighbors

BGP neighbor is 129.213.1.1, remote AS 200, external link
 BGP version 4, remote router ID 175.220.212.1
 BGP state = established, table version = 3, up for 0:10:59
 Last read 0:00:29, hold time is 180, keepalive interval is 60 seconds
 Minimum time between advertisement runs is 30 seconds
 Received 2828 messages, 0 notifications, 0 in queue
 Sent 2826 messages, 0 notifications, 0 in queue
 Connections established 11; dropped 10
```

Anything other than *state = established* means that the peers are not running. The remote router ID is the highest IP address on that router (or the highest loopback interface). Each time the table is updated with new information, the table version number increments. A table version number that continually increments means that a route is flapping, causing continual routing updates.

For exterior protocols, a reference to an IP network from the **network** router configuration command controls only which networks are advertised. This is in contrast to Interior Gateway Protocols (IGPs), such as EIGRP, which also use the **network** command to specify where to send updates.

For detailed descriptions of BGP configuration, see the "IP Routing Protocols" part of the *Cisco IOS IP Configuration Guide, Release 12.4*. For details about specific commands, see the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.4*.

Information About ISO CLNS Routing

Connectionless Routing

The International Organization for Standardization (ISO) Connectionless Network Service (CLNS) protocol is a standard for the network layer of the Open System Interconnection (OSI) model. Addresses in the ISO network architecture are referred to as network service access point (NSAP) addresses and network entity titles (NETs). Each node in an OSI network has one or more NETs. In addition, each node has many NSAP addresses.

When you enable connectionless routing on the switch by using the **clns routing** global configuration command, the switch makes only forwarding decisions, with no routing-related functionality. For dynamic routing, you must also enable a routing protocol. The switch supports the Intermediate System-to-Intermediate System (IS-IS) dynamic routing protocol that is based on the OSI routing protocol for ISO CLNS networks.

When dynamically routing, you use IS-IS. This routing protocol supports the concept of areas. Within an area, all routers know how to reach all the system IDs. Between areas, routers know how to reach the proper area. IS-IS supports two levels of routing: station routing (within an area) and area routing (between areas).

The key difference between the ISO IGRP and IS-IS NSAP addressing schemes is in the definition of area addresses. Both use the system ID for Level 1 routing (routing within an area). However, they differ in the way addresses are specified for area routing. An ISO IGRP NSAP address includes three separate fields for routing: the domain, area, and system ID. An IS-IS address includes two fields: a single continuous area field (comprising the domain and area fields) and the system ID.

Note For more detailed information about ISO CLNS, see the *Cisco IOS Apollo Domain, Banyan VINES, DECnet, ISO CLNS and XNS Configuration Guide, Release 12.4*. For complete syntax and usage information for the commands used in this chapter, see the *Cisco IOS Apollo Domain, Banyan VINES, DECnet, ISO CLNS and XNS Command Reference, Release 12.4,* use the IOS command reference master index, or search online.

IS-IS Dynamic Routing

IS-IS is an ISO dynamic routing protocol (described in ISO 105890). Unlike other routing protocols, enabling IS-IS requires that you create an IS-IS routing process and assign it to a specific interface, rather than to a network. You can specify more than one IS-IS routing process per Layer 3 switch or router by using the multiarea IS-IS configuration syntax. You then configure the parameters for each instance of the IS-IS routing process.

Small IS-IS networks are built as a single area that includes all the routers in the network. As the network grows larger, it is usually reorganized into a backbone area made up of the connected set of all Level 2 routers from all areas, which is in turn connected to local areas. Within a local area, routers know how to reach all system IDs. Between areas, routers know how to reach the backbone, and the backbone routers know how to reach other areas.

Routers establish Level 1 adjacencies to perform routing within a local area (station routing). Routers establish Level 2 adjacencies to perform routing between Level 1 areas (area routing).

A single Cisco router can participate in routing in up to 29 areas and can perform Level 2 routing in the backbone. In general, each routing process corresponds to an area. By default, the first instance of the routing process configured performs both Level 1and Level 2 routing. You can configure additional router instances, which are automatically treated as Level 1 areas. You must configure the parameters for each instance of the IS-IS routing process individually.

For IS-IS multiarea routing, you can configure only one process to perform Level 2 routing, although you can define up to 29 Level 1 areas for each Cisco unit. If Level 2 routing is configured on any process, all additional processes are automatically configured as Level 1. You can configure this process to perform Level 1 routing at the same time. If Level 2 routing is not desired for a router instance, remove the Level 2 capability using the **is-type** global configuration command. Use the **is-type** command also to configure a different router instance as a Level 2 router.

Note For more detailed information about IS-IS, see the “IP Routing Protocols” chapter of the *Cisco IOS IP Configuration Guide, Release 12.4*. For complete syntax and usage information for the commands used in this section, see the *Cisco IOS IP Command Reference, Release 12.4*.

Nonstop Forwarding Awareness

The integrated IS-IS NSF Awareness feature is supported for IPv4G. The feature allows customer premises equipment (CPE) routers that are NSF-aware to help NSF-capable routers perform nonstop forwarding of packets. The local router is not necessarily performing NSF, but its awareness of NSF allows the integrity and accuracy of the routing database and link-state database on the neighboring NSF-capable router to be maintained during the switchover process.

This feature is automatically enabled and requires no configuration. For more information on this feature, see the *Integrated IS-IS Nonstop Forwarding (NSF) Awareness Feature Guide*.

IS-IS Global Parameters

These are some optional IS-IS global parameters that you can configure:

- You can force a default route into an IS-IS routing domain by configuring a default route controlled by a route map. You can also specify other filtering options configurable under a route map.
- You can configure the router to ignore IS-IS LSPs that are received with internal checksum errors or to purge corrupted LSPs, which causes the initiator of the LSP to regenerate it.
- You can assign passwords to areas and domains.
- You can create aggregate addresses that are represented in the routing table by a summary address (route-summarization). Routes learned from other routing protocols can also be summarized. The metric used to advertise the summary is the smallest metric of all the specific routes.
- You can set an overload bit.
- You can configure the LSP refresh interval and the maximum time that an LSP can remain in the router database without a refresh
- You can set the throttling timers for LSP generation, shortest path first computation, and partial route computation.
- You can configure the switch to generate a log message when an IS-IS adjacency changes state (up or down).
- If a link in the network has a maximum transmission unit (MTU) size of less than 1500 bytes, you can lower the LSP MTU so that routing will still occur.
- The partition avoidance router configuration command prevents an area from becoming partitioned when full connectivity is lost among a Level1-2 border router, adjacent Level 1 routers, and end hosts.

IS-IS Interface Parameters

You can optionally configure certain interface-specific IS-IS parameters, independently from other attached routers. However, if you change some values from the defaults, such as multipliers and time intervals, it makes sense to also change them on multiple routers and interfaces. Most of the interface parameters can be configured for level 1, level 2, or both.

These are some interface level parameters you can configure:

- The default metric on the interface, which is used as a value for the IS-IS metric and assigned when there is no quality of service (QoS) routing performed.

- The hello interval (length of time between hello packets sent on the interface) or the default hello packet multiplier used on the interface to determine the hold time sent in IS-IS hello packets. The hold time determines how long a neighbor waits for another hello packet before declaring the neighbor down. This determines how quickly a failed link or neighbor is detected so that routes can be recalculated. Change the hello-multiplier in circumstances where hello packets are lost frequently and IS-IS adjacencies are failing unnecessarily. You can raise the hello multiplier and lower the hello interval correspondingly to make the hello protocol more reliable without increasing the time required to detect a link failure.
- Other time intervals:
    - Complete sequence number PDU (CSNP) interval. CSNPs are sent by the designated router to maintain database synchronization
    - Retransmission interval. This is the time between retransmission of IS-IS LSPs for point-to-point links.
    - IS-IS LSP retransmission throttle interval. This is the maximum rate (number of milliseconds between packets) at which IS-IS LSPs are re-sent on point-to-point links This interval is different from the retransmission interval, which is the time between successive retransmissions of the same LSP
- Designated router election priority, which allows you to reduce the number of adjacencies required on a multiaccess network, which in turn reduces the amount of routing protocol traffic and the size of the topology database.
- The interface circuit type, which is the type of adjacency desired for neighbors on the specified interface
- Password authentication for the interface

How to Configure ISO CLNS Routing

Default IS-IS Configuration

Table 88: Default IS-IS Configuration

Feature	Default Setting
Ignore link-state PDU (LSP) errors	Enabled.
IS-IS type	Conventional IS-IS: the router acts as both a Level 1 (station) and a Level 2 (area) router. Multiarea IS-IS: the first instance of the IS-IS routing process is a Level 1-2 router. Remaining instances are Level 1 routers.
Default-information originate	Disabled.
Log IS-IS adjacency state changes.	Disabled.

Feature	Default Setting
LSP generation throttling timers	Maximum interval between two consecutive occurrences: 5 seconds. Initial LSP generation delay: 50 ms. Hold time between the first and second LSP generation: 5000 ms.
LSP maximum lifetime (without a refresh)	1200 seconds (20 minutes) before t.he LSP packet is deleted.
LSP refresh interval	Send LSP refreshes every 900 seconds (15 minutes).
Maximum LSP packet size	1497 bytes.
NSF Awareness	Enabled. Allows Layer 3 switches to continue forwarding packets from a neighboring NSF-capable router during hardware or software changes.
Partial route computation (PRC) throttling timers	Maximum PRC wait interval: 5 seconds. Initial PRC calculation delay after a topology change: 2000 ms. Hold time between the first and second PRC calculation: 5000 ms.
Partition avoidance	Disabled.
Password	No area or domain password is defined, and authentication is disabled.
Set-overload-bit	Disabled. When enabled, if no arguments are entered, the overload bit is set immediately and remains set until you enter the **no set-overload-bit** command.
Shortest path first (SPF) throttling timers	Maximum interval between consecutive SFPs: 10 seconds. Initial SFP calculation after a topology change: 5500 ms. Holdtime between the first and second SFP calculation: 5500 ms.
Summary-address	Disabled.

Enabling IS-IS Routing

To enable IS-IS, you specify a name and NET for each routing process. You then enable IS-IS routing on the interface and specify the area for each instance of the routing process.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**clns routing** **Example:** `Switch(config)# clns routing`	Enables ISO connectionless routing on the switch.
Step 3	**router isis** [*area tag*] **Example:** `Switch(config)# router isis tag1`	Enables the IS-IS routing for the specified routing process and enter IS-IS routing configuration mode. (Optional) Use the *area tag* argument to identify the area to which the IS-IS router is assigned. You must enter a value if you are configuring multiple IS-IS areas. The first IS-IS instance configured is Level 1-2 by default. Later instances are automatically Level 1. You can change the level of routing by using the **is-type** global configuration command.
Step 4	**net** network-entity-title **Example:** `Switch(config-router)# net 47.0004.004d.0001.0001.0c11.1111.00`	Configures the NETs for the routing process. If you are configuring multiarea IS-IS, specify a NET for each routing process. You can specify a name for a NET and for an address.
Step 5	**is-type** {**level-1** | **level-1-2** | **level-2-only**} **Example:** `Switch(config-router)# is-type level-2-only`	(Optional) Configures the router to act as a Level 1 (station) router, a Level 2 (area) router for multi-area routing, or both (the default): • **level-1—**act as a station router only • **level-1-2—**act as both a station router and an area router • **level 2—**act as an area router only
Step 6	**exit** **Example:** `Switch(config-router)# end`	Returns to global configuration mode.
Step 7	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Specifies an interface to route IS-IS, and enter interface configuration mode. If the interface is not already configured as a Layer 3 interface, enter the **no switchport** command to put it into Layer 3 mode.

	Command or Action	Purpose
Step 8	**ip router isis** [*area tag*] **Example:** `Switch(config-if)# ip router isis tag1`	Configures an IS-IS routing process for ISO CLNS on the interface and attach an area designator to the routing process.
Step 9	**clns router isis** [*area tag*] **Example:** `Switch(config-if)# clns router isis tag1`	Enables ISO CLNS on the interface.
Step 10	**ip address** *ip-address-mask* **Example:** `Switch(config-if)# ip address 10.0.0.5 255.255.255.0`	Define the IP address for the interface. An IP address is required on all interfaces in an area enabled for IS-IS if any one interface is configured for IS-IS routing.
Step 11	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 12	**show isis** [*area tag*] **database detail** **Example:** `Switch# show isis database detail`	Verifies your entries.
Step 13	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring IS-IS Global Parameters

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**clns routing** **Example:** `Switch(config)# clns routing`	Enables ISO connectionless routing on the switch.
Step 3	**router isis** **Example:** `Switch(config)# router isis`	Specifies the IS-IS routing protocol and enter router configuration mode.
Step 4	**default-information originate** [**route-map** *map-name*] **Example:** `Switch(config-router)# default-information originate route-map map1`	(Optional) Forces a default route into the IS-IS routing domain.If you enter **route-map** *map-name*, the routing process generates the default route if the route map is satisfied.
Step 5	**ignore-lsp-errors** **Example:** `Switch(config-router)# ignore-lsp-errors`	(Optional) Configures the router to ignore LSPs with internal checksum errors, instead of purging the LSPs. This command is enabled by default (corrupted LSPs are dropped). To purge the corrupted LSPs, enter the **no ignore-lsp-errors** router configuration command.
Step 6	**area-password** *password* **Example:** `Switch(config-router)# area-password 1password`	(Optional Configures the area authentication password, which is inserted in Level 1 (station router level) LSPs.
Step 7	**domain-password** *password* **Example:** `Switch(config-router)# domain-password 2password`	(Optional) Configures the routing domain authentication password, which is inserted in Level 2 (area router level) LSPs.
Step 8	**summary-address** *address mask* [**level-1** | **level-1-2** | **level-2**] **Example:** `Switch(config-router)# summary-address 10.1.0.0 255.255.0.0 level-2`	(Optional) Creates a summary of addresses for a given level.
Step 9	**set-overload-bit** [**on-startup** {*seconds* | **wait-for-bgp**}] **Example:** `Switch(config-router)#`	(Optional) Sets an overload bit (a hippity bit) to allow other routers to ignore the router in their shortest path first (SPF) calculations if the router is having problems. • (Optional) **on-startup—**sets the overload bit only on startup. If **on-startup** is not specified, the overload bit is set immediately and

	Command or Action	Purpose
	`set-overload-bit on-startup wait-for-bgp`	remains set until you enter the **no set-overload-bit** command. If **on-startup** is specified, you must enter a number of seconds or **wait-for-bgp**. • *seconds*—When the **on-startup** keyword is configured, causes the overload bit to be set upon system startup and remain set for this number of seconds. The range is from 5 to 86400 seconds. • **wait-for-bgp**—When the **on-startup** keyword is configured, causes the overload bit to be set upon system startup and remain set until BGP has converged. If BGP does not signal IS-IS that it is converged, IS-IS will turn off the overload bit after 10 minutes.
Step 10	**lsp-refresh-interval** *seconds* **Example:** `Switch(config-router)# lsp-refresh-interval 1080`	(Optional) Sets an LSP refresh interval in seconds. The range is from 1 to 65535 seconds. The default is to send LSP refreshes every 900 seconds (15 minutes).
Step 11	**max-lsp-lifetime** *seconds* **Example:** `Switch(config-router)# max-lsp-lifetime 1000`	(Optional) Sets the maximum time that LSP packets remain in the router database without being refreshed. The range is from 1 to 65535 seconds. The default is 1200 seconds (20 minutes). After the specified time interval, the LSP packet is deleted.
Step 12	**lsp-gen-interval** [**level-1** | **level-2**] *lsp-max-wait* [*lsp-initial-wait lsp-second-wait*] **Example:** `Switch(config-router)# lsp-gen-interval level-2 2 50 100`	(Optional) Sets the IS-IS LSP generation throttling timers: • *lsp-max-wait*—the maximum interval (in seconds) between two consecutive occurrences of an LSP being generated. The range is 1 to 120, the default is 5. • *lsp-initial-wait*—the initial LSP generation delay (in milliseconds). The range is 1 to 10000; the default is 50. • *lsp-second-wait*—the hold time between the first and second LSP generation (in milliseconds). The range is 1 to 10000; the default is 5000.
Step 13	**spf-interval** [**level-1** | **level-2**] *spf-max-wait* [*spf-initial-wait spf-second-wait*] **Example:** `Switch(config-router)# spf-interval level-2 5 10 20`	(Optional) Sets IS-IS shortest path first (SPF) throttling timers. • *spf-max-wait*—the maximum interval between consecutive SFPs (in seconds). The range is 1 to 120, the default is 10. • *spf-initial-wait*—the initial SFP calculation after a topology change (in milliseconds). The range is 1 to 10000; the default is 5500. • *spf-second-wait*—the holdtime between the first and second SFP calculation (in milliseconds). The range is 1 to 10000; the default is 5500.

	Command or Action	Purpose
Step 14	**prc-interval** *prc-max-wait* [*prc-initial-wait prc-second-wait*] **Example:** `Switch(config-router)# prc-interval 5 10 20`	(Optional) Sets IS-IS partial route computation (PRC) throttling timers. • *prc-max-wait*—the maximum interval (in seconds) between two consecutive PRC calculations. The range is 1 to 120; the default is 5. • *prc-initial-wait*—the initial PRC calculation delay (in milliseconds) after a topology change. The range is 1 to 10,000; the default is 2000. • *prc-second-wait*—the hold time between the first and second PRC calculation (in milliseconds). The range is 1 to 10,000; the default is 5000.
Step 15	**log-adjacency-changes** [**all**] **Example:** `Switch(config-router)# log-adjacency-changes all`	(Optional) Sets the router to log IS-IS adjacency state changes. Enter **all** to include all changes generated by events that are not related to the Intermediate System-to-Intermediate System Hellos, including End System-to-Intermediate System PDUs and link state packets (LSPs).
Step 16	**lsp-mtu** *size* **Example:** `Switch(config-router)# lsp mtu 1560`	(Optional) Specifies the maximum LSP packet size in bytes. The range is 128 to 4352; the default is 1497 bytes. **Note** If any link in the network has a reduced MTU size, you must change the LSP MTU size on all routers in the network.
Step 17	**partition avoidance** **Example:** `Switch(config-router)# partition avoidance`	(Optional) Causes an IS-IS Level 1-2 border router to stop advertising the Level 1 area prefix into the Level 2 backbone when full connectivity is lost among the border router, all adjacent level 1 routers, and end hosts.
Step 18	**end** **Example:** `Switch(config-router)# end`	Returns to privileged EXEC mode.
Step 19	**show clns** **Example:** `Switch# show clns`	Verifies your entries.
Step 20	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring IS-IS Interface Parameters

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Specifies the interface to be configured and enter interface configuration mode. If the interface is not already configured as a Layer 3 interface, enter the **no switchport** command to put it into Layer 3 mode.
Step 3	**isis metric** *default-metric* [**level-1** | **level-2**] **Example:** `Switch(config-if)# isis metric 15`	(Optional) Configures the metric (or cost) for the specified interface. The range is from 0 to 63. The default is 10. If no level is entered, the default is to apply to both Level 1 and Level 2 routers.
Step 4	**isis hello-interval** {*seconds* | **minimal**} [**level-1** | **level-2**] **Example:** `Switch(config-if)# isis hello-interval minimal`	(Optional) Specifies the length of time between hello packets sent by the switch. By default, a value three times the hello interval *seconds* is advertised as the *holdtime* in the hello packets sent. With smaller hello intervals, topological changes are detected faster, but there is more routing traffic. • **minimal—**causes the system to compute the hello interval based on the hello multiplier so that the resulting hold time is 1 second. • *seconds***—**the range is from 1 to 65535. The default is 10 seconds.
Step 5	**isis hello-multiplier** *multiplier* [**level-1** | **level-2**] **Example:** `Switch(config-if)# isis hello-multiplier 5`	(Optional) Specifies the number of IS-IS hello packets a neighbor must miss before the router should declare the adjacency as down. The range is from 3 to 1000. The default is 3. Using a smaller hello-multiplier causes fast convergence, but can result in more routing instability.
Step 6	**isis csnp-interval** *seconds* [**level-1** | **level-2**] **Example:** `Switch(config-if)# isis csnp-interval 15`	(Optional) Configures the IS-IS complete sequence number PDU (CSNP) interval for the interface. The range is from 0 to 65535. The default is 10 seconds.

	Command or Action	Purpose
Step 7	**isis retransmit-interval** *seconds* **Example:** `Switch(config-if)# isis retransmit-interval 7`	(Optional) Configures the number of seconds between retransmission of IS-IS LSPs for point-to-point links. The value you specify should be an integer greater than the expected round-trip delay between any two routers on the network. The range is from 0 to 65535. The default is 5 seconds.
Step 8	**isis retransmit-throttle-interval** *milliseconds* **Example:** `Switch(config-if)# isis retransmit-throttle-interval 4000`	(Optional) Configures the IS-IS LSP retransmission throttle interval, which is the maximum rate (number of milliseconds between packets) at which IS-IS LSPs will be re-sent on point-to-point links. The range is from 0 to 65535. The default is determined by the **isis lsp-interval** command.
Step 9	**isis priority** *value* [**level-1** | **level-2**] **Example:** `Switch(config-if)# isis priority 50`	(Optional) Configures the priority to use for designated router election. The range is from 0 to 127. The default is 64.
Step 10	**isis circuit-type** {**level-1** | **level-1-2** | **level-2-only**} **Example:** `Switch(config-if)# isis circuit-type level-1-2`	(Optional) Configures the type of adjacency desired for neighbors on the specified interface (specify the interface circuit type). • **level-1—**a Level 1 adjacency is established if there is at least one area address common to both this node and its neighbors. • **level-1-2—**a Level 1 and 2 adjacency is established if the neighbor is also configured as both Level 1 and Level 2 and there is at least one area in common. If there is no area in common, a Level 2 adjacency is established. This is the default. • **level 2—**a Level 2 adjacency is established. If the neighbor router is a Level 1 router, no adjacency is established.
Step 11	**isis password** *password* [**level-1** | **level-2**] **Example:** `Switch(config-if)# isis password secret`	(Optional) Configures the authentication password for an interface. By default, authentication is disabled. Specifying Level 1 or Level 2 enables the password only for Level 1 or Level 2 routing, respectively. If you do not specify a level, the default is Level 1 and Level 2.
Step 12	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 13	**show clns interface** *interface-id* **Example:** `Switch# show clns interface gigabitethernet 1/0/1`	Verifies your entries.

	Command or Action	Purpose
Step 14	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring and Maintaining ISO IGRP and IS-IS

You can remove all contents of a CLNS cache or remove information for a particular neighbor or route. You can display specific CLNS or IS-IS statistics, such as the contents of routing tables, caches, and databases. You can also display information about specific interfaces, filters, or neighbors.

Table 41-13 lists the privileged EXEC commands for clearing and displaying ISO CLNS and IS-IS routing. For explanations of the display fields, see the *Cisco IOS Apollo Domain, Banyan VINES, DECnet, ISO CLNS and XNS Command Reference, Release 12.4,* use the Cisco IOS command reference master index, or search online.

Table 89: ISO CLNS and IS-IS Clear and Show Commands

clear clns cache	Clears and reinitializes the CLNS routing cache.
clear clns es-neighbors	Removes end system (ES) neighbor information from the adjacency database.
clear clns is-neighbors	Removes intermediate system (IS) neighbor information from the adjacency database.
clear clns neighbors	Removes CLNS neighbor information from the adjacency database.
clear clns route	Removes dynamically derived CLNS routing information.
show clns	Displays information about the CLNS network.
show clns cache	Displays the entries in the CLNS routing cache.
show clns es-neighbors	Displays ES neighbor entries, including the associated areas.
show clns filter-expr	Displays filter expressions.
show clns filter-set	Displays filter sets.
show clns interface [*interface-id*]	Displays the CLNS-specific or ES-IS information about each interface.

show clns neighbor	Displays information about IS-IS neighbors.
show clns protocol	List the protocol-specific information for each IS-IS or ISO IGRP routing process in this router.
show clns route	Displays all the destinations to which this router knows how to route CLNS packets.
show clns traffic	Displays information about the CLNS packets this router has seen.
show ip route isis	Displays the current state of the ISIS IP routing table.
show isis database	Displays the IS-IS link-state database.
show isis routes	Displays the IS-IS Level 1 routing table.
show isis spf-log	Displays a history of the shortest path first (SPF) calculations for IS-IS.
show isis topology	Displays a list of all connected routers in all areas.
show route-map	Displays all route maps configured or only the one specified.
trace clns *destination*	Discover the paths taken to a specified destination by packets in the network.
which-route {*nsap-address* | *clns-name*}	Displays the routing table in which the specified CLNS destination is found.

Configuration Examples for ISO CLNS Routing

Example: Configuring IS-IS Routing

This example shows how to configure three routers to run conventional IS-IS as an IP routing protocol. In conventional IS-IS, all routers act as Level 1 and Level 2 routers (by default).

Router A:

```
Switch(config)# clns routing
Switch(config)# router isis
Switch(config-router)# net 49.0001.0000.0000.000a.00
Switch(config-router)# exit
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip router isis
Switch(config-if)# clns router isis
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# ip router isis
Switch(config-if)# clns router isis
Switch(config-router)# exit
```

Router B:

```
Switch(config)# clns routing
Switch(config)# router isis
Switch(config-router)# net 49.0001.0000.0000.000b.00
Switch(config-router)# exit
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip router isis
Switch(config-if)# clns router isis
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# ip router isis
Switch(config-if)# clns router isis
Switch(config-router)# exit
```

Router C:

```
Switch(config)# clns routing
Switch(config)# router isis
Switch(config-router)# net 49.0001.0000.0000.000c.00
Switch(config-router)# exit
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip router isis
Switch(config-if)# clns router isis
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# ip router isis
Switch(config-if)# clns router isis
Switch(config-router)# exit
```

Information About Multi-VRF CE

Virtual Private Networks (VPNs) provide a secure way for customers to share bandwidth over an ISP backbone network. A VPN is a collection of sites sharing a common routing table. A customer site is connected to the service-provider network by one or more interfaces, and the service provider associates each interface with a VPN routing table, called a VPN routing/forwarding (VRF) table.

The switch supports multiple VPN routing/forwarding (multi-VRF) instances in customer edge (CE) devices (multi-VRF CE) when the it is running the IP services or advanced IP services feature set. Multi-VRF CE allows a service provider to support two or more VPNs with overlapping IP addresses.

Note The switch does not use Multiprotocol Label Switching (MPLS) to support VPNs. For information about MPLS VRF, see the *Cisco IOS Switching Services Configuration Guide, Release 12.4*.

Understanding Multi-VRF CE

Multi-VRF CE is a feature that allows a service provider to support two or more VPNs, where IP addresses can be overlapped among the VPNs. Multi-VRF CE uses input interfaces to distinguish routes for different VPNs and forms virtual packet-forwarding tables by associating one or more Layer 3 interfaces with each VRF. Interfaces in a VRF can be either physical, such as Ethernet ports, or logical, such as VLAN SVIs, but an interface cannot belong to more than one VRF at any time.

Note Multi-VRF CE interfaces must be Layer 3 interfaces.

Multi-VRF CE includes these devices:

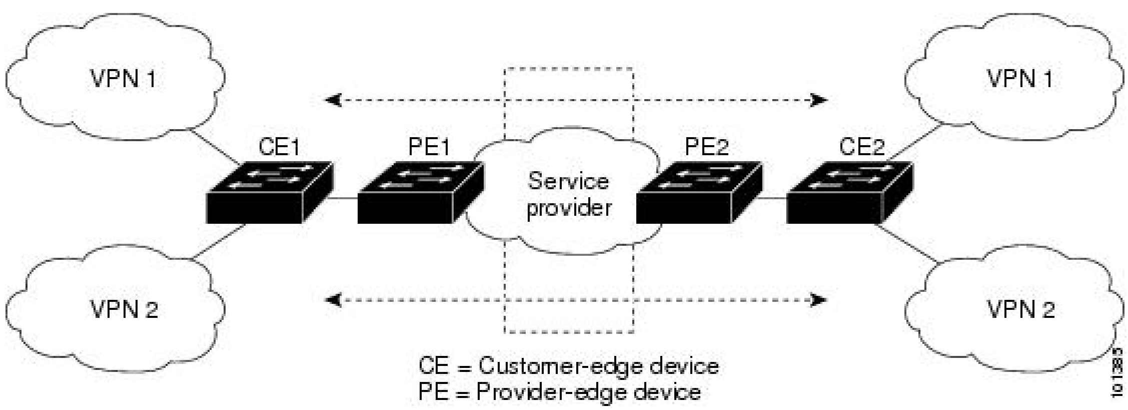
- Customer edge (CE) devices provide customers access to the service-provider network over a data link to one or more provider edge routers. The CE device advertises the site's local routes to the router and learns the remote VPN routes from it. A switch can be a CE.
- Provider edge (PE) routers exchange routing information with CE devices by using static routing or a routing protocol such as BGP, RIPv2, OSPF, or EIGRP. The PE is only required to maintain VPN routes for those VPNs to which it is directly attached, eliminating the need for the PE to maintain all of the service-provider VPN routes. Each PE router maintains a VRF for each of its directly connected sites. Multiple interfaces on a PE router can be associated with a single VRF if all of these sites participate in the same VPN. Each VPN is mapped to a specified VRF. After learning local VPN routes from CEs, a PE router exchanges VPN routing information with other PE routers by using internal BGP (IBPG).
- Provider routers or core routers are any routers in the service provider network that do not attach to CE devices.

With multi-VRF CE, multiple customers can share one CE, and only one physical link is used between the CE and the PE. The shared CE maintains separate VRF tables for each customer and switches or routes packets for each customer based on its own routing table. Multi-VRF CE extends limited PE functionality to a CE device, giving it the ability to maintain separate VRF tables to extend the privacy and security of a VPN to the branch office.

Network Topology

Figure 41-6 shows a configuration using switches as multiple virtual CEs. This scenario is suited for customers who have low bandwidth requirements for their VPN service, for example, small companies. In this case, multi-VRF CE support is required in the switches. Because multi-VRF CE is a Layer 3 feature, each interface in a VRF must be a Layer 3 interface.

Figure 23: Switches Acting as Multiple Virtual CEs



When the CE switch receives a command to add a Layer 3 interface to a VRF, it sets up the appropriate mapping between the VLAN ID and the policy label (PL) in multi-VRF-CE-related data structures and adds the VLAN ID and PL to the VLAN database.

When multi-VRF CE is configured, the Layer 3 forwarding table is conceptually partitioned into two sections:

- The multi-VRF CE routing section contains the routes from different VPNs.
- The global routing section contains routes to non-VPN networks, such as the Internet.

VLAN IDs from different VRFs are mapped into different policy labels, which are used to distinguish the VRFs during processing. For each new VPN route learned, the Layer 3 setup function retrieves the policy label by using the VLAN ID of the ingress port and inserts the policy label and new route to the multi-VRF CE routing section. If the packet is received from a routed port, the port internal VLAN ID number is used; if the packet is received from an SVI, the VLAN number is used.

Packet-Forwarding Process

This is the packet-forwarding process in a multi-VRF-CE-enabled network:

- When the switch receives a packet from a VPN, the switch looks up the routing table based on the input policy label number. When a route is found, the switch forwards the packet to the PE.
- When the ingress PE receives a packet from the CE, it performs a VRF lookup. When a route is found, the router adds a corresponding MPLS label to the packet and sends it to the MPLS network.
- When an egress PE receives a packet from the network, it strips the label and uses the label to identify the correct VPN routing table. Then it performs the normal route lookup. When a route is found, it forwards the packet to the correct adjacency.
- When a CE receives a packet from an egress PE, it uses the input policy label to look up the correct VPN routing table. If a route is found, it forwards the packet within the VPN.

Network Components

To configure VRF, you create a VRF table and specify the Layer 3 interface associated with the VRF. Then configure the routing protocols in the VPN and between the CE and the PE. BGP is the preferred routing protocol used to distribute VPN routing information across the provider's backbone. The multi-VRF CE network has three major components:

- VPN route target communities—lists of all other members of a VPN community. You need to configure VPN route targets for each VPN community member.
- Multiprotocol BGP peering of VPN community PE routers—propagates VRF reachability information to all members of a VPN community. You need to configure BGP peering in all PE routers within a VPN community.
- VPN forwarding—transports all traffic between all VPN community members across a VPN service-provider network.

VRF-Aware Services

IP services can be configured on global interfaces, and these services run within the global routing instance. IP services are enhanced to run on multiple routing instances; they are VRF-aware. Any configured VRF in the system can be specified for a VRF-aware service.

VRF-Aware services are implemented in platform-independent modules. VRF means multiple routing instances in Cisco IOS. Each platform has its own limit on the number of VRFs it supports.

VRF-aware services have the following characteristics:

- The user can ping a host in a user-specified VRF.
- ARP entries are learned in separate VRFs. The user can display Address Resolution Protocol (ARP) entries for specific VRFs.

How to Configure Multi-VRF CE

Default Multi-VRF CE Configuration

Table 90: Default VRF Configuration

Feature	Default Setting
VRF	Disabled. No VRFs are defined.
Maps	No import maps, export maps, or route maps are defined.
VRF maximum routes	Fast Ethernet switches: 8000 Gigabit Ethernet switches: 12000.
Forwarding table	The default for an interface is the global routing table.

Multi-VRF CE Configuration Guidelines

Note To use multi-VRF CE, you must have the IP services or advanced IP services feature set enabled on your switch.

- A switch with multi-VRF CE is shared by multiple customers, and each customer has its own routing table.
- Because customers use different VRF tables, the same IP addresses can be reused. Overlapped IP addresses are allowed in different VPNs.
- Multi-VRF CE lets multiple customers share the same physical link between the PE and the CE. Trunk ports with multiple VLANs separate packets among customers. Each customer has its own VLAN.
- Multi-VRF CE does not support all MPLS-VRF functionality. It does not support label exchange, LDP adjacency, or labeled packets.
- For the PE router, there is no difference between using multi-VRF CE or using multiple CEs. In Figure 41-6, multiple virtual Layer 3 interfaces are connected to the multi-VRF CE device.
- The switch supports configuring VRF by using physical ports, VLAN SVIs, or a combination of both. The SVIs can be connected through an access port or a trunk port.
- A customer can use multiple VLANs as long as they do not overlap with those of other customers. A customer's VLANs are mapped to a specific routing table ID that is used to identify the appropriate routing tables stored on the switch.
- The switch supports one global network and up to 26 VRFs.
- Most routing protocols (BGP, OSPF, RIP, and static routing) can be used between the CE and the PE. However, we recommend using external BGP (EBGP) for these reasons:
    - BGP does not require multiple algorithms to communicate with multiple CEs.
    - BGP is designed for passing routing information between systems run by different administrations.
    - BGP makes it easy to pass attributes of the routes to the CE.
- Multi-VRF CE does not affect the packet switching rate.
- VPN multicast is not supported.
- You can enable VRF on a private VLAN, and the reverse.
- You cannot enable VRF when policy-based routing (PBR) is enabled on an interface, and the reverse.
- You cannot enable VRF when Web Cache Communication Protocol (WCCP) is enabled on an interface, and the reverse.

Configuring VRFs

For complete syntax and usage information for the commands, see the switch command reference for this release and the *Cisco IOS Switching Services Command Reference, Release 12.4*.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip routing** **Example:** `Switch(config)# ip routing`	Enables IP routing.
Step 3	**ip vrf** *vrf-name* **Example:** `Switch(config)# ip vrf vpn1`	Names the VRF, and enter VRF configuration mode.
Step 4	**rd** *route-distinguisher* **Example:** `Switch(config-vrf)# rd 100:2`	Creates a VRF table by specifying a route distinguisher. Enter either an AS number and an arbitrary number (xxx:y) or an IP address and arbitrary number (A.B.C.D:y)
Step 5	**route-target** {**export** | **import** | **both**} *route-target-ext-community* **Example:** `Switch(config-vrf)# route-target both 100:2`	Creates a list of import, export, or import and export route target communities for the specified VRF. Enter either an AS system number and an arbitrary number (xxx:y) or an IP address and an arbitrary number (A.B.C.D:y). The *route-target-ext-community* should be the same as the *route-distinguisher* entered in Step 4.
Step 6	**import map** *route-map* **Example:** `Switch(config-vrf)# import map importmap1`	(Optional) Associates a route map with the VRF.
Step 7	**interface** *interface-id* **Example:** `Switch(config-vrf)# interface gigabitethernet 1/0/1`	Specifies the Layer 3 interface to be associated with the VRF, and enter interface configuration mode. The interface can be a routed port or SVI.

	Command or Action	Purpose
Step 8	**ip vrf forwarding** *vrf-name* **Example:** `Switch(config-if)# ip vrf forwarding vpn1`	Associates the VRF with the Layer 3 interface.
Step 9	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 10	**show ip vrf** [**brief** | **detail** | **interfaces**] [*vrf-name*] **Example:** `Switch# show ip vrf interfaces vpn1`	Verifies the configuration. Displays information about the configured VRFs.
Step 11	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring VRF-Aware Services

These services are VRF-Aware:

- ARP
- Ping
- Simple Network Management Protocol (SNMP)
- Unicast Reverse Path Forwarding (uRPF)
- Syslog
- Traceroute
- FTP and TFTP

Note The switch does not support VRF-aware services for Unicast Reverse Path Forwar......ding (uRPF) or Network Time Protocol (NTP).

Configuring VRF-Aware Services for ARP

For complete syntax and usage information for the commands, see the switch command reference for this release and the *Cisco IOS Switching Services Command Reference, Release 12.4*.

show ip arp vrf vrf-name

DETAILED STEPS

	Command or Action	Purpose
Step 1	**Example:** `Switch# show ip arp vrf vpn1` Displays the ARP table in the specified VRF.	

Configuring VRF-Aware Services for Ping

For complete syntax and usage information for the commands, see the switch command reference for this release and the *Cisco IOS Switching Services Command Reference, Release 12.4*.

DETAILED STEPS

	Command or Action	Purpose
Step 1	ping vrf vrf-name ip-host **Example:** `Switch# ping vrf vpn1 ip-host`	Displays the ARP table in the specified VRF.

Configuring VRF-Aware Services for SNMP

For complete syntax and usage information for the commands, refer to the switch command reference for this release and the *Cisco IOS Switching Services Command Reference, Release 12.4*.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**snmp-server trap authentication vrf** **Example:** `Switch(config)# snmp-server trap authentication vrf`	Enables SNMP traps for packets on a VRF.

	Command or Action	Purpose
Step 3	**snmp-server engineID remote** *host* **vrf** *vpn-instance engine-id string* **Example:** `Switch(config)# snmp-server engineID remote 172.16.20.3 vrf vpn1 80000009030000B064EFE100`	Configures a name for the remote SNMP engine on a switch.
Step 4	**snmp-server host** *host* **vrf** *vpn-instance* **traps** *community* **Example:** `Switch(config)# snmp-server host 172.16.20.3 vrf vpn1 traps comaccess`	Specifies the recipient of an SNMP trap operation and specifies the VRF table to be used for sending SNMP traps.
Step 5	**snmp-server host** *host* **vrf** *vpn-instance* **informs** *community* **Example:** `Switch(config)# snmp-server host 172.16.20.3 vrf vpn1 informs comaccess`	Specifies the recipient of an SNMP inform operation and specifies the VRF table to be used for sending SNMP informs.
Step 6	**snmp-server user** *user group* **remote** *host* **vrf** *vpn-instance security model* **Example:** `Switch(config)# snmp-server user abcd remote 172.16.20.3 vrf vpn1 priv v2c 3des secure3des`	Adds a user to an SNMP group for a remote host on a VRF for SNMP access.
Step 7	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring VRF-Aware Servcies for uRPF

uRPF can be configured on an interface assigned to a VRF, and source lookup is done in the VRF table.

For complete syntax and usage information for the commands, refer to the switch command reference for this release and the *Cisco IOS Switching Services Command Reference, Release 12.4*.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**interface** interface-id Switch(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 3	**no switchport** **Example:** `Switch(config-if)# no switchport`	Removes the interface from Layer 2 configuration mode if it is a physical interface.
Step 4	**ip vrf forwarding** *vrf-name* **Example:** `Switch(config-if)# ip vrf forwarding vpn2`	Configures VRF on the interface.
Step 5	**ip address** *ip-address* **Example:** `Switch(config-if)# ip address 10.1.5.1`	Enters the IP address for the interface.
Step 6	**ip verify unicast reverse-path** **Example:** `Switch(config-if)# ip verify unicast reverse-path`	Enables uRPF on the interface.
Step 7	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Configuring VRF-Aware RADIUS

To configure VRF-Aware RADIUS, you must first enable AAA on a RADIUS server. The switch supports the **ip vrf forwarding** *vrf-name* server-group configuration and the **ip radius source-interface** global configuration commands, as described in the Per VRF AAA Feature Guide.

Configuring VRF-Aware Services for Syslog

For complete syntax and usage information for the commands, refer to the switch command reference for this release and the *Cisco IOS Switching Services Command Reference, Release 12.4*.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**logging on** **Example:** `Switch(config)# logging on`	Enables or temporarily disables logging of storage router event message.
Step 3	**logging host** *ip-address* **vrf** *vrf-name* **Example:** `Switch(config)# logging host 10.10.1.0 vrf vpn1`	Specifies the host address of the syslog server where logging messages are to be sent.
Step 4	**logging buffered** *logging buffered size* **debugging** **Example:** `Switch(config)# logging buffered critical 6000 debugging`	Logs messages to an internal buffer.
Step 5	**logging trap debugging** **Example:** `Switch(config)# logging trap debugging`	Limits the logging messages sent to the syslog server.
Step 6	**logging facility** *facility* **Example:** `Switch(config)# logging facility user`	Sends system logging messages to a logging facility.
Step 7	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring VRF-Aware Services for Traceroute

For complete syntax and usage information for the commands, refer to the switch command reference for this release and the Cisco IOS Switching Services Command Reference, Release 12.4.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**traceroute vrf** *vrf-name ipaddress* **Example:** `Switch(config)# traceroute vrf vpn2 10.10.1.1`	Specifies the name of a VPN VRF in which to find the destination address.

Configuring VRF-Aware Services for FTP and TFTP

So that FTP and TFTP are VRF-aware, you must configure some FTP/TFTP CLIs. For example, if you want to use a VRF table that is attached to an interface, say E1/0, you need to configure the ip tftp source-interface E1/0 or the ip ftp source-interface E1/0 command to inform TFTP or FTP server to use a specific routing table. In this example, the VRF table is used to look up the destination IP address. These changes are backward-compatible and do not affect existing behavior. That is, you can use the source-interface CLI to send packets out a particular interface even if no VRF is configured on that interface.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip ftp source-interface** *interface-type interface-number* **Example:** `Switch(config)# ip ftp source-interface gigabitethernet 1/0/2`	Specifies the source IP address for FTP connections.
Step 3	**end** **Example:** `Switch(config)#end`	Returns to privileged EXEC mode.
Step 4	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 5	**ip tftp source-interface** *interface-type interface-number* **Example:** `Switch(config)# ip tftp source-interface gigabitethernet 1/0/2`	Specifies the source IP address for TFTP connections.
Step 6	**end** **Example:** `Switch(config)#end`	Returns to privileged EXEC mode.

Configuring Multicast VRFs

For complete syntax and usage information for the commands, see the switch command reference for this release and the *Cisco IOS Switching Services Command Reference, Release 12.4*.

For more information about configuring a multicast within a Multi-VRF CE, see the *Cisco IOS IP Multicast Configuration Guide, Release 12.4*.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip routing** **Example:** `Switch(config)# ip routing`	Enables IP routing mode.
Step 3	**ip vrf** *vrf-name* **Example:** `Switch(config)# ip vrf vpn1`	Names the VRF, and enter VRF configuration mode.
Step 4	**rd** *route-distinguisher* **Example:** `Switch(config-vrf)# rd 100:2`	Creates a VRF table by specifying a route distinguisher. Enter either an AS number and an arbitrary number (xxx:y) or an IP address and an arbitrary number (A.B.C.D:y)
Step 5	**route-target** {**export** | **import** | **both**} *route-target-ext-community*	Creates a list of import, export, or import and export route target communities for the specified VRF. Enter either an

	Command or Action	Purpose
	Example: `Switch(config-vrf)# route-target import 100:2`	AS system number and an arbitrary number (xxx:y) or an IP address and an arbitrary number (A.B.C.D:y). The *route-target-ext-community* should be the same as the *route-distinguisher* entered in Step 4.
Step 6	**import map** *route-map* **Example:** `Switch(config-vrf)# import map importmap1`	(Optional) Associates a route map with the VRF.
Step 7	**ip multicast-routing vrf** *vrf-name* **distributed** **Example:** `Switch(config-vrf)# ip multicast-routing vrf vpn1 distributed`	(Optional) Enables global multicast routing for VRF table.
Step 8	**interface** *interface-id* **Example:** `Switch(config-vrf)# interface gigabitethernet 1/0/2`	Specifies the Layer 3 interface to be associated with the VRF, and enter interface configuration mode. The interface can be a routed port or an SVI.
Step 9	**ip vrf forwarding** *vrf-name* **Example:** `Switch(config-if)# ip vrf forwarding vpn1`	Associates the VRF with the Layer 3 interface.
Step 10	**ip address** *ip-address mask* **Example:** `Switch(config-if)# ip address 10.1.5.1 255.255.255.0`	Configures IP address for the Layer 3 interface.
Step 11	**ip pim sparse-dense mode** **Example:** `Switch(config-if)# ip pim sparse-dense mode`	Enables PIM on the VRF-associated Layer 3 interface.
Step 12	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 13	**show ip vrf** [**brief** | **detail** | **interfaces**] [*vrf-name*] **Example:** `Switch# show ip vrf detail vpn1`	Verifies the configuration. Displays information about the configured VRFs.

	Command or Action	Purpose
Step 14	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring a VPN Routing Session

Routing within the VPN can be configured with any supported routing protocol (RIP, OSPF, EIGRP, or BGP) or with static routing. The configuration shown here is for OSPF, but the process is the same for other protocols.

Note To configure an EIGRP routing process to run within a VRF instance, you must configure an autonomous-system number by entering the **autonomous-system** *autonomous-system-number* address-family configuration mode command.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router ospf** *process-id* **vrf** *vrf-name* **Example:** `Switch(config)# router ospf 1 vrf vpn1`	Enables OSPF routing, specifies a VPN forwarding table, and enter router configuration mode.
Step 3	**log-adjacency-changes** **Example:** `Switch(config-router)# log-adjacency-changes`	(Optional) Logs changes in the adjacency state. This is the default state.
Step 4	**redistribute bgp** *autonomous-system-number* **subnets** **Example:** `Switch(config-router)# redistribute bgp 10 subnets`	Sets the switch to redistribute information from the BGP network to the OSPF network.

	Command or Action	Purpose
Step 5	**network** *network-number* **area** *area-id* **Example:** `Switch(config-router)# network 1 area 2`	Defines a network address and mask on which OSPF runs and the area ID for that network address.
Step 6	**end** **Example:** `Switch(config-router)# end`	Returns to privileged EXEC mode.
Step 7	**show ip ospf** *process-id* **Example:** `Switch# show ip ospf 1`	Verifies the configuration of the OSPF network.
Step 8	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring BGP PE to CE Routing Sessions

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router bgp** *autonomous-system-number* **Example:** `Switch(config)# router bgp 2`	Configures the BGP routing process with the AS number passed to other BGP routers, and enter router configuration mode.
Step 3	**network** *network-number* **mask** *network-mask* **Example:** `Switch(config-router)# network 5 mask 255.255.255.0`	Specifies a network and mask to announce using BGP.

	Command or Action	Purpose
Step 4	**redistribute ospf** *process-id* **match internal** **Example:** `Switch(config-router)# redistribute ospf 1 match internal`	Sets the switch to redistribute OSPF internal routes.
Step 5	**network** *network-number* **area** *area-id* **Example:** `Switch(config-router)# network 5 area 2`	Defines a network address and mask on which OSPF runs and the area ID for that network address.
Step 6	**address-family ipv4 vrf** *vrf-name* **Example:** `Switch(config-router)# address-family ipv4 vrf vpn1`	Defines BGP parameters for PE to CE routing sessions, and enter VRF address-family mode.
Step 7	**neighbor** *address* **remote-as** *as-number* **Example:** `Switch(config-router)# neighbor 10.1.1.2 remote-as 2`	Defines a BGP session between PE and CE routers.
Step 8	**neighbor** *address* **activate** **Example:** `Switch(config-router)# neighbor 10.2.1.1 activate`	Activates the advertisement of the IPv4 address family.
Step 9	**end** **Example:** `Switch(config-router)# end`	Returns to privileged EXEC mode.
Step 10	**show ip bgp** [**ipv4**] [**neighbors**] **Example:** `Switch# show ip bgp ipv4 neighbors`	Verifies BGP configuration.
Step 11	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring Multi-VRF CE

Table 91: Commands for Displaying Multi-VRF CE Information

show ip protocols vrf *vrf-name*	Displays routing protocol information associated with a VRF.
show ip route vrf *vrf-name* [**connected**] [*protocol* [*as-number*]] [**list**] [**mobile**] [**odr**] [**profile**] [**static**] [**summary**] [**supernets-only**]	Displays IP routing table information associated with a VRF.
show ip vrf [**brief** | **detail** | **interfaces**] [*vrf-name*]	Displays information about the defined VRF instances.

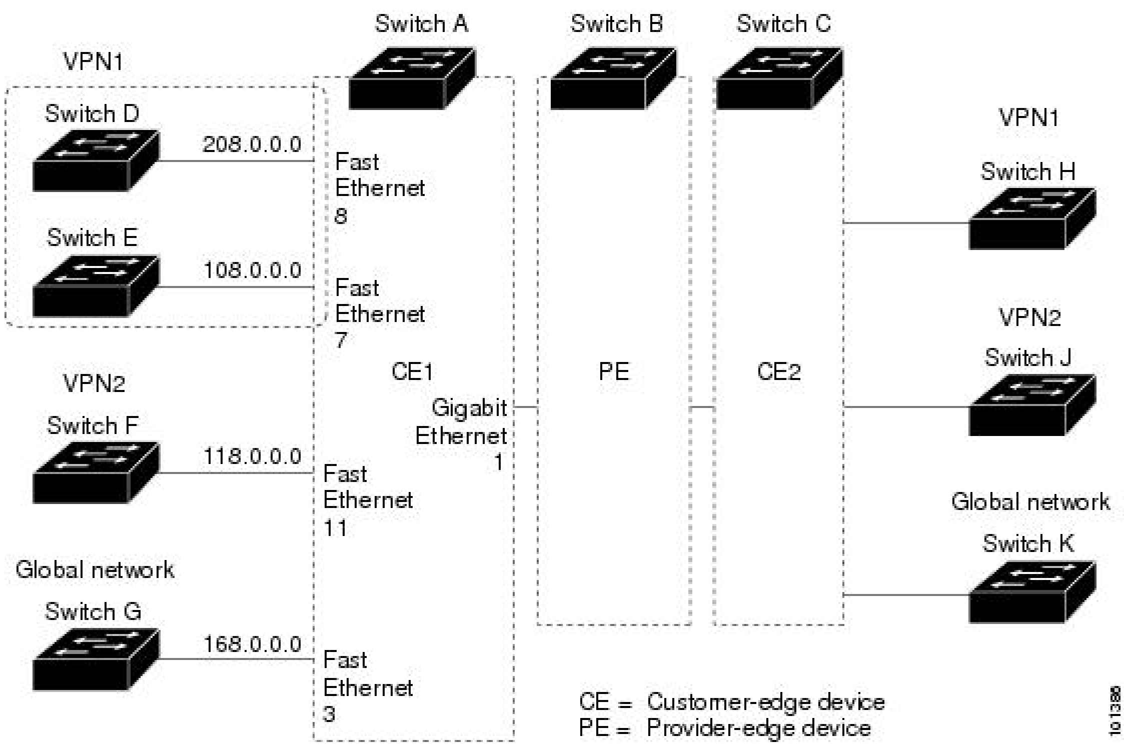
For more information about the information in the displays, see the *Cisco IOS Switching Services Command Reference, Release 12.4*.

Configuration Examples for Multi-VRF CE

Multi-VRF CE Configuration Example

Figure 41-7 is a simplified example of the physical connections in a network similar to that in Figure 41-6. OSPF is the protocol used in VPN1, VPN2, and the global network. BGP is used in the CE to PE connections. The examples following the illustration show how to configure a switch as CE Switch A, and the VRF configuration for customer switches D and F. Commands for configuring CE Switch C and the other customer switches are not included but would be similar. The example also includes commands for configuring traffic to Switch A for a Catalyst 6000 or Catalyst 6500 switch acting as a PE router.

Figure 24: Multi-VRF CE Configuration Example



On Switch A, enable routing and configure VRF.

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip routing
Switch(config)# ip vrf v11
Switch(config-vrf)# rd 800:1
Switch(config-vrf)# route-target export 800:1
Switch(config-vrf)# route-target import 800:1
Switch(config-vrf)# exit
Switch(config)# ip vrf v12
Switch(config-vrf)# rd 800:2
Switch(config-vrf)# route-target export 800:2
Switch(config-vrf)# route-target import 800:2
Switch(config-vrf)# exit
```

Configure the loopback and physical interfaces on Switch A. Gigabit Ethernet port 1 is a trunk connection to the PE. Gigabit Ethernet ports 8 and 11 connect to VPNs:

```
Switch(config)# interface loopback1
Switch(config-if)# ip vrf forwarding v11
Switch(config-if)# ip address 8.8.1.8 255.255.255.0
Switch(config-if)# exit

Switch(config)# interface loopback2
Switch(config-if)# ip vrf forwarding v12
Switch(config-if)# ip address 8.8.2.8 255.255.255.0
Switch(config-if)# exit

Switch(config)# interface gigabitethernet1/0/5
```

```
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# no ip address
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/8
Switch(config-if)# switchport access vlan 208
Switch(config-if)# no ip address
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/11
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# no ip address
Switch(config-if)# exit
```

Configure the VLANs used on Switch A. VLAN 10 is used by VRF 11 between the CE and the PE. VLAN 20 is used by VRF 12 between the CE and the PE. VLANs 118 and 208 are used for the VPNs that include Switch F and Switch D, respectively:

```
Switch(config)# interface vlan10
Switch(config-if)# ip vrf forwarding v11
Switch(config-if)# ip address 38.0.0.8 255.255.255.0
Switch(config-if)# exit
Switch(config)# interface vlan20
Switch(config-if)# ip vrf forwarding v12
Switch(config-if)# ip address 83.0.0.8 255.255.255.0
Switch(config-if)# exit
Switch(config)# interface vlan118
Switch(config-if)# ip vrf forwarding v12
Switch(config-if)# ip address 118.0.0.8 255.255.255.0
Switch(config-if)# exit
Switch(config)# interface vlan208
Switch(config-if)# ip vrf forwarding v11
Switch(config-if)# ip address 208.0.0.8 255.255.255.0
Switch(config-if)# exit
```

Configure OSPF routing in VPN1 and VPN2.

```
Switch(config)# router ospf 1 vrf vl1
Switch(config-router)# redistribute bgp 800 subnets
Switch(config-router)# network 208.0.0.0 0.0.0.255 area 0
Switch(config-router)# exit
Switch(config)# router ospf 2 vrf vl2
Switch(config-router)# redistribute bgp 800 subnets
Switch(config-router)# network 118.0.0.0 0.0.0.255 area 0
Switch(config-router)# exit
```

Configure BGP for CE to PE routing.

```
Switch(config)# router bgp 800
Switch(config-router)# address-family ipv4 vrf vl2
Switch(config-router-af)# redistribute ospf 2 match internal
Switch(config-router-af)# neighbor 83.0.0.3 remote-as 100
Switch(config-router-af)# neighbor 83.0.0.3 activate
Switch(config-router-af)# network 8.8.2.0 mask 255.255.255.0
Switch(config-router-af)# exit
Switch(config-router)# address-family ipv4 vrf vl1
Switch(config-router-af)# redistribute ospf 1 match internal
Switch(config-router-af)# neighbor 38.0.0.3 remote-as 100
Switch(config-router-af)# neighbor 38.0.0.3 activate
Switch(config-router-af)# network 8.8.1.0 mask 255.255.255.0
Switch(config-router-af)# end
```

Switch D belongs to VPN 1. Configure the connection to Switch A by using these commands.

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip routing
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# no switchport
Switch(config-if)# ip address 208.0.0.20 255.255.255.0
Switch(config-if)# exit
```

```
Switch(config)# router ospf 101
Switch(config-router)# network 208.0.0.0 0.0.0.255 area 0
Switch(config-router)# end
```

Switch F belongs to VPN 2. Configure the connection to Switch A by using these commands.

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip routing
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# no ip address
Switch(config-if)# exit

Switch(config)# interface vlan118
Switch(config-if)# ip address 118.0.0.11 255.255.255.0
Switch(config-if)# exit

Switch(config)# router ospf 101
Switch(config-router)# network 118.0.0.0 0.0.0.255 area 0
Switch(config-router)# end
```

When used on switch B (the PE router), these commands configure only the connections to the CE device, Switch A.

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# ip vrf v1
Router(config-vrf)# rd 100:1
Router(config-vrf)# route-target export 100:1
Router(config-vrf)# route-target import 100:1
Router(config-vrf)# exit

Router(config)# ip vrf v2
Router(config-vrf)# rd 100:2
Router(config-vrf)# route-target export 100:2
Router(config-vrf)# route-target import 100:2
Router(config-vrf)# exit
Router(config)# ip cef
Router(config)# interface Loopback1
Router(config-if)# ip vrf forwarding v1
Router(config-if)# ip address 3.3.1.3 255.255.255.0
Router(config-if)# exit

Router(config)# interface Loopback2
Router(config-if)# ip vrf forwarding v2
Router(config-if)# ip address 3.3.2.3 255.255.255.0
Router(config-if)# exit

Router(config)# interface gigabitethernet1/1/0.10
Router(config-if)# encapsulation dot1q 10
Router(config-if)# ip vrf forwarding v1
Router(config-if)# ip address 38.0.0.3 255.255.255.0
Router(config-if)# exit

Router(config)# interface gigabitethernet1/1/0.20
Router(config-if)# encapsulation dot1q 20
Router(config-if)# ip vrf forwarding v2
Router(config-if)# ip address 83.0.0.3 255.255.255.0
Router(config-if)# exit

Router(config)# router bgp 100
Router(config-router)# address-family ipv4 vrf v2
Router(config-router-af)# neighbor 83.0.0.8 remote-as 800
Router(config-router-af)# neighbor 83.0.0.8 activate
Router(config-router-af)# network 3.3.2.0 mask 255.255.255.0
Router(config-router-af)# exit
Router(config-router)# address-family ipv4 vrf vl
Router(config-router-af)# neighbor 38.0.0.8 remote-as 800
Router(config-router-af)# neighbor 38.0.0.8 activate
```

```
Router(config-router-af)# network 3.3.1.0 mask 255.255.255.0
Router(config-router-af)# end
```

Configuring Unicast Reverse Path Forwarding

The unicast reverse path forwarding (unicast RPF) feature helps to mitigate problems that are caused by the introduction of malformed or forged (spoofed) IP source addresses into a network by discarding IP packets that lack a verifiable IP source address. For example, a number of common types of denial-of-service (DoS) attacks, including Smurf and Tribal Flood Network (TFN), can take advantage of forged or rapidly changing source IP addresses to allow attackers to thwart efforts to locate or filter the attacks. For Internet service providers (ISPs) that provide public access, Unicast RPF deflects such attacks by forwarding only packets that have source addresses that are valid and consistent with the IP routing table. This action protects the network of the ISP, its customer, and the rest of the Internet.

Note

- Unicast RPF is supported only in IP services.
- Do not configure unicast RPF if the switch is in a mixed hardware stack combining more than one switch type: Catalyst 3750-X, Catalyst 3750-E, and Catalyst 3750 switches.

For detailed IP unicast RPF configuration information, see the *Other Security Features* chapter in the *Cisco IOS Security Configuration Guide, Release 12.4*.

Protocol-Independent Features

This section describes IP routing protocol-independent features that are available on switches running the IP base or the IP services feature set; except that with the IP base feature set, protocol-related features are available only for RIP. For a complete description of the IP routing protocol-independent commands in this chapter, see the "IP Routing Protocol-Independent Commands" chapter of the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.4.*

Distributed Cisco Express Forwarding

Information About Cisco Express Forwarding

Cisco Express Forwarding (CEF) is a Layer 3 IP switching technology used to optimize network performance. CEF implements an advanced IP look-up and forwarding algorithm to deliver maximum Layer 3 switching performance. CEF is less CPU-intensive than fast switching route caching, allowing more CPU processing power to be dedicated to packet forwarding. In a switch stack, the hardware uses distributed CEF (dCEF) in the stack. In dynamic networks, fast switching cache entries are frequently invalidated because of routing changes, which can cause traffic to be process switched using the routing table, instead of fast switched using the route cache. CEF and dCEF use the Forwarding Information Base (FIB) lookup table to perform destination-based switching of IP packets.

The two main components in CEF and dCEF are the distributed FIB and the distributed adjacency tables.

- The FIB is similar to a routing table or information base and maintains a mirror image of the forwarding information in the IP routing table. When routing or topology changes occur in the network, the IP routing table is updated, and those changes are reflected in the FIB. The FIB maintains next-hop address information based on the information in the IP routing table. Because the FIB contains all known routes that exist in the routing table, CEF eliminates route cache maintenance, is more efficient for switching traffic, and is not affected by traffic patterns.
- Nodes in the network are said to be adjacent if they can reach each other with a single hop across a link layer. CEF uses adjacency tables to prepend Layer 2 addressing information. The adjacency table maintains Layer 2 next-hop addresses for all FIB entries.

Because the switch or switch stack uses Application Specific Integrated Circuits (ASICs) to achieve Gigabit-speed line rate IP traffic, CEF or dCEF forwarding applies only to the software-forwarding path, that is, traffic that is forwarded by the CPU.

How to Configure Cisco Express Forwarding

CEF or distributed CEF is enabled globally by default. If for some reason it is disabled, you can re-enable it by using the **ip cef** or **ip cef distributed** global configuration command.

The default configuration is CEF or dCEF enabled on all Layer 3 interfaces. Entering the **no ip route-cache cef** interface configuration command disables CEF for traffic that is being forwarded by software. This command does not affect the hardware forwarding path. Disabling CEF and using the **debug ip packet detail** privileged EXEC command can be useful to debug software-forwarded traffic. To enable CEF on an interface for the software-forwarding path, use the **ip route-cache cef** interface configuration command.

Caution Although the **no ip route-cache cef** interface configuration command to disable CEF on an interface is visible in the CLI, we strongly recommend that you do not disable CEF or dCEF on interfaces except for debugging purposes.

To enable CEF or dCEF globally and on an interface for software-forwarded traffic if it has been disabled:

1. **configure terminal**

   ```
   Switch# configure terminal
   ```
   Enters global configuration mode.

2 **ip cef**

```
Switch(config)# ip cef
```

Enables CEF operation on a non-stacking switch.

Go to Step 4.

3 **ip cef distributed**

```
Switch(config)# ip cef distributed
```

Enables CEF operation on a active switch.

4 **interface** *interface-id*

```
Switch(config)# interface gigabitethernet 1/0/1
```

Enters interface configuration mode, and specifies the Layer 3 interface to configure.

5 **ip route-cache cef**

```
Switch(config-if)# ip route-cache cef
```

Enables CEF on the interface for software-forwarded traffic.

6 **end**

```
Switch(config-if)# end
```

Returns to privileged EXEC mode.

7 **show ip cef**

```
Switch# show ip cef
```

Displays the CEF status on all interfaces.

8 **show cef linecard** [**detail**]

```
Switch# show cef linecard detail
```

(Optional) Displays CEF-related interface information on a non-stacking switch.

9 **show cef linecard** [*slot-number*] [**detail**]

```
Switch# show cef linecard 5 detail
```

(Optional) Displays CEF-related interface information on a switch by stack member for all switches in the stack or for the specified switch.

(Optional) For *slot-number*, enter the stack member switch number.

10 **show cef interface** [*interface-id*]

```
Switch# show cef interface gigabitethernet 1/0/1
```

Displays detailed CEF information for all interfaces or the specified interface.

11 **show adjacency**

```
Switch# show adjacency
```

Displays CEF adjacency table information.

12 **copy running-config startup-config**

```
Switch# copy running-config startup-config
```

(Optional) Saves your entries in the configuration file.

Number of Equal-Cost Routing Paths

Information About Equal-Cost Routing Paths

When a router has two or more routes to the same network with the same metrics, these routes can be thought of as having an equal cost. The term parallel path is another way to see occurrences of equal-cost routes in a routing table. If a router has two or more equal-cost paths to a network, it can use them concurrently. Parallel paths provide redundancy in case of a circuit failure and also enable a router to load balance packets over the available paths for more efficient use of available bandwidth. Equal-cost routes are supported across switches in a stack.

Even though the router automatically learns about and configures equal-cost routes, you can control the maximum number of parallel paths supported by an IP routing protocol in its routing table. Although the switch software allows a maximum of 32 equal-cost routes, the switch hardware will never use more than 16 paths per route.

How to Configure Equal-Cost Routing Paths

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router** {**rip** | **ospf** | **eigrp**} **Example:** `Switch(config)# router eigrp`	Enters router configuration mode.
Step 3	**maximum-paths** *maximum* **Example:** `Switch(config-router)# maximum-paths 2`	Sets the maximum number of parallel paths for the protocol routing table. The range is from 1 to 16; the default is 4 for most IP routing protocols, but only 1 for BGP.
Step 4	**end** **Example:** `Switch(config-router)# end`	Returns to privileged EXEC mode.
Step 5	**show ip protocols** **Example:** `Switch# show ip protocols`	Verifies the setting in the *Maximum path* field.

	Command or Action	Purpose
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Static Unicast Routes

Information About Static Unicast Routes

Static unicast routes are user-defined routes that cause packets moving between a source and a destination to take a specified path. Static routes can be important if the router cannot build a route to a particular destination and are useful for specifying a gateway of last resort to which all unroutable packets are sent.

The switch retains static routes until you remove them. However, you can override static routes with dynamic routing information by assigning administrative distance values. Each dynamic routing protocol has a default administrative distance, as listed in Table 41-16. If you want a static route to be overridden by information from a dynamic routing protocol, set the administrative distance of the static route higher than that of the dynamic protocol.

Table 92: Dynamic Routing Protocol Default Administrative Distances

Route Source	Default Distance
Connected interface	0
Static route	1
Enhanced IRGP summary route	5
External BGP	20
Internal Enhanced IGRP	90
IGRP	100
OSPF	110
Internal BGP	200
Unknown	225

Static routes that point to an interface are advertised through RIP, IGRP, and other dynamic routing protocols, whether or not static **redistribute** router configuration commands were specified for those routing protocols. These static routes are advertised because static routes that point to an interface are considered in the routing table to be connected and hence lose their static nature. However, if you define a static route to an interface that is not one of the networks defined in a network command, no dynamic routing protocols advertise the route unless a **redistribute** static command is specified for these protocols.

When an interface goes down, all static routes through that interface are removed from the IP routing table. When the software can no longer find a valid next hop for the address specified as the forwarding router's address in a static route, the static route is also removed from the IP routing table.

Configuring Static Unicast Routes

Static unicast routes are user-defined routes that cause packets moving between a source and a destination to take a specified path. Static routes can be important if the router cannot build a route to a particular destination and are useful for specifying a gateway of last resort to which all unroutable packets are sent.

Beginning in privileged EXEC mode, follow these steps to configure a static route:

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip route prefix mask** {*address* | *interface*} [*distance*] **Example:** `Switch(config)# ip route prefix mask gigabitethernet 1/0/4`	Establish a static route.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show ip route** **Example:** `Switch# show ip route`	Displays the current state of the routing table to verify the configuration.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Default Routes and Networks

Information About Default Routes and Networks

A router might not be able to learn the routes to all other networks. To provide complete routing capability, you can use some routers as smart routers and give the remaining routers default routes to the smart router. (Smart routers have routing table information for the entire internetwork.) These default routes can be dynamically learned or can be configured in the individual routers. Most dynamic interior routing protocols include a mechanism for causing a smart router to generate dynamic default information that is then forwarded to other routers.

If a router has a directly connected interface to the specified default network, the dynamic routing protocols running on that device generate a default route. In RIP, it advertises the pseudonetwork 0.0.0.0.

A router that is generating the default for a network also might need a default of its own. One way a router can generate its own default is to specify a static route to the network 0.0.0.0 through the appropriate device.

When default information is passed through a dynamic routing protocol, no further configuration is required. The system periodically scans its routing table to choose the optimal default network as its default route. In IGRP networks, there might be several candidate networks for the system default. Cisco routers use administrative distance and metric information to set the default route or the gateway of last resort.

If dynamic default information is not being passed to the system, candidates for the default route are specified with the **ip default-network** global configuration command. If this network appears in the routing table from any source, it is flagged as a possible choice for the default route. If the router has no interface on the default network, but does have a path to it, the network is considered as a possible candidate, and the gateway to the best default path becomes the gateway of last resort.

How to Configure Default Routes and Networks

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip default-network** *network number* **Example:** `Switch(config)# ip default-network 1`	Specifies a default network.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 4	**show ip route** **Example:** `Switch# show ip route`	Displays the selected default route in the gateway of last resort display.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Route Maps to Redistribute Routing Information

Information About Route Maps

The switch can run multiple routing protocols simultaneously, and it can redistribute information from one routing protocol to another. Redistributing information from one routing protocol to another applies to all supported IP-based routing protocols.

You can also conditionally control the redistribution of routes between routing domains by defining enhanced packet filters or route maps between the two domains. The **match** and **set** route-map configuration commands define the condition portion of a route map. The **match** command specifies that a criterion must be matched. The **set** command specifies an action to be taken if the routing update meets the conditions defined by the match command. Although redistribution is a protocol-independent feature, some of the **match** and **set** route-map configuration commands are specific to a particular protocol.

One or more **match** commands and one or more **set** commands follow a **route-map** command. If there are no **match** commands, everything matches. If there are no **set** commands, nothing is done, other than the match. Therefore, you need at least one **match** or **set** command.

Note A route map with no **set** route-map configuration commands is sent to the CPU, which causes high CPU utilization.

You can also identify route-map statements as **permit** or **deny**. If the statement is marked as a deny, the packets meeting the match criteria are sent back through the normal forwarding channels (destination-based routing). If the statement is marked as permit, set clauses are applied to packets meeting the match criteria. Packets that do not meet the match criteria are forwarded through the normal routing channel.

You can use the BGP route map **continue** clause to execute additional entries in a route map after an entry is executed with successful match and set clauses. You can use the **continue** clause to configure and organize more modular policy definitions so that specific policy configurations need not be repeated within the same route map. The switch supports the **continue** clause for outbound policies. For more information about using the route map **continue** clause, see the BGP Route-Map Continue Support for an Outbound Policy feature guide for Cisco IOS Release 12.4(4)T.

How to Configure a Route Map

Although each of Steps 3 through 14 in the following section is optional, you must enter at least one **match** route-map configuration command and one **set** route-map configuration command.

Note The keywords are the same as defined in the procedure to control the route distribution.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**route-map** map-tag [**permit** | **deny**] [sequence number] **Example:** `Switch(config)# route-map rip-to-ospf permit 4`	Defines any route maps used to control redistribution and enter route-map configuration mode. *map-tag*—A meaningful name for the route map. The **redistribute** router configuration command uses this name to reference this route map. Multiple route maps might share the same map tag name. (Optional) If **permit** is specified and the match criteria are met for this route map, the route is redistributed as controlled by the set actions. If **deny** is specified, the route is not redistributed. *sequence number* (Optional)— Number that indicates the position a new route map is to have in the list of route maps already configured with the same name.
Step 3	**match as-path** *path-list-number* **Example:** `Switch(config-route-map)#match as-path 10`	Matches a BGP AS path access list.
Step 4	**match community-list** *community-list-number* [**exact**] **Example:** `Switch(config-route-map)# match community-list 150`	Matches a BGP community list.
Step 5	**match ip address** {*access-list-number* | *access-list-name*} [*...access-list-number* | *...access-list-name*]	Matches a standard access list by specifying the name or number. It can be an integer from 1 to 199.

	Command or Action	Purpose
	Example: `Switch(config-route-map)# match ip address 5 80`	
Step 6	**match metric** metric-value **Example:** `Switch(config-route-map)# match metric 2000`	Matches the specified route metric. The *metric-value* can be an EIGRP metric with a specified value from 0 to 4294967295.
Step 7	**match ip next-hop** {*access-list-number* | *access-list-name*} [*...access-list-number* | *...access-list-name*] **Example:** `Switch(config-route-map)# match ip next-hop 8 45`	Matches a next-hop router address passed by one of the access lists specified (numbered from 1 to 199).
Step 8	**match tag** tag value [...tag-value] **Example:** `Switch(config-route-map)# match tag 3500`	Matches the specified tag value in a list of one or more route tag values. Each can be an integer from 0 to 4294967295.
Step 9	**match interface** type number [...type number] **Example:** `Switch(config-route-map)# match interface gigabitethernet 1/0/1`	Matches the specified next hop route out one of the specified interfaces.
Step 10	**match ip route-source** {*access-list-number* | *access-list-name*} [*...access-list-number* | *...access-list-name*] **Example:** `Switch(config-route-map)# match ip route-source 10 30`	Matches the address specified by the specified advertised access lists.
Step 11	**match route-type** {**local** | **internal** | **external** [**type-1** | **type-2**]} **Example:** `Switch(config-route-map)# match route-type local`	Matches the specified **route-type**: • **local—**Locally generated BGP routes. • **internal—**OSPF intra-area and interarea routes or EIGRP internal routes. • **external—**OSPF external routes (Type 1 or Type 2) or EIGRP external routes.
Step 12	**set dampening** *halflife reuse suppress max-suppress-time*	Sets BGP route dampening factors.

	Command or Action	Purpose
	Example: `Switch(config-route-map)# set dampening 30 1500 10000 120`	
Step 13	**set local-preference** *value* **Example:** `Switch(config-route-map)# set local-preference 100`	Assigns a value to a local BGP path.
Step 14	**set origin** {**igp** | **egp** *as* | **incomplete**} **Example:** `Switch(config-route-map)#set origin igp`	Sets the BGP origin code.
Step 15	**set as-path** {**tag** | **prepend** *as-path-string*} **Example:** `Switch(config-route-map)# set as-path tag`	Modifies the BGP autonomous system path.
Step 16	**set level** {**level-1** | **level-2** | **level-1-2** | **stub-area** | **backbone**} **Example:** `Switch(config-route-map)# set level level-1-2`	Sets the level for routes that are advertised into the specified area of the routing domain. The **stub-area** and **backbone** are OSPF NSSA and backbone areas.
Step 17	**set metric** metric value **Example:** `Switch(config-route-map)# set metric 100`	Sets the metric value to give the redistributed routes (for EIGRP only). The *metric value* is an integer from -294967295 to 294967295.
Step 18	**set metric** bandwidth delay reliability loading mtu **Example:** `Switch(config-route-map)# set metric 10000 10 255 1 1500`	Sets the metric value to give the redistributed routes (for EIGRP only): • *bandwidth*—Metric value or IGRP bandwidth of the route in kilobits per second in the range 0 to 4294967295 • *delay*—Route delay in tens of microseconds in the range 0 to 4294967295. • *reliability*—Likelihood of successful packet transmission expressed as a number between 0 and 255, where 255 means 100 percent reliability and 0 means no reliability. • *loading*—Effective bandwidth of the route expressed as a number from 0 to 255 (255 is 100 percent loading). • *mtu*—Minimum maximum transmission unit (MTU) size of the route in bytes in the range 0 to 4294967295.

	Command or Action	Purpose
Step 19	**set metric-type** {**type-1** | **type-2**} **Example:** `Switch(config-route-map)# set metric-type type-2`	Sets the OSPF external metric type for redistributed routes.
Step 20	**set metric-type internal** **Example:** `Switch(config-route-map)# set metric-type internal`	Sets the multi-exit discriminator (MED) value on prefixes advertised to external BGP neighbor to match the IGP metric of the next hop.
Step 21	**set weight** *number* **Example:** `Switch(config-route-map)# set weight 100`	Sets the BGP weight for the routing table. The value can be from 1 to 65535.
Step 22	**end** **Example:** `Switch(config-route-map)# end`	Returns to privileged EXEC mode.
Step 23	**show route-map** **Example:** `Switch# show route-map`	Displays all route maps configured or only the one specified to verify configuration.
Step 24	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

How to Control Route Distribution

Although each of Steps 3 through 14 in the following section is optional, you must enter at least one **match** route-map configuration command and one **set** route-map configuration command.

Note The keywords are the same as defined in the procedure to configure the route map for redistritbution.

The metrics of one routing protocol do not necessarily translate into the metrics of another. For example, the RIP metric is a hop count, and the IGRP metric is a combination of five qualities. In these situations, an artificial metric is assigned to the redistributed route. Uncontrolled exchanging of routing information between different routing protocols can create routing loops and seriously degrade network operation.

If you have not defined a default redistribution metric that replaces metric conversion, some automatic metric translations occur between routing protocols:

- RIP can automatically redistribute static routes. It assigns static routes a metric of 1 (directly connected).
- Any protocol can redistribute other routing protocols if a default mode is in effect.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router** { | **rip** | **ospf** | **eigrp**} **Example:** `Switch(config)# router bgp`	Enters router configuration mode.
Step 3	**redistribute** *protocol* [*process-id*] {**level-1** | **level-1-2** | **level-2**} [**metric** *metric-value*] [**metric-type** *type-value*] [**match internal** | **external** *type-value*] [**tag** *tag-value*] [**route-map** *map-tag*] [**weight** *weight*] [**subnets**] **Example:** `Switch(config-router)# redistribute bgp 300 level-1-2 route-map bgp-to-ospf`	Redistributes routes from one routing protocol to another routing protocol. If no route-maps are specified, all routes are redistributed. If the keyword **route-map** is specified with no *map-tag*, no routes are distributed.
Step 4	**default-metric** *number* **Example:** `Switch(config-router)# default-metric 1024`	Cause the current routing protocol to use the same metric value for all redistributed routes (BGP, RIP and OSPF).
Step 5	**default-metric** bandwidth delay reliability loading mtu **Example:** `Switch(config-router)# default-metric 1000 100 250 100 1500`	Cause the EIGRP routing protocol to use the same metric value for all non-EIGRP redistributed routes.
Step 6	**end** **Example:** `Switch(config-router)# end`	Returns to privileged EXEC mode.
Step 7	**show route-map** **Example:** `Switch# show route-map`	Displays all route maps configured or only the one specified to verify configuration.

	Command or Action	Purpose
Step 8	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Policy-Based Routing

Information About Policy-Based Routing

You can use policy-based routing (PBR) to configure a defined policy for traffic flows. By using PBR, you can have more control over routing by reducing the reliance on routes derived from routing protocols. PBR can specify and implement routing policies that allow or deny paths based on:

- Identity of a particular end system
- Application
- Protocol

You can use PBR to provide equal-access and source-sensitive routing, routing based on interactive versus batch traffic, or routing based on dedicated links. For example, you could transfer stock records to a corporate office on a high-bandwidth, high-cost link for a short time while transmitting routine application data such as e-mail over a low-bandwidth, low-cost link.

With PBR, you classify traffic using access control lists (ACLs) and then make traffic go through a different path. PBR is applied to incoming packets. All packets received on an interface with PBR enabled are passed through route maps. Based on the criteria defined in the route maps, packets are forwarded (routed) to the appropriate next hop.

- Route map statement marked as permit is processed as follows:
    - A match command can match on length or multiple ACLs. A route map statement can contain multiple match commands. Logical or algorithm function is performed across all the match commands to reach a permit or deny decision.
      For example:

      match length A B

      match ip address acl1 acl2

      match ip address acl3

  A packet is permitted if it is permitted by match length A B or acl1 or acl2 or acl3

    - If the decision reached is permit, then the action specified by the set command is applied on the packet .
    - If the decision reached is deny, then the PBR action (specified in the set command) is not applied. Instead the processing logic moves forward to look at the next route-map statement in the sequence

(the statement with the next higher sequence number). If no next statement exists, PBR processing terminates, and the packet is routed using the default IP routing table.

- For PBR, route-map statements marked as deny are not supported.

For more information about configuring route maps, see the “Using Route Maps to Redistribute Routing Information” section.

You can use standard IP ACLs to specify match criteria for a source address or extended IP ACLs to specify match criteria based on an application, a protocol type, or an end station. The process proceeds through the route map until a match is found. If no match is found, normal destination-based routing occurs. There is an implicit deny at the end of the list of match statements.

If match clauses are satisfied, you can use a set clause to specify the IP addresses identifying the next hop router in the path.

For details about PBR commands and keywords, see the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols* .

How to Configure PBR

- To use PBR, you must have the IP Base feature set enabled on the switch or stack master.
- Multicast traffic is not policy-routed. PBR applies to only to unicast traffic.
- You can enable PBR on a routed port or an SVI.
- The switch supports PBR based on match length.
- You can apply a policy route map to an EtherChannel port channel in Layer 3 mode, but you cannot apply a policy route map to a physical interface that is a member of the EtherChannel. If you try to do so, the command is rejected. When a policy route map is applied to a physical interface, that interface cannot become a member of an EtherChannel.
- You can define a maximum of 128 IP policy route maps on the switch or switch stack.
- You can define a maximum of 512 access control entries (ACEs) for PBR on the switch or switch stack.
- When configuring match criteria in a route map, follow these guidelines:
    - Do not match ACLs that permit packets destined for a local address. PBR would forward these packets, which could cause ping or Telnet failure or route protocol flappping.
- VRF and PBR are mutually exclusive on a switch interface. You cannot enable VRF when PBR is enabled on an interface. The reverse is also true, you cannot enable PBR when VRF is enabled on an interface.
- Web Cache Communication Protocol (WCCP) and PBR are mutually exclusive on a switch interface. You cannot enable WCCP when PBR is enabled on an interface. The reverse is also true, you cannot enable PBR when WCCP is enabled on an interface.
- The number of hardware entries used by PBR depends on the route map itself, the ACLs used, and the order of the ACLs and route-map entries.
- PBR based on TOS, DSCP & IP Precedence are not supported.
- Set interface, set default next-hop and set default interface are not supported.

- Policy-maps with no set actions are supported. Matching packets are routed normally.
- Policy-maps with no match clauses are supported. Set actions are applied to all packets.

By default, PBR is disabled on the switch. To enable PBR, you must create a route map that specifies the match criteria and the resulting action. Then, you must enable PBR for that route map on an interface. All packets arriving on the specified interface matching the match clauses are subject to PBR.

Packets that are generated by the switch, or local packets, are not normally policy-routed. When you globally enable local PBR on the switch, all packets that originate on the switch are subject to local PBR. Local PBR is disabled by default.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**route-map** map-tag [**permit**] [sequence number] **Example:** `Switch(config)# route-map pbr-map permit`	Define any route maps used to control where packets are output, and enter route-map configuration mode. • *map-tag*—A meaningful name for the route map. The**ip policy route-map**interface configuration command uses this name to reference the route map. Multiple route-map statements with the same map-tag define a single route-map. • (Optional) If **permit** is specified and the match criteria are met for this route map, the route is policy-routed as controlled by the set actions. • *sequence number* (Optional)— Number that shows the position of the route-map statement in the given route-map.
Step 3	**match ip address** {*access-list-number* | *access-list-name*} [*access-list-number* |*...access-list-name*] **Example:** `Switch(config-route-map)# match ip address 110 140`	Match the source and destination IP address that is permitted by one or more standard or extended access lists. ACLs can match on more than source and destination IP addresses. If you do not specify a **match** command, the route map applies to all packets.
Step 4	**match length min max** **Example:** `Switch(config-route-map)# match length 64 1500`	Matches against the length of the packet.
Step 5	**set ip next-hop** *ip-address* [*...ip-address*] **Example:** `Switch(config-route-map)# set ip next-hop 10.1.6.2`	Specifies the action to take on the packets that match the criteria. Sets next hop to which to route the packet (the next hop must be adjacent).

	Command or Action	Purpose
Step 6	**exit** **Example:** `Switch(config-route-map)# exit`	Returns to global configuration mode.
Step 7	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the interface to configure.
Step 8	**ip policy route-map** *map-tag* **Example:** `Switch(config-if)# ip policy route-map pbr-map`	Enables PBR on a Layer 3 interface, and identify the route map to use. You can configure only one route map on an interface. However, you can have multiple route map entries with different sequence numbers. These entries are evaluated in sequence number order until the first match. If there is no match, packets are routed as usual.
Step 9	**ip route-cache policy** **Example:** `Switch(config-if)# ip route-cache policy`	(Optional) Enables fast-switching PBR. You must first enable PBR before enabling fast-switching PBR.
Step 10	**exit** **Example:** `Switch(config-if)# exit`	Returns to global configuration mode.
Step 11	**ip local policy route-map** *map-tag* **Example:** `Switch(config)# ip local policy route-map local-pbr`	(Optional) Enables local PBR to perform policy-based routing on packets originating at the switch. This applies to packets generated by the switch and not to incoming packets.
Step 12	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 13	**show route-map** [*map-name*] **Example:** `Switch# show route-map`	(Optional) Displays all route maps configured or only the one specified to verify configuration.
Step 14	**show ip policy** **Example:** `Switch# show ip policy`	(Optional) Displays policy route maps attached to interface
Step 15	**show ip local policy** **Example:** `Switch# show ip local policy`	(Optional) Displays whether or not local policy routing is enabled and, if so, the route map being used.

Filtering Routing Information

You can filter routing protocol information by performing the tasks described in this section.

Note When routes are redistributed between OSPF processes, no OSPF metrics are preserved.

Setting Passive Interfaces

To prevent other routers on a local network from dynamically learning about routes, you can use the **passive-interface** router configuration command to keep routing update messages from being sent through a router interface. When you use this command in the OSPF protocol, the interface address you specify as passive appears as a stub network in the OSPF domain. OSPF routing information is neither sent nor received through the specified router interface.

In networks with many interfaces, to avoid having to manually set them as passive, you can set all interfaces to be passive by default by using the **passive-interface default** router configuration command and manually setting interfaces where adjacencies are desired.

Use a network monitoring privileged EXEC command such as **show ip ospf interface** to verify the interfaces that you enabled as passive, or use the **show ip interface** privileged EXEC command to verify the interfaces that you enabled as active.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router** { | **rip** | **ospf** | **eigrp**} **Example:** `Switch(config)# router ospf`	Enters router configuration mode.
Step 3	**passive-interface** *interface-id* **Example:** `Switch(config-router)# passive-interface gigabitethernet 1/0/1`	Suppresses sending routing updates through the specified Layer 3 interface.
Step 4	**passive-interface default** **Example:** `Switch(config-router)# passive-interface default`	(Optional) Sets all interfaces as passive by default.

	Command or Action	Purpose
Step 5	**no passive-interface** *interface type* **Example:** `Switch(config-router)# no passive-interface gigabitethernet1/0/3 gigabitethernet 1/0/5`	(Optional) Activates only those interfaces that need to have adjacencies sent.
Step 6	**network** *network-address* **Example:** `Switch(config-router)# network 10.1.1.1`	(Optional) Specifies the list of networks for the routing process. The *network-address* is an IP address.
Step 7	**end** **Example:** `Switch(config-router)# end`	Returns to privileged EXEC mode.
Step 8	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Controlling Advertising and Processing in Routing Updates

You can use the **distribute-list** router configuration command with access control lists to suppress routes from being advertised in routing updates and to prevent other routers from learning one or more routes. When used in OSPF, this feature applies to only external routes, and you cannot specify an interface name.

You can also use a **distribute-list** router configuration command to avoid processing certain routes listed in incoming updates. (This feature does not apply to OSPF.)

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router** { | **rip** | **eigrp**} **Example:** `Switch(config)# router eigrp`	Enters router configuration mode.

	Command or Action	Purpose
Step 3	**distribute-list** {*access-list-number* | *access-list-name*} **out** [*interface-name* | *routing process* | *autonomous-system-number*] **Example:** `Switch(config-router)# distribute 120 out gigabitethernet 1/0/7`	Permits or denies routes from being advertised in routing updates, depending upon the action listed in the access list.
Step 4	**distribute-list** {*access-list-number* | *access-list-name*} **in** [*type-number*] **Example:** `Switch(config-router)# distribute-list 125 in`	Suppresses processing in routes listed in updates.
Step 5	**end** **Example:** `Switch(config-router)# end`	Returns to privileged EXEC mode.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Filtering Sources of Routing Information

Because some routing information might be more accurate than others, you can use filtering to prioritize information coming from different sources. An administrative distance is a rating of the trustworthiness of a routing information source, such as a router or group of routers. In a large network, some routing protocols can be more reliable than others. By specifying administrative distance values, you enable the router to intelligently discriminate between sources of routing information. The router always picks the route whose routing protocol has the lowest administrative distance. Table 41-16 on page 41-122 shows the default administrative distances for various routing information sources.

Because each network has its own requirements, there are no general guidelines for assigning administrative distances.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**router** { | **rip** | **ospf** | **eigrp**} **Example:** `Switch(config)# router bgp`	Enters router configuration mode.
Step 3	**distance** *weight* {*ip-address* {*ip-address mask*}} [*ip access list*] **Example:** `Switch(config-router)# distance 50 10.1.5.1`	Defines an administrative distance. *weight*—The administrative distance as an integer from 10 to 255. Used alone, *weight* specifies a default administrative distance that is used when no other specification exists for a routing information source. Routes with a distance of 255 are not installed in the routing table. (Optional) *ip access list*—An IP standard or extended access list to be applied to incoming routing updates.
Step 4	**end** **Example:** `Switch(config-router)# end`	Returns to privileged EXEC mode.
Step 5	**show ip protocols** **Example:** `Switch# show ip protocols`	Displays the default administrative distance for a specified routing process.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Managing Authentication Keys

Key management is a method of controlling authentication keys used by routing protocols. Not all protocols can use key management. Authentication keys are available for EIGRP and RIP Version 2.

Prerequisites

Before you manage authentication keys, you must enable authentication. See the appropriate protocol section to see how to enable authentication for that protocol. To manage authentication keys, define a key chain, identify the keys that belong to the key chain, and specify how long each key is valid. Each key has its own key identifier (specified with the **key** *number* key chain configuration command), which is stored locally. The combination of the key identifier and the interface associated with the message uniquely identifies the authentication algorithm and Message Digest 5 (MD5) authentication key in use.

How to Configure Authentication Keys

You can configure multiple keys with life times. Only one authentication packet is sent, regardless of how many valid keys exist. The software examines the key numbers in order from lowest to highest, and uses the first valid key it encounters. The lifetimes allow for overlap during key changes. Note that the router must know these lifetimes.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**key chain** *name-of-chain* **Example:** `Switch(config)# key chain key10`	Identifies a key chain, and enter key chain configuration mode.
Step 3	**key** *number* **Example:** `Switch(config-keychain)# key 2000`	Identifies the key number. The range is 0 to 2147483647.
Step 4	**key-string** *text* **Example:** `Switch(config-keychain)# Room 20, 10th floor`	Identifies the key string. The string can contain from 1 to 80 uppercase and lowercase alphanumeric characters, but the first character cannot be a number.
Step 5	**accept-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*} **Example:** `Switch(config-keychain)# accept-lifetime 12:30:00 Jan 25 1009 infinite`	(Optional) Specifies the time period during which the key can be received. The *start-time* and *end-time* syntax can be either *hh:mm:ss Month date year* or *hh:mm:ss date Month year*. The default is forever with the default *start-time* and the earliest acceptable date as January 1, 1993. The default *end-time* and **duration** is **infinite**.
Step 6	**send-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*} **Example:** `Switch(config-keychain)# accept-lifetime 23:30:00 Jan 25 1019 infinite`	(Optional) Specifies the time period during which the key can be sent. The *start-time* and *end-time* syntax can be either *hh:mm:ss Month date year* or *hh:mm:ss date Month year*. The default is forever with the default *start-time* and the earliest acceptable date as January 1, 1993. The default *end-time* and **duration** is **infinite**.

	Command or Action	Purpose
Step 7	**end** **Example:** `Switch(config-keychain)# end`	Returns to privileged EXEC mode.
Step 8	**show key chain** **Example:** `Switch# show key chain`	Displays authentication key information.
Step 9	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring and Maintaining the IP Network

You can remove all contents of a particular cache, table, or database. You can also display specific statistics.

Table 93: Commands to Clear IP Routes or Display Route Status

show ip route summary	Displays the current state of the routing table in summary form.

PART VI

Multicast

CHAPTER 38

Configuring IGMP

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Restrictions for Configuring IGMP

The following are the restrictions for configuring IGMP:

- The switch supports IGMP Versions 1, 2 , and 3.

Note For IGMP Version 3, only IGMP Version 3 BISS (Basic IGMPv3 Snooping Support) is supported.

- IGMP Version 3 uses new membership report messages that might not be correctly recognized by older IGMP snooping switches.
- IGMP filtering and throttling is not supported under the WLAN.
- You cannot have a switch stack containing a mix of Catalyst 3850 and Catalyst 3650 switches.

Information About IGMP

To participate in IP multicasting, multicast hosts, routers, and multilayer switches must have the Internet Group Management Protocol (IGMP) operating. This protocol defines the querier and host roles:

- A querier is a network device that sends query messages to discover which network devices are members of a given multicast group.
- A host is a receiver that sends report messages (in response to query messages) to inform a querier of a host membership.

A set of queriers and hosts that receive multicast data streams from the same source is called a multicast group. Queriers and hosts use IGMP messages to join and leave multicast groups.

Any host, regardless of whether it is a member of a group, can send to a group. However, only the members of a group receive the message. Membership in a multicast group is dynamic; hosts can join and leave at any time. There is no restriction on the location or number of members in a multicast group. A host can be a member of more than one multicast group at a time. How active a multicast group is and what members it has can vary from group to group and from time to time. A multicast group can be active for a long time, or it can be very short-lived. Membership in a group can constantly change.

IP Multicast Group Addresses

IP multicast traffic uses group addresses, which are class D addresses. The high-order bits of a Class D address are 1110. Therefore, host group addresses can be in the range 224.0.0.0 through 239.255.255.255. Multicast addresses in the range 224.0.0.0 to 224.0.0.255 are reserved for use by routing protocols and other network control traffic. The address 224.0.0.0 is guaranteed not to be assigned to any group.

IGMP packets are sent using these IP multicast group addresses:

- IGMP general queries are destined to the address 224.0.0.1 (all systems on a subnet).
- IGMP group-specific queries are destined to the group IP address for which the switch is querying.
- IGMP group membership reports are destined to the group IP address for which the switch is reporting.
- IGMP Version 2 (IGMPv2) leave messages are destined to the address 224.0.0.2 (all multicast routers on a subnet). In some old host IP stacks, leave messages might be destined to the group IP address rather than to the all-routers address.

Related Topics

Configuring the Switch as a Member of a Group , on page 796

Example: Configuring the Switch as a Member of a Multicast Group, on page 832

IGMP Versions

The switch supports IGMP version 1, IGMP version 2, and IGMP version 3. These versions are interoperable on the switch. For example, if IGMP snooping is enabled and the querier's version is IGMPv2, and the switch receives an IGMPv3 report from a host, then the switch can forward the IGMPv3 report to the multicast router.

IGMP Version 1

IGMP version 1 (IGMPv1) primarily uses a query-response model that enables the multicast router and multilayer switch to find which multicast groups are active (have one or more hosts interested in a multicast group) on the local subnet. IGMPv1 has other processes that enable a host to join and leave a multicast group. For more information, see RFC 1112.

IGMP Version 2

IGMPv2 extends IGMP functionality by providing such features as the IGMP leave process to reduce leave latency, group-specific queries, and an explicit maximum query response time. IGMPv2 also adds the capability for routers to elect the IGMP querier without depending on the multicast protocol to perform this task. For more information, see RFC 2236.

Note IGMP version 2 is the default version for the switch.

IGMP Version 3

The switch supports IGMP version 3. The following are considerations for the switch and IGMP version 3:

- An IGMPv3 switch supports Basic IGMPv3 Snooping Support (BISS), which includes support for the snooping features on IGMPv1 and IGMPv2 switches and for IGMPv3 membership report messages. BISS constrains the flooding of multicast traffic when your network includes IGMPv3 hosts. It constrains traffic to approximately the same set of ports as the IGMP snooping feature on IGMPv2 or IGMPv1 hosts.
- The switch supports IGMPv3 snooping based only on the destination multicast IP address. It does not support snooping based on a source IP address or proxy report.
- IGMPv3 join and leave messages are not supported on switches running IGMP filtering or Multicast VLAN registration (MVR).
- An IGMPv3 switch can receive messages from and forward messages to a device running the Source Specific Multicast (SSM) feature.

IGMPv3 Host Signalling

In IGMPv3, hosts signal membership to last hop routers of multicast groups. Hosts can signal group membership with filtering capabilities with respect to sources. A host can either signal that it wants to receive traffic from all sources sending to a group except for some specific sources (called exclude mode), or that it wants to receive traffic only from some specific sources sending to the group (called include mode).

IGMPv3 can operate with both Internet Standard Multicast (ISM) and Source Specific Multicast (SSM). In ISM, both exclude and include mode reports are applicable. In SSM, only include mode reports are accepted by the last-hop router. Exclude mode reports are ignored.

IGMP Snooping

Layer 2 switches can use IGMP snooping to constrain the flooding of multicast traffic by dynamically configuring Layer 2 interfaces so that multicast traffic is forwarded to only those interfaces associated with IP multicast devices. As the name implies, IGMP snooping requires the LAN switch to snoop on the IGMP transmissions between the host and the router and to keep track of multicast groups and member ports. When the switch receives an IGMP report from a host for a particular multicast group, the switch adds the host port number to the forwarding table entry; when it receives an IGMP Leave Group message from a host, it removes the host port from the table entry. It also periodically deletes entries if it does not receive IGMP membership reports from the multicast clients.

Note For more information on IP multicast and IGMP, see RFC 1112 and RFC 2236.

The multicast router (which could be a switch with the IP services feature set on the active switch) sends out periodic general queries to all VLANs. All hosts interested in this multicast traffic send join requests and are added to the forwarding table entry. The switch creates one entry per VLAN in the IGMP snooping IP multicast forwarding table for each group from which it receives an IGMP join request.

The switch supports IP multicast group-based bridging, instead of MAC-addressed based groups. With multicast MAC address-based groups, if an IP address being configured translates (aliases) to a previously configured MAC address or to any reserved multicast MAC addresses (in the range 224.0.0.xxx), the command fails. Because the switch uses IP multicast groups, there are no address aliasing issues.

The IP multicast groups learned through IGMP snooping are dynamic. However, you can statically configure multicast groups by using the **ip igmp snooping vlan** *vlan-id* **static** *ip_address* **interface** *interface-id* global configuration command. If you specify group membership for a multicast group address statically, your setting supersedes any automatic manipulation by IGMP snooping. Multicast group membership lists can consist of both user-defined and IGMP snooping-learned settings.

You can configure an IGMP snooping querier to support IGMP snooping in subnets without multicast interfaces because the multicast traffic does not need to be routed.

If a port spanning-tree, a port group, or a VLAN ID change occurs, the IGMP snooping-learned multicast groups from this port on the VLAN are deleted.

Related Topics

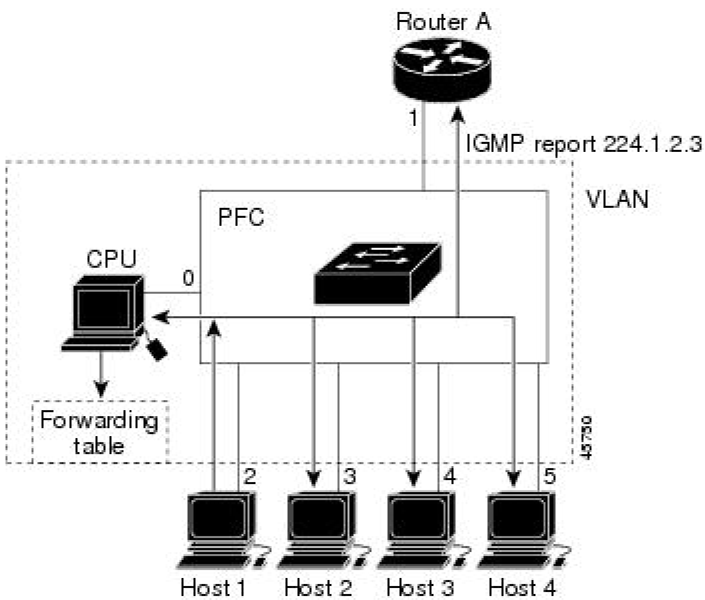
Enabling or Disabling IGMP Snooping on a Switch , on page 811

Examples: Configuring IGMP Snooping, on page 833

Joining a Multicast Group

When a host connected to the switch wants to join an IP multicast group and it is an IGMP version 2 client, it sends an unsolicited IGMP join message, specifying the IP multicast group to join. Alternatively, when the switch receives a general query from the router, it forwards the query to all ports in the VLAN. IGMP version 1 or version 2 hosts wanting to join the multicast group respond by sending a join message to the switch. The switch CPU creates a multicast forwarding-table entry for the group if it is not already present. The CPU also adds the interface where the join message was received to the forwarding-table entry. The host associated with that interface receives multicast traffic for that multicast group.

Figure 25: Initial IGMP Join Message



Router A sends a general query to the switch, which forwards the query to ports 2 through 5, which are all members of the same VLAN. Host 1 wants to join multicast group 224.1.2.3 and multicasts an IGMP membership report (IGMP join message) to the group. The switch CPU uses the information in the IGMP report to set up a forwarding-table entry that includes the port numbers connected to Host 1 and to the router.

Table 94: IGMP Snooping Forwarding Table

Destination Address	Type of Packet	Ports
224.1.2.3	IGMP	1, 2

The switch hardware can distinguish IGMP information packets from other packets for the multicast group. The information in the table tells the switching engine to send frames addressed to the 224.1.2.3 multicast IP address that are not IGMP packets to the router and to the host that has joined the group.

If another host (for example, Host 4) sends an unsolicited IGMP join message for the same group, the CPU receives that message and adds the port number of Host 4 to the forwarding table. Because the forwarding table directs IGMP messages only to the CPU, the message is not flooded to other ports on the switch. Any known multicast traffic is forwarded to the group and not to the CPU.

Figure 26: Second Host Joining a Multicast Group

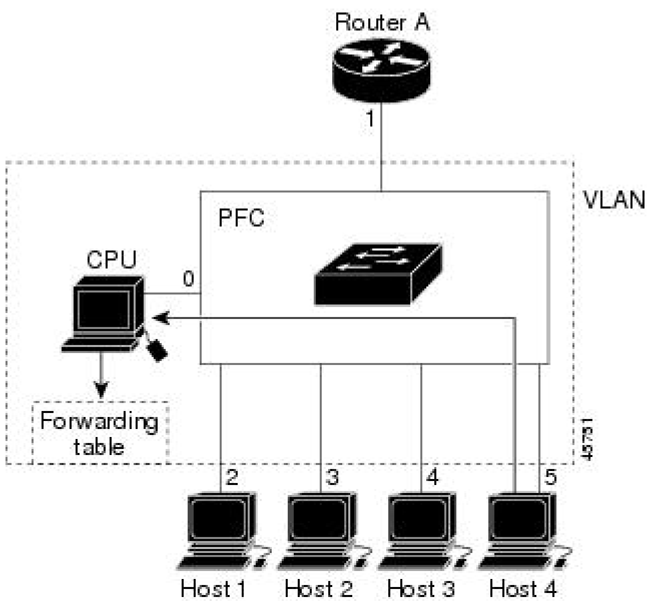


Table 95: Updated IGMP Snooping Forwarding Table

Destination Address	Type of Packet	Ports
224.1.2.3	IGMP	1, 2, 5

Related Topics

Configuring the Switch as a Member of a Group , on page 796

Example: Configuring the Switch as a Member of a Multicast Group, on page 832

Leaving a Multicast Group

The router sends periodic multicast general queries, and the switch forwards these queries through all ports in the VLAN. Interested hosts respond to the queries. If at least one host in the VLAN wants to receive multicast traffic, the router continues forwarding the multicast traffic to the VLAN. The switch forwards multicast group traffic only to those hosts listed in the forwarding table for that IP multicast group maintained by IGMP snooping.

When hosts want to leave a multicast group, they can silently leave, or they can send a leave message. When the switch receives a leave message from a host, it sends a group-specific query to learn if any other devices connected to that interface are interested in traffic for the specific multicast group. The switch then updates the forwarding table for that MAC group so that only those hosts interested in receiving multicast traffic for the group are listed in the forwarding table. If the router receives no reports from a VLAN, it removes the group for the VLAN from its IGMP cache.

Immediate Leave

The switch uses IGMP snooping Immediate Leave to remove from the forwarding table an interface that sends a leave message without the switch sending group-specific queries to the interface. The VLAN interface is pruned from the multicast tree for the multicast group specified in the original leave message. Immediate Leave ensures optimal bandwidth management for all hosts on a switched network, even when multiple multicast groups are simultaneously in use.

Immediate Leave is only supported on IGMP version 2 hosts. IGMP version 2 is the default version for the switch.

Note You should only use the Immediate Leave feature on VLANs where a single host is connected to each port. If Immediate Leave is enabled in VLANs where more than one host is connected to a port, some hosts might inadvertently be dropped.

IGMP Configurable-Leave Timer

You can configure the time that the switch waits after sending a group-specific query to determine if hosts are still interested in a specific multicast group. The IGMP leave response time can be configured from 100 to 5000 milliseconds. The timer can be set either globally or on a per-VLAN basis. The VLAN configuration of the leave time overrides the global configuration.

Related Topics

Configuring the IGMP Leave Timer , on page 818

IGMP Report Suppression

Note IGMP report suppression is supported only when the multicast query has IGMPv1 and IGMPv2 reports. This feature is not supported when the query includes IGMPv3 reports.

The switch uses IGMP report suppression to forward only one IGMP report per multicast router query to multicast devices. When IGMP report suppression is enabled (the default), the switch sends the first IGMP report from all hosts for a group to all the multicast routers. The switch does not send the remaining IGMP reports for the group to the multicast routers. This feature prevents duplicate reports from being sent to the multicast devices.

If the multicast router query includes requests only for IGMPv1 and IGMPv2 reports, the switch forwards only the first IGMPv1 or IGMPv2 report from all hosts for a group to all the multicast routers.

If the multicast router query also includes requests for IGMPv3 reports, the switch forwards all IGMPv1, IGMPv2, and IGMPv3 reports for a group to the multicast devices.

If you disable IGMP report suppression, all IGMP reports are forwarded to the multicast routers.

IGMP Snooping and Switch Stacks

IGMP snooping functions across the switch stack; that is, IGMP control information from one switch is distributed to all switches in the stack. Regardless of the stack member through which IGMP multicast data enters the stack, the data reaches the hosts that have registered for that group.

If a switch in the stack fails or is removed from the stack, only the members of the multicast group that are on that switch will not receive the multicast data. All other members of a multicast group on other switches in the stack continue to receive multicast data streams. However, multicast groups that are common for both Layer 2 and Layer 3 (IP multicast routing) might take longer to converge if the active switch is removed.

IGMP Filtering and Throttling Overview

In some environments, for example, metropolitan or multiple-dwelling unit (MDU) installations, you might want to control the set of multicast groups to which a user on a switch port can belong. You can control the distribution of multicast services, such as IP/TV, based on some type of subscription or service plan. You might also want to limit the number of multicast groups to which a user on a switch port can belong.

With the IGMP filtering feature, you can filter multicast joins on a per-port basis by configuring IP multicast profiles and associating them with individual switch ports. An IGMP profile can contain one or more multicast groups and specifies whether access to the group is permitted or denied. If an IGMP profile denying access to a multicast group is applied to a switch port, the IGMP join report requesting the stream of IP multicast traffic is dropped, and the port is not allowed to receive IP multicast traffic from that group. If the filtering action permits access to the multicast group, the IGMP report from the port is forwarded for normal processing. You can also set the maximum number of IGMP groups that a Layer 2 interface can join.

IGMP filtering controls only group-specific query and membership reports, including join and leave reports. It does not control general IGMP queries. IGMP filtering has no relationship with the function that directs the forwarding of IP multicast traffic. The filtering feature operates in the same manner whether CGMP or MVR is used to forward the multicast traffic.

IGMP filtering applies only to the dynamic learning of IP multicast group addresses, not static configuration.

With the IGMP throttling feature, you can set the maximum number of IGMP groups that a Layer 2 interface can join. If the maximum number of IGMP groups is set, the IGMP snooping forwarding table contains the maximum number of entries, and the interface receives an IGMP join report, you can configure an interface to drop the IGMP report or to replace the randomly selected multicast entry with the received IGMP report.

Note IGMPv3 join and leave messages are not supported on switches running IGMP filtering.

Related Topics

Configuring the IGMP Throttling Action , on page 809

Displaying IGMP Filtering and Throttling Configuration, on page 832

Examples: Configuring Filtering and Throttling, on page 834

Default IGMP Configuration

This table displays the default IGMP configuration for the switch.

Table 96: Default IGMP Configuration

Feature	Default Setting
Multilayer switch as a member of a multicast group	No group memberships are defined.
Access to multicast groups	All groups are allowed on an interface.
IGMP version	Version 2 on all interfaces.
IGMP host-query message interval	60 seconds on all interfaces.
IGMP query timeout	60 seconds on all interfaces.
IGMP maximum query response time	10 seconds on all interfaces.
Multilayer switch as a statically connected member	Disabled.

Default IGMP Snooping Configuration

This table displays the default IGMP snooping configuration for the switch.

Table 97: Default IGMP Snooping Configuration

Feature	Default Setting
IGMP snooping	Enabled globally and per VLAN
Multicast routers	None configured
IGMP snooping Immediate Leave	Disabled
Static groups	None configured
TCN[11] flood query count	2
TCN query solicitation	Disabled
IGMP snooping querier	Disabled
IGMP report suppression	Enabled

11 (1) TCN = Topology Change Notification

Default IGMP Filtering and Throttling Configuration

This table displays the default IGMP filtering and throttling configuration for the switch.

Table 98: Default IGMP Filtering Configuration

Feature	Default Setting
IGMP filters	None applied.
IGMP maximum number of IGMP groups	No maximum set. **Note** When the maximum number of groups is in the forwarding table, the default IGMP throttling action is to deny the IGMP report.
IGMP profiles	None defined.
IGMP profile action	Deny the range addresses.

How to Configure IGMP

Configuring the Switch as a Member of a Group

You can configure the switch as a member of a multicast group and discover multicast reachability in a network. If all the multicast-capable routers and multilayer switches that you administer are members of a multicast group, pinging that group causes all of these devices to respond. The devices respond to ICMP echo-request packets addressed to a group of which they are members. Another example is the multicast trace-route tools provided in the software.

Caution Performing this procedure might impact the CPU performance because the CPU will receive all data traffic for the group address.

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip igmp join-group** *group-address*
4. **end**
5. **show ip igmp interface** [*interface-id*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Specifies the interface on which you want to enable multicast routing, and enters interface configuration mode.
Step 3	**ip igmp join-group** *group-address* **Example:** `Switch(config-if)# ip igmp join-group 225.2.2.2`	Configures the switch to join a multicast group. By default, no group memberships are defined. For *group-address*, specify the multicast IP address in dotted decimal notation.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show ip igmp interface** [*interface-id*] **Example:** `Switch# show ip igmp interface`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Joining a Multicast Group, on page 791

Example: Configuring the Switch as a Member of a Multicast Group, on page 832

IP Multicast Group Addresses, on page 788

Example: Configuring the Switch as a Member of a Multicast Group, on page 832

Controlling Access to IP Multicast Group

The switch sends IGMP host-query messages to find which multicast groups have members on attached local networks. The switch then forwards to these group members all packets addressed to the multicast group. You can place a filter on each interface to restrict the multicast groups that hosts on the subnet serviced by the interface can join.

To limit the number of joins on the interface, configure the port for the filter which associates with the IGMP profile.

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp profile**
3. **permit**
4. **exit**
5. **interface** *interface-id*
6. **ip igmp filter** *filter_number*
7. **end**
8. **show ip igmp interface** [*interface-id*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**ip igmp profile**
	
	Example:
	`Switch(config)# ip igmp profile 10`
	`Switch(config-igmp-profile)# ?`
		Enters an IGMP filter profile number from 1 to 4294967295.
		For additional information about configuring IGMP filter profiles, see Configuring IGMP Profiles , on page 805.

Step 3	**permit** **Example:** `Switch(config-igmp-profile)# permit 229.9.9.0`	Enters an IGMP profile configuration action. The following IGMP profile configuration actions are supported: • **deny**—Matching IP addresses are denied. • **exit**—Exits from the IGMP profile configuration mode. • **no**—Negates a command or set its defaults. • **permit**—Matching addresses are permitted. • **range**—Adds a range to the set.

	Command or Action	Purpose
Step 4	**exit** **Example:** `Switch(config-igmp-profile)# exit`	Returns to global configuration mode.
Step 5	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Specifies the interface to be configured, and enters interface configuration mode.
Step 6	**ip igmp filter** *filter_number* **Example:** `Switch(config-if)# ip igmp filter 10`	Specifies the IGMP filter profile number. For additional information about applying IGMP filter profiles, see Applying IGMP Profiles , on page 807.
Step 7	**end** **Example:** `Switch(config-igmp-profile)# end`	Returns to privileged EXEC mode.
Step 8	**show ip igmp interface** [*interface-id*] **Example:** `Switch# show ip igmp interface`	Verifies your entries.

Modifying the IGMP Host-Query Message Interval

The switch periodically sends IGMP host-query messages to discover which multicast groups are present on attached networks. These messages are sent to the all-hosts multicast group (224.0.0.1) with a time-to-live (TTL) of 1. The switch sends host-query messages to refresh its knowledge of memberships present on the network. If, after some number of queries, the software discovers that no local hosts are members of a multicast group, the software stops forwarding multicast packets to the local network from remote origins for that group and sends a prune message upstream toward the source.

The switch elects a PIM designated router (DR) for the LAN (subnet). The DR is the router or multilayer switch with the highest IP address for IGMPv2. For IGMPv1, the DR is elected according to the multicast routing protocol that runs on the LAN. The designated router is responsible for sending IGMP host-query messages to all hosts on the LAN. In sparse mode, the designated router also sends PIM register and PIM join messages toward the RP router.

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip igmp query-interval** *seconds*
4. **end**
5. **show ip igmp interface** [*interface-id*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Specifies the interface on which you want to enable multicast routing, and enters interface configuration mode.
Step 3	**ip igmp query-interval** *seconds* **Example:** `Switch(config-if)# ip igmp query-interval 75`	Configures the frequency at which the designated router sends IGMP host-query messages. By default, the designated router sends IGMP host-query messages every 60 seconds to keep the IGMP overhead very low on hosts and networks.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show ip igmp interface** [*interface-id*] **Example:** `Switch# show ip igmp interface`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config`	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	`startup-config`	

Changing the IGMP Query Timeout for IGMPv2

If you are using IGMPv2, you can specify the period of time before the switch takes over as the querier for the interface. By default, the switch waits twice the query interval period controlled by the **ip igmp query-interval** interface configuration command. After that time, if the switch has received no queries, it becomes the querier.

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip igmp querier-timeout** *seconds*
4. **end**
5. **show ip igmp interface** [*interface-id*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Specifies the interface on which you want to enable multicast routing, and enters interface configuration mode.
Step 3	**ip igmp querier-timeout** *seconds* **Example:** `Switch(config-if)# ip igmp querier-timeout 120`	Specifies the IGMP query timeout. The default is 60 seconds (twice the query interval). The range is 60 to 300.

	Command or Action	Purpose
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show ip igmp interface** [*interface-id*] **Example:** `Switch# show ip igmp interface`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Changing the Maximum Query Response Time for IGMPv2

If you are using IGMPv2, you can change the maximum query response time advertised in IGMP queries. The maximum query response time enables the switch to quickly detect that there are no more directly connected group members on a LAN. Decreasing the value enables the switch to prune groups faster.

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip igmp query-max-response-time** *seconds*
4. **end**
5. **show ip igmp interface** [*interface-id*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Specifies the interface on which you want to enable multicast routing, and enters interface configuration mode.
Step 3	**ip igmp query-max-response-time** *seconds* **Example:** `Switch(config-if)# ip igmp query-max-response-time 15`	Changes the maximum query response time advertised in IGMP queries. The default is 10 seconds. The range is 1 to 25.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show ip igmp interface** [*interface-id*] **Example:** `Switch# show ip igmp interface`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring the Switch as a Statically Connected Member

At various times, either there is not a group member on a network segment or a host that cannot report its group membership by using IGMP. However, you may want multicast traffic to be sent to that network segment. The following commands are used to pull multicast traffic down to a network segment:

- **ip igmp join-group**—The switch accepts the multicast packets in addition to forwarding them. Accepting the multicast packets prevents the switch from fast switching.
- **ip igmp static-group**—The switch does not accept the packets itself, but only forwards them. This method enables fast switching. The outgoing interface appears in the IGMP cache, but the switch itself is not a member, as evidenced by lack of an L (local) flag in the multicast route entry.

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip igmp static-group** *group-address*
4. **end**
5. **show ip igmp interface** [*interface-id*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Specifies the interface on which you want to enable multicast routing, and enters interface configuration mode.
Step 3	**ip igmp static-group** *group-address* **Example:** `Switch(config-if)# ip igmp static-group 239.100.100.101`	Configures the switch as a statically connected member of a group. By default, this feature is disabled.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	**show ip igmp interface** [*interface-id*] **Example:** `Switch# show ip igmp interface gigabitethernet 1/0/1`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring IGMP Profiles

To configure an IGMP profile, use the **ip igmp profile** global configuration command with a profile number to create an IGMP profile and to enter IGMP profile configuration mode. From this mode, you can specify the parameters of the IGMP profile to be used for filtering IGMP join requests from a port. When you are in IGMP profile configuration mode, you can create the profile by using these commands:

- **deny—**Specifies that matching addresses are denied; this is the default.
- **exit—**Exits from igmp-profile configuration mode.
- **no—**Negates a command or returns to its defaults.
- **permit—**Specifies that matching addresses are permitted.
- **range—**Specifies a range of IP addresses for the profile. You can enter a single IP address or a range with a start and an end address.

The default is for the switch to have no IGMP profiles configured. When a profile is configured, if neither the **permit** nor **deny** keyword is included, the default is to deny access to the range of IP addresses.

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp profile** *profile number*
3. **permit | deny**
4. **range** *ip multicast address*
5. **end**
6. **show ip igmp profile** *profile number*
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp profile** *profile number* **Example:** `Switch(config)# ip igmp profile 3`	Assigns a number to the profile you are configuring, and enters IGMP profile configuration mode. The profile number range is 1 to 4294967295. **Note** To delete a profile, use the **no ip igmp profile** *profile number* global configuration command.
Step 3	**permit | deny** **Example:** `Switch(config-igmp-profile)# permit`	(Optional) Sets the action to permit or deny access to the IP multicast address. If no action is configured, the default for the profile is to deny access.
Step 4	**range** *ip multicast address* **Example:** `Switch(config-igmp-profile)# range 229.9.9.0`	Enters the IP multicast address or range of IP multicast addresses to which access is being controlled. If entering a range, enter the low IP multicast address, a space, and the high IP multicast address. You can use the **range** command multiple times to enter multiple addresses or ranges of addresses. **Note** To delete an IP multicast address or range of IP multicast addresses, use the **no range** *ip multicast address* IGMP profile configuration command.
Step 5	**end** **Example:** `Switch(config-igmp-profile)# end`	Returns to privileged EXEC mode.
Step 6	**show ip igmp profile** *profile number* **Example:** `Switch# show ip igmp profile 3`	Verifies the profile configuration.
Step 7	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Applying IGMP Profiles

To control access as defined in an IGMP profile, use the **ip igmp filter** interface configuration command to apply the profile to the appropriate interfaces. You can apply IGMP profiles only to Layer 2 access ports; you cannot apply IGMP profiles to routed ports or SVIs. You cannot apply profiles to ports that belong to an EtherChannel port group. You can apply a profile to multiple interfaces, but each interface can have only one profile applied to it.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip igmp filter** *profile number*
4. **end**
5. **show running-config interface** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Specifies the physical interface, and enters interface configuration mode. The interface must be a Layer 2 port that does not belong to an EtherChannel port group.
Step 3	**ip igmp filter** *profile number* **Example:** `Switch(config-if)# ip igmp filter 321`	Applies the specified IGMP profile to the interface. The range is 1 to 4294967295. **Note** To remove a profile from an interface, use the **no ip igmp filter** *profile number* interface configuration command.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	**show running-config interface** *interface-id* **Example:** `Switch# show running-config interface gigabitethernet1/0/1`	Verifies the configuration.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Setting the Maximum Number of IGMP Groups

You can set the maximum number of IGMP groups that a Layer 2 interface can join by using the **ip igmp max-groups** interface configuration command. Use the **no** form of this command to set the maximum back to the default, which is no limit.

This restriction can be applied to Layer 2 ports only; you cannot set a maximum number of IGMP groups on routed ports or SVIs. You also can use this command on a logical EtherChannel interface but cannot use it on ports that belong to an EtherChannel port group.

Beginning in privileged EXEC mode, follow these steps to set the maximum number of IGMP groups in the forwarding table:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip igmp max-groups** *number*
4. **end**
5. **show running-config interface** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/2`	Specifies the interface to be configured, and enters interface configuration mode. The interface can be a Layer 2 port that does not belong to an EtherChannel group or a EtherChannel interface.
Step 3	**ip igmp max-groups** *number* **Example:** `Switch(config-if)# ip igmp max-groups 20`	Sets the maximum number of IGMP groups that the interface can join. The range is 0 to 4294967294. The default is to have no maximum set. **Note** The switch supports a maximum number of 4096 Layer 2 IGMP groups and 2048 Layer 3 IGMP groups.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config interface** *interface-id* **Example:** `Switch# show running-config interface gigabitethernet1/0/1`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring the IGMP Throttling Action

After you set the maximum number of IGMP groups that a Layer 2 interface can join, you can configure an interface to replace the existing group with the new group for which the IGMP report was received by using the **ip igmp max-groups action replace** interface configuration command. Use the **no** form of this command to return to the default, which is to drop the IGMP join report.

Follow these guidelines when configuring the IGMP throttling action:

- This restriction can be applied only to Layer 2 ports. You can use this command on a logical EtherChannel interface but cannot use it on ports that belong to an EtherChannel port group.
- When the maximum group limitation is set to the default (no maximum), entering the **ip igmp max-groups action** {**deny** | **replace**} command has no effect.

- If you configure the throttling action and set the maximum group limitation after an interface has added multicast entries to the forwarding table, the forwarding-table entries are either aged out or removed, depending on the throttling action.
  - If you configure the throttling action as **deny**, the entries that were previously in the forwarding table are not removed but are aged out. After these entries are aged out and the maximum number of entries is in the forwarding table, the switch drops the next IGMP report received on the interface.
  - If you configure the throttling action as **replace**, the entries that were previously in the forwarding table are removed. When the maximum number of entries is in the forwarding table, the switch replaces a randomly selected entry with the received IGMP report.

    To prevent the switch from removing the forwarding-table entries, you can configure the IGMP throttling action before an interface adds entries to the forwarding table.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip igmp max-groups action** {**deny** | **replace**}
4. **end**
5. **show running-config interface** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Specifies the physical interface to be configured, and enters interface configuration mode. The interface can be a Layer 2 port that does not belong to an EtherChannel group or an EtherChannel interface. The interface cannot be a trunk port.
Step 3	**ip igmp max-groups action** {**deny** | **replace**} **Example:** `Switch(config-if)# ip igmp max-groups action replace`	When an interface receives an IGMP report and the maximum number of entries is in the forwarding table, specifies the action that the interface takes: • **deny**—Drops the report. • **replace**—Replaces the existing group with the new group for which the IGMP report was received.

	Command or Action	Purpose
		Note To return to the default action of dropping the report, use the **no ip igmp max-groups action** interface configuration command.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config interface** *interface-id* **Example:** `Switch# show running-config interface gigabitethernet1/0/1`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

IGMP Filtering and Throttling Overview, on page 794

Displaying IGMP Filtering and Throttling Configuration, on page 832

Examples: Configuring Filtering and Throttling, on page 834

How to Configure IGMP Snooping

Enabling or Disabling IGMP Snooping on a Switch

By default, IGMP snooping is globally enabled on the switch. When globally enabled or disabled, it is also enabled or disabled in all existing VLAN interfaces. IGMP snooping is by default enabled on all VLANs, but can be enabled and disabled on a per-VLAN basis.

Global IGMP snooping overrides the VLAN IGMP snooping. If global snooping is disabled, you cannot enable VLAN snooping. If global snooping is enabled, you can enable or disable VLAN snooping.

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping**
3. **end**
4. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp snooping** **Example:** `Switch(config)# ip igmp snooping`	Globally enables IGMP snooping in all existing VLAN interfaces. **Note** To globally disable IGMP snooping on all VLAN interfaces, use the **no ip igmp snooping** global configuration command.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

IGMP Snooping, on page 790

Examples: Configuring IGMP Snooping, on page 833

Enabling or Disabling IGMP Snooping on a VLAN Interface

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping vlan** *vlan-id*
3. **end**
4. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp snooping vlan** *vlan-id* **Example:** `Switch(config)# ip igmp snooping vlan 7`	Enables IGMP snooping on the VLAN interface. The VLAN ID range is 1 to 1001 and 1006 to 4094. IGMP snooping must be globally enabled before you can enable VLAN snooping. **Note** To disable IGMP snooping on a VLAN interface, use the **no ip igmp snooping vlan** *vlan-id* global configuration command for the specified VLAN number.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Setting the Snooping Method

Multicast-capable router ports are added to the forwarding table for every Layer 2 multicast entry. The switch learns of the ports through one of these methods:

- Snooping on IGMP queries

- Statically connecting to a multicast router port using the **ip igmp snooping mrouter** global configuration command

Beginning in privileged EXEC mode, follow these steps to alter the method in which a VLAN interface accesses a multicast router:

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping vlan** *vlan-id* **mrouter interface** {**GigabitEthernet** | **Port-Channel** | **TenGigabitEthernet**}
3. **end**
4. **show ip igmp snooping**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**ip igmp snooping vlan** *vlan-id* **mrouter interface** {**GigabitEthernet** | **Port-Channel** | **TenGigabitEthernet**}
	
	Example:
	
	`Switch(config)# ip igmp snooping`
	`vlan 1 mrouter interface GigabitEthernet1/0/3`
		Enables IGMP snooping on a VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094.

Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show ip igmp snooping** **Example:** `Switch# show ip igmp snooping`	Verifies the configuration.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring a Multicast Router Port

To add a multicast router port (add a static connection to a multicast router), use the **ip igmp snooping vlan mrouter** global configuration command on the switch.

Note Static connections to multicast routers are supported only on switch ports.

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping vlan** *vlan-id* **mrouter interface** *interface-id*
3. **end**
4. **show ip igmp snooping mrouter** [**vlan** *vlan-id*]
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp snooping vlan** *vlan-id* **mrouter interface** *interface-id* **Example:** `Switch(config)# ip igmp snooping vlan 5 mrouter interface gigabitethernet1/0/1`	Specifies the multicast router VLAN ID and the interface to the multicast router. • The VLAN ID range is 1 to 1001 and 1006 to 4094. • The interface can be a physical interface or a port channel. The port-channel range is 1 to 128. **Note** To remove a multicast router port from the VLAN, use the **no ip igmp snooping vlan** *vlan-id* **mrouter interface** *interface-id* global configuration command.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 4	**show ip igmp snooping mrouter** [**vlan** *vlan-id*] **Example:** `Switch# show ip igmp snooping mrouter vlan 5`	Verifies that IGMP snooping is enabled on the VLAN interface.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring a Host Statically to Join a Group

Hosts or Layer 2 ports normally join multicast groups dynamically, but you can also statically configure a host on an interface.

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping vlan** *vlan-id* **static** *ip_address* **interface** *interface-id*
3. **end**
4. **show ip igmp snooping groups**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp snooping vlan** *vlan-id* **static** *ip_address* **interface** *interface-id* **Example:** `Switch(config)# ip igmp snooping vlan 105 static 230.0.0.1 interface gigabitethernet1/0/1`	Statically configures a Layer 2 port as a member of a multicast group: • *vlan-id* is the multicast group VLAN ID. The range is 1 to 1001 and 1006 to 4094. • *ip-address* is the group IP address. • *interface-id* is the member port. It can be a physical interface or a port channel (1 to 128).

	Command or Action	Purpose
		Note To remove the Layer 2 port from the multicast group, use the **no ip igmp snooping vlan** *vlan-id* **static** *mac-address* **interface** *interface-id* global configuration command.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show ip igmp snooping groups** **Example:** `Switch# show ip igmp snooping groups`	Verifies the member port and the IP address.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Enabling IGMP Immediate Leave

When you enable IGMP Immediate Leave, the switch immediately removes a port when it detects an IGMP Version 2 leave message on that port. You should only use the Immediate-Leave feature when there is a single receiver present on every port in the VLAN.

Note Immediate Leave is supported only on IGMP Version 2 hosts. IGMP Version 2 is the default version for the switch.

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping vlan** *vlan-id* **immediate-leave**
3. **end**
4. **show ip igmp snooping vlan** *vlan-id*
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp snooping vlan** *vlan-id* **immediate-leave** **Example:** `Switch(config)# ip igmp snooping vlan 21 immediate-leave`	Enables IGMP Immediate Leave on the VLAN interface. **Note** To disable IGMP Immediate Leave on a VLAN, use the **no ip igmp snooping vlan** *vlan-id* **immediate-leave** global configuration command.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show ip igmp snooping vlan** *vlan-id* **Example:** `Switch# show ip igmp snooping vlan 21`	Verifies that Immediate Leave is enabled on the VLAN interface.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring the IGMP Leave Timer

Follow these guidelines when configuring the IGMP leave timer:

- You can configure the leave time globally or on a per-VLAN basis.
- Configuring the leave time on a VLAN overrides the global setting.
- The default leave time is 1000 milliseconds.
- The IGMP configurable leave time is only supported on hosts running IGMP Version 2. IGMP version 2 is the default version for the switch.
- The actual leave latency in the network is usually the configured leave time. However, the leave time might vary around the configured time, depending on real-time CPU load conditions, network delays and the amount of traffic sent through the interface.

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping last-member-query-interval** *time*
3. **ip igmp snooping vlan** *vlan-id* **last-member-query-interval** *time*
4. **end**
5. **show ip igmp snooping**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp snooping last-member-query-interval** *time* **Example:** `Switch(config)# ip igmp snooping last-member-query-interval 1000`	Configures the IGMP leave timer globally. The range is 100 to 32768 milliseconds. The default is 1000 seconds. **Note** To globally reset the IGMP leave timer to the default setting, use the **no ip igmp snooping last-member-query-interval** global configuration command.
Step 3	**ip igmp snooping vlan** *vlan-id* **last-member-query-interval** *time* **Example:** `Switch(config)# ip igmp snooping vlan 210 last-member-query-interval 1000`	(Optional) Configures the IGMP leave time on the VLAN interface. The range is 100 to 32768 milliseconds. **Note** Configuring the leave time on a VLAN overrides the globally configured timer. **Note** To remove the configured IGMP leave-time setting from the specified VLAN, use the **no ip igmp snooping vlan** *vlan-id* **last-member-query-interval** global configuration command.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show ip igmp snooping** **Example:** `Switch# show ip igmp snooping`	(Optional) Displays the configured IGMP leave time.

	Command or Action	Purpose
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

IGMP Configurable-Leave Timer, on page 793

Configuring the IGMP Robustness-Variable

Use the following procedure to configure the IGMP robustness variable on the switch.

The robustness variable is the integer used by IGMP snooping during calculations for IGMP messages. The robustness variable provides fine tuning to allow for expected packet loss.

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping robustness-variable** *count*
3. **ip igmp snooping vlan** *vlan-id* **robustness-variable** *count*
4. **end**
5. **show ip igmp snooping**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp snooping robustness-variable** *count* **Example:** `Switch(config)# ip igmp snooping robustness-variable 3`	Configures the IGMP robustness variable. The range is 1 to 3 times. The recommended value for the robustness variable is 2. Use this command to change the value of the robustness variable for IGMP snooping from the default (2) to a specified value.

	Command or Action	Purpose
Step 3	**ip igmp snooping vlan** *vlan-id* **robustness-variable** *count* **Example:** `Switch(config)#`**`ip igmp snooping vlan 100 robustness-variable 3`**	(Optional) Configures the IGMP robustness variable on the VLAN interface. The range is 1 to 3 times. The recommended value for the robustness variable is 2. **Note** Configuring the robustness variable count on a VLAN overrides the globally configured value.
Step 4	**end** **Example:** `Switch(config)#` **`end`**	Returns to privileged EXEC mode.
Step 5	**show ip igmp snooping** **Example:** `Switch#` **`show ip igmp snooping`**	(Optional) Displays the configured IGMP robustness variable count.
Step 6	**copy running-config startup-config** **Example:** `Switch#` **`copy running-config startup-config`**	(Optional) Saves your entries in the configuration file.

Configuring the IGMP Last Member Query Count

To configure the number of times the switch sends IGMP group-specific or group-source-specific (with IGMP version 3) query messages in response to receiving a group-specific or group-source-specific leave message, use this command.

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping last-member-query-count** *count*
3. **ip igmp snooping vlan** *vlan-id* **last-member-query-count** *count*
4. **end**
5. **show ip igmp snooping**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp snooping last-member-query-count** *count* **Example:** `Switch(config)# ip igmp snooping last-member-query-count 3`	Configures the IGMP last member query count. The range is 1 to 7 messages. The default is 2 messages.
Step 3	**ip igmp snooping vlan** *vlan-id* **last-member-query-count** *count* **Example:** `Switch(config)#ip igmp snooping vlan 100 last-member-query-count 3`	(Optional) Configures the IGMP last member query count on the VLAN interface. The range is 1 to 7 messages. **Note** Configuring the last member query count on a VLAN overrides the globally configured timer.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show ip igmp snooping** **Example:** `Switch# show ip igmp snooping`	(Optional) Displays the configured IGMP last member query count.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring TCN-Related Commands

Controlling the Multicast Flooding Time After a TCN Event

You can control the time that multicast traffic is flooded after a topology change notification (TCN) event by using the **ip igmp snooping tcn flood query count** global configuration command. This command configures the number of general queries for which multicast data traffic is flooded after a TCN event. Some examples of TCN events are when the client changed its location and the receiver is on same port that was blocked but is now forwarding, and when a port went down without sending a leave message.

If you set the TCN flood query count to 1 by using the **ip igmp snooping tcn flood query count** command, the flooding stops after receiving 1 general query. If you set the count to 7, the flooding continues until 7 general queries are received. Groups are relearned based on the general queries received during the TCN event.

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping tcn flood query count** *count*
3. **end**
4. **show ip igmp snooping**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp snooping tcn flood query count** *count* **Example:** `Switch(config)# ip igmp snooping tcn flood query count 3`	Specifies the number of IGMP general queries for which the multicast traffic is flooded. The range is 1 to 10. By default, the flooding query count is 2. **Note** To return to the default flooding query count, use the **no ip igmp snooping tcn flood query count** global configuration command.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show ip igmp snooping** **Example:** `Switch# show ip igmp snooping`	Verifies the TCN settings.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Recovering from Flood Mode

When a topology change occurs, the spanning-tree root sends a special IGMP leave message (also known as global leave) with the group multicast address 0.0.0.0. However, when you enable the **ip igmp snooping tcn query solicit** global configuration command, the switch sends the global leave message whether or not it is the spanning-tree root. When the router receives this special leave, it immediately sends general queries, which expedite the process of recovering from the flood mode during the TCN event. Leaves are always sent if the switch is the spanning-tree root regardless of this configuration command. By default, query solicitation is disabled.

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping tcn query solicit**
3. **end**
4. **show ip igmp snooping**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp snooping tcn query solicit** **Example:** `Switch(config)# ip igmp snooping tcn query solicit`	Sends an IGMP leave message (global leave) to speed the process of recovering from the flood mode caused during a TCN event. By default, query solicitation is disabled. **Note** To return to the default query solicitation, use the **no ip igmp snooping tcn query solicit** global configuration command.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show ip igmp snooping** **Example:** `Switch# show ip igmp snooping`	Verifies the TCN settings.

	Command or Action	Purpose
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Disabling Multicast Flooding During a TCN Event

When the switch receives a TCN, multicast traffic is flooded to all the ports until 2 general queries are received. If the switch has many ports with attached hosts that are subscribed to different multicast groups, this flooding might exceed the capacity of the link and cause packet loss. You can use the **ip igmp snooping tcn flood** interface configuration command to control this operation function.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **no ip igmp snooping tcn flood**
4. **end**
5. **show ip igmp snooping**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Specifies the interface to be configured, and enters interface configuration mode.
Step 3	**no ip igmp snooping tcn flood** **Example:** `Switch(config-if)# no ip igmp snooping tcn`	Disables the flooding of multicast traffic during a spanning-tree TCN event. By default, multicast flooding is enabled on an interface. **Note** To re-enable multicast flooding on an interface, use the **ip igmp snooping tcn flood** interface configuration command.

	Command or Action	Purpose
	flood	
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show ip igmp snooping** **Example:** `Switch# show ip igmp snooping`	Verifies the TCN settings.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring the IGMP Snooping Querier

Follow these guidelines when configuring the IGMP snooping querier:

- Configure the VLAN in global configuration mode.
- Configure an IP address on the VLAN interface. When enabled, the IGMP snooping querier uses the IP address as the query source address.
- If there is no IP address configured on the VLAN interface, the IGMP snooping querier tries to use the configured global IP address for the IGMP querier. If there is no global IP address specified, the IGMP querier tries to use the VLAN switch virtual interface (SVI) IP address (if one exists). If there is no SVI IP address, the switch uses the first available IP address configured on the switch. The first IP address available appears in the output of the **show ip interface** privileged EXEC command. The IGMP snooping querier does not generate an IGMP general query if it cannot find an available IP address on the switch.
- The IGMP snooping querier supports IGMP Versions 1 and 2.
- When administratively enabled, the IGMP snooping querier moves to the nonquerier state if it detects the presence of a multicast router in the network.
- When it is administratively enabled, the IGMP snooping querier moves to the operationally disabled state under these conditions:
    - IGMP snooping is disabled in the VLAN.
    - PIM is enabled on the SVI of the corresponding VLAN.

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping querier**
3. **ip igmp snooping querier address** *ip_address*
4. **ip igmp snooping querier query-interval** *interval-count*
5. **ip igmp snooping querier tcn query** [**count** *count* | **interval** *interval*]
6. **ip igmp snooping querier timer expiry** *timeout*
7. **ip igmp snooping querier version** *version*
8. **end**
9. **show ip igmp snooping vlan** *vlan-id*
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp snooping querier** **Example:** `Switch(config)# ip igmp snooping querier`	Enables the IGMP snooping querier.
Step 3	**ip igmp snooping querier address** *ip_address* **Example:** `Switch(config)# ip igmp snooping querier address 172.16.24.1`	(Optional) Specifies an IP address for the IGMP snooping querier. If you do not specify an IP address, the querier tries to use the global IP address configured for the IGMP querier. **Note** The IGMP snooping querier does not generate an IGMP general query if it cannot find an IP address on the switch.
Step 4	**ip igmp snooping querier query-interval** *interval-count* **Example:** `Switch(config)# ip igmp snooping querier query-interval 30`	(Optional) Sets the interval between IGMP queriers. The range is 1 to 18000 seconds.

	Command or Action	Purpose
Step 5	**ip igmp snooping querier tcn query** [**count** *count* | **interval** *interval*] **Example:** `Switch(config)# ip igmp snooping querier tcn query interval 20`	(Optional) Sets the time between Topology Change Notification (TCN) queries. The count range is 1 to 10. The interval range is 1 to 255 seconds.
Step 6	**ip igmp snooping querier timer expiry** *timeout* **Example:** `Switch(config)# ip igmp snooping querier timer expiry 180`	(Optional) Sets the length of time until the IGMP querier expires. The range is 60 to 300 seconds.
Step 7	**ip igmp snooping querier version** *version* **Example:** `Switch(config)# ip igmp snooping querier version 2`	(Optional) Selects the IGMP version number that the querier feature uses. Select 1 or 2.
Step 8	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 9	**show ip igmp snooping vlan** *vlan-id* **Example:** `Switch# show ip igmp snooping vlan 30`	(Optional) Verifies that the IGMP snooping querier is enabled on the VLAN interface. The VLAN ID range is 1 to 1001 and 1006 to 4094.
Step 10	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Disabling IGMP Report Suppression

Note IGMP report suppression is supported only when the multicast query has IGMPv1 and IGMPv2 reports. This feature is not supported when the query includes IGMPv3 reports.

IGMP report suppression is enabled by default. When it is enabled, the switch forwards only one IGMP report per multicast router query. When report suppression is disabled, all IGMP reports are forwarded to the multicast routers.

SUMMARY STEPS

1. **configure terminal**
2. **no ip igmp snooping report-suppression**
3. **end**
4. **show ip igmp snooping**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**no ip igmp snooping report-suppression** **Example:** `Switch(config)# no ip igmp snooping report-suppression`	Disables IGMP report suppression. **Note** To re-enable IGMP report suppression, use the **ip igmp snooping report-suppression** global configuration command.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show ip igmp snooping** **Example:** `Switch# show ip igmp snooping`	Verifies that IGMP report suppression is disabled.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring IGMP

You can display specific statistics, such as the contents of IP routing tables, caches, and databases.

Note This release does not support per-route statistics.

You can display information to learn resource usage and solve network problems. You can also display information about node reachability and discover the routing path that packets of your device are taking through the network.

You can use any of the privileged EXEC commands in the following table to display various routing statistics.

Table 99: Commands for Displaying System and Network Statistics

Command	Purpose
ping [*group-name* | *group-address*]	Sends an ICMP Echo Request to a multicast group address.
show ip igmp filter	Displays IGMP filter information.
show ip igmp groups [*type-number* | *detail*]	Displays the multicast groups that are directly connected to the switch and that were learned through IGMP.
show ip igmp interface [*type number*]	Displays multicast-related information about an interface.
show ip igmp membership [*name/group address* | **all** | **tracked**]	Displays IGMP membership information for forwarding.
show ip igmp profile [*profile_number*]	Displays IGMP profile information.
show ip igmp ssm-mapping [*hostname/IP address*]	Displays IGMP SSM mapping information.
show ip igmp static-group {**class-map** [**interface** [*type*]]	Displays static group information.
show ip igmp vrf	Displays the selected VPN routing/forwarding instance by name.

Displaying IGMP Snooping Information

You can display IGMP snooping information for dynamically learned and statically configured router ports and VLAN interfaces. You can also display MAC address multicast entries for a VLAN configured for IGMP snooping.

Table 100: Commands for Displaying IGMP Snooping Information

Command	Purpose
show ip igmp snooping detail	Displays the operational state information.
show ip igmp snooping groups [**count** | [**vlan** *vlan-id* [*A.B.C.D* | **count**]]	Displays multicast table information for the switch or about a specific parameter: • **count**—Displays the total number of groups. • **vlan**—Displays group information by VLAN ID.
show ip igmp snooping igmpv2-tracking	Displays the IGMP snooping tracking. **Note** This command displays group and IP address entries only for wireless multicast IGMP joins and not for wired IGMP joins. Wireless IP multicast must be enabled for this command to display.
show ip igmp snooping mrouter [**vlan** *vlan-id*]	Displays information on dynamically learned and manually configured multicast router interfaces. **Note** When you enable IGMP snooping, the switch automatically learns the interface to which a multicast router is connected. These are dynamically learned interfaces. (Optional) Enter **vlan** *vlan-id* to display information for a single VLAN.
show ip igmp snooping querier [**detail** | **vlan** *vlan-id*]	Displays information about the IP address and receiving port for the most-recently received IGMP query messages in the VLAN. (Optional) Enter **detail** to display the detailed IGMP querier information in a VLAN. (Optional) Enter **vlan** *vlan-id* to display information for a single VLAN.
show ip igmp snooping [**vlan** *vlan-id* [**detail**]]	Displays the snooping configuration information for all VLANs on the switch or for a specified VLAN. (Optional) Enter **vlan** *vlan-id* to display information for a single VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094.
show ip igmp snooping wireless mgid	Displays wireless-related events.

Displaying IGMP Filtering and Throttling Configuration

You can display IGMP profile characteristics, and you can display the IGMP profile and maximum group configuration for all interfaces on the switch or for a specified interface. You can also display the IGMP throttling configuration for all interfaces on the switch or for a specified interface.

Table 101: Commands for Displaying IGMP Filtering and Throttling Configuration

Command	Purpose
show ip igmp profile [*profile number*]	Displays the specified IGMP profile or all the IGMP profiles defined on the switch.
show running-config [**interface** *interface-id*]	Displays the configuration of the specified interface or the configuration of all interfaces on the switch, including (if configured) the maximum number of IGMP groups to which an interface can belong and the IGMP profile applied to the interface.

Related Topics

Configuring the IGMP Throttling Action , on page 809

IGMP Filtering and Throttling Overview, on page 794

Configuration Examples for IGMP

Example: Configuring the Switch as a Member of a Multicast Group

This example shows how to enable the switch to join multicast group 255.2.2.2:

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip igmp join-group 255.2.2.2
Switch(config-if)#
```

Related Topics

Configuring the Switch as a Member of a Group , on page 796

Joining a Multicast Group, on page 791

Configuring the Switch as a Member of a Group , on page 796

IP Multicast Group Addresses, on page 788

Example: Controlling Access to Multicast Groups

To limit the number of joins on the interface, configure the port for filter which associates with the IGMP profile.

```
Switch# configure terminal
Switch(config)# ip igmp profile 10
Switch(config-igmp-profile)# ?

IGMP profile configuration commands:
deny matching addresses are denied
exit Exit from igmp profile configuration mode
no Negate a command or set its defaults
permit matching addresses are permitted
range add a range to the set

Switch(config-igmp-profile)# range 172.16.5.1
Switch(config-igmp-profile)# exit
Switch(config)#
Switch(config)# interface gigabitEthernet 2/0/10
Switch(config-if)# ip igmp filter 10
```

Examples: Configuring IGMP Snooping

This example shows how to enable a static connection to a multicast router:

```
Switch# configure terminal
Switch(config)# ip igmp snooping vlan 200 mrouter interface gigabitethernet1/0/2
Switch(config)# end
```

This example shows how to statically configure a host on a port:

```
Switch# configure terminal
Switch(config)# ip igmp snooping vlan 105 static 224.2.4.12 interface gigabitethernet1/0/1
Switch(config)# end
```

This example shows how to enable IGMP Immediate Leave on VLAN 130:

```
Switch# configure terminal
Switch(config)# ip igmp snooping vlan 130 immediate-leave
Switch(config)# end
```

This example shows how to set the IGMP snooping querier source address to 10.0.0.64:

```
Switch# configure terminal
Switch(config)# ip igmp snooping querier 10.0.0.64
Switch(config)# end
```

This example shows how to set the IGMP snooping querier maximum response time to 25 seconds:

```
Switch# configure terminal
Switch(config)# ip igmp snooping querier query-interval 25
Switch(config)# end
```

This example shows how to set the IGMP snooping querier timeout to 60 seconds:

```
Switch# configure terminal
Switch(config)# ip igmp snooping querier timer expiry 60
```

```
Switch(config)# end
```

This example shows how to set the IGMP snooping querier feature to Version 2:

```
Switch# configure terminal
Switch(config)# no ip igmp snooping querier version 2
Switch(config)# end
```

Related Topics

Enabling or Disabling IGMP Snooping on a Switch , on page 811

IGMP Snooping, on page 790

Examples: Configuring Filtering and Throttling

This example shows how to create IGMP profile 4 allowing access to the single IP multicast address and how to verify the configuration. If the action was to deny (the default), it would not appear in the **show ip igmp profile** output display.

```
Switch(config)# ip igmp profile 4
Switch(config-igmp-profile)# permit
Switch(config-igmp-profile)# range 229.9.9.0
Switch(config-igmp-profile)# end
Switch# show ip igmp profile 4
IGMP Profile 4
    permit
    range 229.9.9.0 229.9.9.0
```

This example shows how to apply IGMP profile 4 to a port:

```
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# ip igmp filter 4
Switch(config-if)# end
```

This example shows how to limit to 25 the number of IGMP groups that a port can join:

```
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# ip igmp max-groups 25
Switch(config-if)# end
```

Related Topics

Configuring the IGMP Throttling Action , on page 809

IGMP Filtering and Throttling Overview, on page 794

Example: Interface Configuration as a Routed Port

This example shows how to configure an interface on the switch as a routed port. This configuration is required on the interface for several IP multicast routing configuration procedures that require running the **no switchport** command.

```
Switch configure terminal
Switch(config)#  interface GigabitEthernet1/0/9
Switch(config-if)# description interface to be use as routed port
Switch(config-if)# no switchport
Switch(config-if)# ip address 20.20.20.1 255.255.255.0
```

```
Switch(config-if)# ip pim sparse-dense-mode
Switch(config-if)# ip igmp join-group 224.1.2.3 source 15.15.15.2
Switch(config-if)# end
Switch# configure terminal
Switch# show run interface gigabitEthernet 1/0/9

Current configuration : 166 bytes
!
interface GigabitEthernet1/0/9
 no switchport
 ip address 20.20.20.1 255.255.255.0
 ip pim sparse-dense-mode
 ip igmp static-group 224.1.2.3 source 15.15.15.2
end
```

Example: Interface Configuration as an SVI

This example shows how to configure an interface on the switch as an SVI. This configuration is required on the interface for several IP multicast routing configuration procedures that require running the **no switchport** command.

```
Switch(config)# interface vlan 150
Switch(config-if)# ip address 20.20.20.1 255.255.255.0
Switch(config-if)# ip pim sparse-dense-mode
Switch(config-if)# ip igmp join-group 224.1.2.3 source 15.15.15.2
Switch(config-if)# end
Switch# configure terminal
Switch(config)#  ip igmp snooping vlan 20 static 224.1.2.3
interface gigabitEthernet 1/0/9
Switch# show run interface vlan 150

Current configuration : 137 bytes
!
interface Vlan150
 ip address 20.20.20.1 255.255.255.0
 ip pim sparse-dense-mode
 ip igmp static-group 224.1.2.3 source 15.15.15.2
end
```

Where to Go Next for IGMP

You can configure the following:

- Wireless Multicast
- PIM
- SSM
- IP Multicast Routing
- Service Discovery Gateway

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	*IP Multicast Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) IP Multicast Command Reference, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*
Platform-independent configuration information	• *IP Multicast: PIM Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)* • *IP Multicast: IGMP Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)* • *IP Multicast: Multicast Optimization Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*

Standards and RFCs

Standard/RFC	Title
RFC 1112	*Host Extensions for IP Multicasting*
RFC 2236	*Internet Group Management Protocol, Version 2*

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for IGMP

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 39

Configuring Wireless Multicast

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for Configuring Wireless Multicast

- The IP multicast routing must be enabled and the PIM version and PIM mode must be configured. The default routes should be available in the device. After performing these tasks, the device can then forward multicast packets and can populate its multicast routing table.
- To participate in IP multicasting, the multicast hosts, routers, and multilayer switches must have IGMP operating.
- When enabling multicast mode on the switch, a CAPWAP multicast group address should also be configured. Access points listen to the CAPWAP multicast group using IGMP.

Restrictions for Configuring Wireless Multicast

The following are the restrictions for configuring IP multicast routing:

- Access points in monitor mode, sniffer mode, or rogue detector mode do not join the CAPWAP multicast group address.
- The CAPWAP multicast group configured on the switch should be different for different switches.
- Multicast routing should not be enabled for the management interface.

Information About Wireless Multicast

If the network supports packet multicasting, the multicast method that the switch uses can be configured. The switch performs multicasting in two modes:

- Unicast mode—The switch unicasts every multicast packet to every access point associated to the switch. This mode is inefficient but might be required on networks that do not support multicasting.
- Multicast mode—The switch sends multicast packets to a CAPWAP multicast group. This method reduces overhead on the switch processor and shifts the work of packet replication to the network, which is much more efficient than the unicast method.

When the multicast mode is enabled and the switch receives a multicast packet from the wired LAN, the switch encapsulates the packet using CAPWAP and forwards the packet to the CAPWAP multicast group address. The switch always uses the management VLAN for sending multicast packets. Access points in the multicast group receive the packet and forward it to all the BSSIDs mapped to the VLAN on which clients receive multicast traffic.

The switch supports all the capabilities of v1 including Multicast Listener Discovery (MLD) v1 snooping but the v2 and v3 capabilities are limited. This feature keeps track of and delivers IPv6 multicast flows to the clients that request them. To support IPv6 multicast, global multicast mode should be enabled.

Internet Group Management Protocol (IGMP) snooping is introduced to better direct multicast packets. When this feature is enabled, the switch snooping gathers IGMP reports from the clients, processes them, creates unique multicast group IDs (MGIDs) based on the Layer 3 multicast address and the VLAN number, and sends the IGMP reports to the IGMP querier. The switch then updates the access point MGID table on the access point with the client MAC address. When the switch receives multicast traffic for a particular multicast group, it forwards it to all the access points, but only those access points that have active clients listening or subscribed to that multicast group send multicast traffic on that particular WLAN. IP packets are forwarded with an MGID that is unique for an ingress VLAN and the destination multicast group. Layer 2 multicast packets are forwarded with an MGID that is unique for the ingress VLAN.

MGID is a 14-bit value filled in the 16-bit reserved field of wireless information in CAPWAP header. The remaining 2 bits should be set to zero.

Related Topics

Configuring Wireless Multicast-MCMC Mode , on page 841
Configuring Wireless Multicast-MCUC Mode , on page 842
Configuring Wireless Multicast-MCUC Mode , on page 842

Information About Multicast Optimization

Multicast used to be based on the group of the multicast addresses and the VLAN as one entity, MGID. With the VLAN group, duplicate packets might increase. Using the VLAN group feature, every client listens to the multicast stream on a different VLAN. As a result, the switch creates different MGIDs for each multicast address and VLAN. Therefore, in a worst case situation, the upstream router sends one copy for each VLAN, which results in as many copies as the number of VLANs in the group. Because the WLAN remains the same for all clients, multiple copies of the multicast packet are sent over the wireless network. To suppress the duplication of a multicast stream on the wireless medium between the switch and the access points, the multicast optimization feature can be used.

Multicast optimization enables you to create a multicast VLAN that can be used for multicast traffic. One of the VLANs in the switch can be configured as a multicast VLAN where multicast groups are registered. The clients are allowed to listen to a multicast stream on the multicast VLAN. The MGID is generated using the mulicast VLAN and multicast IP addresses. If multiple clients on different VLANs of the same WLAN are listening to a single multicast IP address, a single MGID is generated. The switch makes sure that all multicast streams from the clients on this VLAN group always go out on the multicast VLAN to ensure that the upstream router has one entry for all the VLANs of the VLAN group. Only one multicast stream hits the VLAN group even if the clients are on different VLANs. Therefore, the multicast packets that are sent out over the network is just one stream.

Related Topics

Configuring IP Multicast VLAN for WLAN , on page 848
Configuring IP Multicast VLAN for WLAN , on page 848

How to Configure Wireless Multicast

Configuring Wireless Multicast-MCMC Mode

SUMMARY STEPS

1. **configure terminal**
2. **wireless multicast**
3. **ap capwap multicast ipaddr**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.

	Command or Action	Purpose

Step 2	**wireless multicast**
	
	Example:
	`Switch(config)# wireless multicast`
	
	`Switch(config)# no wireless multicast`
		Enables the multicast traffic for wireless clients. The default value is disable. Add **no** in the command to disable the multicast traffic for wireless clients.

Step 3	**ap capwap multicast ipaddr**
	
	Example:
	`Switch(config)# ap capwap multicast 231.1.1.1`
	
	`Switch(config)# no ap capwap multicast 231.1.1.1`
		Enables the forwarding mode in multicast. Add **no** in the command to disable the multicast mode.

Step 4	**end** **Example:** `Switch(config)# end`	Exits the configuration mode. Alternatively, press **Ctrl-Z** to exit the configuration mode.

Related Topics

Information About Wireless Multicast, on page 840

Configuring Wireless Multicast-MCUC Mode

SUMMARY STEPS

1. **configure terminal**
2. **wireless multicast**
3. **no ap capwap multicast** *ipaddr*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.
Step 2	**wireless multicast** **Example:** `Switch(config)# wireless multicast`	Enables the multicast traffic for wireless clients and enables mDNS bridging. The default value is disable. Add **no** in the command to disable the multicast traffic for wireless clients and disable mDNS bridging.

	Command or Action	Purpose
Step 3	**no ap capwap multicast** *ipaddr* **Example:** `Switch(config)# no ap capwap multicast 231.1.1.1`	Enables forwarding mode in multicast. Add **no** in the command to disable the multicast mode.
Step 4	**end** **Example:** `Switch(config)# end`	Exits the configuration mode. Alternatively, press **Ctrl-Z** to exit the configuration mode.

Related Topics

Information About Wireless Multicast, on page 840

Information About Wireless Multicast, on page 840

Configuring IPv6 Snooping

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 mld snooping**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.
Step 2	**ipv6 mld snooping** **Example:** `Switch(config)# ipv6 mld snooping`	Enables MLD snooping.

Configuring IPv6 Snooping Policy

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 snooping policy** *policy-name*
3. **security-level guard**
4. **device-role node**
5. **protocol** {**dhcp** | **ndp**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.
Step 2	**ipv6 snooping policy** *policy-name* **Example:** `Switch(config)# ipv6 snooping policy mypolicy`	Configures an IPv6 snooping policy with a name.
Step 3	**security-level guard** **Example:** `Switch(config-ipv6-snooping)# security-level guard`	Configures security level to inspect and drop any unauthorized messages.
Step 4	**device-role node** **Example:** `Switch(config-ipv6-snooping)# device-role node`	Configures the role of the device, which is a node, to the attached port.
Step 5	**protocol** {**dhcp** | **ndp**} **Example:** `Switch(config-ipv6-snooping)# protocol ndp`	Sets the protocol to glean addresses in DHCP or NDP packets.

Configuring Layer 2 Port as Multicast Router Port

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 mld snooping vlan** *vlan-id* **mrouter interface Port-channel** *port-channel-interface-number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.
Step 2	**ipv6 mld snooping vlan** *vlan-id* **mrouter interface Port-channel** *port-channel-interface-number* **Example:** `Switch(config)# ipv6 mld snooping vlan 2 mrouter interface Port-channel 22`	Configures a Layer 2 port as a Multicast router port. The VLAN is the client VLAN.

Configuring RA Guard

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 nd raguard policy** *policy-name*
3. **trusted-port**
4. **device-role** {**host** | **monitor** | **router** | **switch**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.
Step 2	**ipv6 nd raguard policy** *policy-name* **Example:** `Switch(config)# ipv6 nd raguard policy myraguardpolicy`	Configures a policy for RA Guard.
Step 3	**trusted-port** **Example:** `Switch(config-nd-raguard)# trusted-port`	Sets up a trusted port.
Step 4	**device-role** {**host** | **monitor** | **router** | **switch**} **Example:** `Switch(config-nd-raguard)# device-role router`	Sets the role of the device attached to the port.

Configuring Non-IP Wireless Multicast

SUMMARY STEPS

1. **configure terminal**
2. **wireless multicast non-ip**
3. **wireless multicast non-ip** *vlanid*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.

Step 2	**wireless multicast non-ip**
	
	Example:
	`Switch(config)# wireless multicast non-ip`
	
	`Switch(config)# no wireless multicast non-ip`
		Enables non-IP multicast in all VLANs. Default value is **enable**. Wireless multicast must be enabled for the traffic to pass. Add **no** in the command to disable the non-IP multicast in all VLANs.

Step 3	**wireless multicast non-ip** *vlanid*
	
	Example:
	`Switch(config)# wireless multicast non-ip 5`
	
	`Switch(config)# no wireless multicast non-ip 5`
		Enables non-IP multicast per VLAN. Default value is **enable**. Both wireless multicast and wireless multicast non-IP must be enabled for traffic to pass. Add **no** in the command to disable the non-IP multicast per VLAN.

Step 4	**end** **Example:** `Switch(config)# end`	Exits the configuration mode. Alternatively, press **Ctrl-Z** to exit the configuration mode.

Configuring Wireless Broadcast

SUMMARY STEPS

1. **configure terminal**
2. **wireless broadcast**
3. **wireless broadcast vlan** *vlanid*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.

Step 2	**wireless broadcast**
	
	Example:
	`Switch(config)# wireless broadcast`
	
	`Switch(config)# no wireless broadcast`
		Enables broadcast packets for wireless clients. Default value is disable. Enabling **wireless broadcast** enables broadcast traffic for each VLAN. Add **no** in the command to disable broadcasting packets.

Step 3	**wireless broadcast vlan** *vlanid*
	
	Example:
	`Switch(config)# wireless broadcast vlan 3`
	
	`Switch(config)# no wireless broadcast vlan 3`
		Enables broadcast packets for single VLAN. Default value is **enable**. Wireless broadcast must be enabled for broadcasting. Add **no** in the command to disable the broadcast traffic for each VLAN.

Step 4	**end** **Example:** `Switch(config)# end`	Exits the configuration mode. Alternatively, press **Ctrl-Z** to exit the configuration mode.

Configuring IP Multicast VLAN for WLAN

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan_name*
3. **shutdown**
4. **ip multicast vlan** {*vlan_name vlan_id*}
5. **no shutdown**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.
Step 2	**wlan** *wlan_name* **Example:** `Switch(config)# wlan test 1`	Enters the configuration mode to configure various parameters in the WLAN.
Step 3	**shutdown** **Example:** `Switch(config-wlan)# shutdown`	Disables WLAN.

Step 4	**ip multicast vlan** {*vlan_name vlan_id*}
	
	Example:
	`Switch(config-wlan)# ip multicast vlan 5`
	`Switch(config-wlan)# no ip multicast vlan 5`
		Configures multicast VLAN for WLAN. Add **no** in the command to disable the multicast VLAN for WLAN.

Step 5	**no shutdown** **Example:** `Switch(config-wlan)# no shutdown`	Enables the disabled WLAN.
Step 6	**end** **Example:** `Switch(config)# end`	Exits the configuration mode. Alternatively, press **Ctrl-Z** to exit the configuration mode.

Related Topics

Information About Multicast Optimization, on page 841
Information About Multicast Optimization, on page 841

Monitoring Wireless Multicast

Table 102: Commands for Monitoring Wireless Multicast

Commands	Description
show wireless multicast	Displays the multicast status and IP multicast mode, each VLAN's broadcast and non-IP multicast status. Also displays the mDNS bridging state.
show wireless multicast group summary	Displays all (Source, Group and VLAN) lists and the corresponding MGID value.
show wireless multicast [**source** *source*] **group** *group* **vlan** *vlanid*	Displays details of the given (S,G,V) and shows all of the clients associated with it and their MC2UC status .
show ip igmp snooping wireless mcast-spi-count	Displays statistics of the number of multicast SPIs per MGID sent between IOS and the Wireless Controller Module.
show ip igmp snooping wireless mgid	Displays the MGID mappings.
show ip igmp snooping igmpv2-tracking	Displays the client-to-SGV mappings and SGV-to-client mappings.
show ip igmp snooping querier vlan *vlanid*	Displays IGMP querier information for the specified VLAN.
show ip igmp snooping querier detail	Displays detailed IGMP querier information of all the VLANs.
show ipv6 mld snooping querier vlan *vlanid*	Displays MLD querier information for the specified VLAN.
show ipv6 mld snooping wireless mgid	Displays MGIDs for IPv6 multicast group.

Where to Go Next for Wireless Multicast

You can configure the following:

- IGMP
- PIM
- SSM
- IP Multicast Routing
- Service Discovery Gateway

CHAPTER 40

Configuring PIM

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for Configuring PIM

The following are the prerequisites for configuring PIM and PIM stub routing:

- Before configuring PIM stub routing, you must have IP multicast routing configured on both the stub router and the central router. You must also have PIM mode (dense-mode, sparse-mode, or sparse-dense-mode) configured on the uplink interface of the stub router.
- Before configuring PIM stub routing, you must also configure either Enhanced Interior Gateway Routing Protocol (EIGRP) stub routing or Open Shortest Path First (OSPF) stub routing on the switch. The PIM stub router does not route the transit traffic between the distribution routers. Unicast (EIGRP) stub routing enforces this behavior. You must configure unicast stub routing to assist the PIM stub router behavior.

Note For information about EIGRP or OSPF configurations, see the *Catalyst 3850 Routing Configuration Guide, Release 3SE*.

Restrictions for Configuring PIM

The following are the restrictions for configuring PIM:

- PIM
  - PIM is not supported when running the LAN Base feature set.
  - Bidirectional PIM is not supported.
- PIM stub routing
  - The IP Services image contains complete multicast routing.
  - In a network using PIM stub routing, the only allowable route for IP traffic to the user is through a switch that is configured with PIM stub routing.
  - The redundant PIM stub router topology is not supported. Only the nonredundant access router topology is supported by the PIM stub feature.
  - Only directly connected multicast (IGMP) receivers and sources are allowed in the Layer 2 access domains. The PIM protocol is not supported in access domains.

Restrictions for Configuring Auto-RP

The following are restrictions for configuring Auto-RP (if used in your network configuration):

- Auto-RP is not supported when running the LAN Base feature set.
- If you configure PIM in sparse mode or sparse-dense mode and do not configure Auto-RP, you must manually configure an RP.

- If routed interfaces are configured in sparse mode, Auto-RP can still be used if all devices are configured with a manual RP address for the Auto-RP groups.
- If routed interfaces are configured in sparse mode and you enter the **ip pim autorp listener** global configuration command, Auto-RP can still be used even if all devices are not configured with a manual RP address for the Auto-RP groups.

Restrictions for Configuring Auto-RP and BSR

The following are restrictions for configuring Auto-RP and BSR (if used in your network configuration):

- If your network is all Cisco routers and multilayer switches, you can use either Auto-RP or BSR.
- If you have non-Cisco routers in your network, you must use BSR.
- If you have Cisco PIMv1 and PIMv2 routers and multilayer switches and non-Cisco routers, you must use both Auto-RP and BSR. If your network includes routers from other vendors, configure the Auto-RP mapping agent and the BSR on a Cisco PIMv2 device. Ensure that no PIMv1 device is located in the path a between the BSR and a non-Cisco PIMv2 device.

Note There are two approaches to using PIMv2. You can use Version 2 exclusively in your network or migrate to Version 2 by employing a mixed PIM version environment.

- Because bootstrap messages are sent hop-by-hop, a PIMv1 device prevents these messages from reaching all routers and multilayer switches in your network. Therefore, if your network has a PIMv1 device in it and only Cisco routers and multilayer switches, it is best to use Auto-RP.
- If you have a network that includes non-Cisco routers, configure the Auto-RP mapping agent and the BSR on a Cisco PIMv2 router or multilayer switch. Ensure that no PIMv1 device is on the path between the BSR and a non-Cisco PIMv2 router.
- If you have non-Cisco PIMv2 routers that need to interoperate with Cisco PIMv1 routers and multilayer switches, both Auto-RP and a BSR are required. We recommend that a Cisco PIMv2 device be both the Auto-RP mapping agent and the BSR.

Information About PIM

Protocol-Independent Multicast (PIM) is called protocol-independent because regardless of the unicast routing protocols used to populate the unicast routing table, PIM uses this information to perform multicast forwarding instead of maintaining a separate multicast routing table.

PIM can leverage whichever unicast routing protocols are used to populate the unicast routing table, including EIGRP, OSPF, BGP, or static routes. PIM uses this unicast routing information to perform the multicast forwarding function, so it is IP protocol independent. Although PIM is called a multicast routing protocol, it actually uses the unicast routing table to perform the reverse path forwarding (RPF) check function instead of building up a completely independent multicast routing table. PIM does not send and receive multicast routing updates between routers as the other routing protocols do.

PIM is defined in RFC 4601, *Protocol-Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification*. PIM is defined in these Internet Engineering Task Force (IETF) Internet drafts:

- *Protocol Independent Multicast (PIM): Motivation and Architecture*
- *Protocol Independent Multicast (PIM), Dense Mode Protocol Specification*
- *Protocol Independent Multicast (PIM), Sparse Mode Protocol Specification*
- *draft-ietf-idmr-igmp-v2-06.txt, Internet Group Management Protocol, Version 2*
- *draft-ietf-pim-v2-dm-03.txt, PIM Version 2 Dense Mode*

PIM Versions

PIMv2 includes these improvements over PIMv1:

- A single, active rendezvous point (RP) exists per multicast group, with multiple backup RPs. This single RP compares to multiple active RPs for the same group in PIMv1.
- A bootstrap router (BSR) provides a fault-tolerant, automated RP discovery and distribution function that enables routers and multilayer switches to dynamically learn the group-to-RP mappings.
- Sparse mode and dense mode are properties of a group, as opposed to an interface.

Note We strongly recommend using sparse-dense mode as opposed to either sparse mode or dense mode only.

- PIM join and prune messages have more flexible encoding for multiple address families.
- A more flexible hello packet format replaces the query packet to encode current and future capability options.
- Register messages sent to an RP specify whether they are sent by a border router or a designated router.
- PIM packets are no longer inside IGMP packets; they are standalone packets.

PIMv1 and PIMv2 Interoperability

To avoid misconfiguring multicast routing on your switch, review the information in this section.

The Cisco PIMv2 implementation provides interoperability and transition between Version 1 and Version 2, although there might be some minor problems.

You can upgrade to PIMv2 incrementally. PIM Versions 1 and 2 can be configured on different routers and multilayer switches within one network. Internally, all routers and multilayer switches on a shared media network must run the same PIM version. Therefore, if a PIMv2 device detects a PIMv1 device, the Version 2 device downgrades itself to Version 1 until all Version 1 devices have been shut down or upgraded.

PIMv2 uses the BSR to discover and announce RP-set information for each group prefix to all the routers and multilayer switches in a PIM domain. PIMv1, together with the Auto-RP feature, can perform the same tasks as the PIMv2 BSR. However, Auto-RP is a standalone protocol, separate from PIMv1, and is a proprietary Cisco protocol. PIMv2 is a standards track protocol in the IETF.

Note We recommend that you use PIMv2. The BSR function interoperates with Auto-RP on Cisco routers and multilayer switches.

When PIMv2 devices interoperate with PIMv1 devices, Auto-RP should have already been deployed. A PIMv2 BSR that is also an Auto-RP mapping agent automatically advertises the RP elected by Auto-RP. That is, Auto-RP sets its single RP on every router or multilayer switch in the group. Not all routers and switches in the domain use the PIMv2 hash function to select multiple RPs.

Dense-mode groups in a mixed PIMv1 and PIMv2 region need no special configuration; they automatically interoperate.

Sparse-mode groups in a mixed PIMv1 and PIMv2 region are possible because the Auto-RP feature in PIMv1 interoperates with the PIMv2 RP feature. Although all PIMv2 devices can also use PIMv1, we recommend that the RPs be upgraded to PIMv2. To ease the transition to PIMv2, we recommend:

- Using Auto-RP throughout the region.
- Configuring sparse-dense mode throughout the region.

If Auto-RP is not already configured in the PIMv1 regions, configure Auto-RP.

PIM Modes

PIM can operate in dense mode (DM), sparse mode (SM), or in sparse-dense mode (PIM DM-SM), which handles both sparse groups and dense groups at the same time.

PIM DM

PIM DM builds source-based multicast distribution trees. In dense mode, a PIM DM router or multilayer switch assumes that all other routers or multilayer switches forward multicast packets for a group. If a PIM DM device receives a multicast packet and has no directly connected members or PIM neighbors present, a prune message is sent back to the source to stop unwanted multicast traffic. Subsequent multicast packets are not flooded to this router or switch on this pruned branch because branches without receivers are pruned from the distribution tree, leaving only branches that contain receivers.

When a new receiver on a previously pruned branch of the tree joins a multicast group, the PIM DM device detects the new receiver and immediately sends a graft message up the distribution tree toward the source. When the upstream PIM DM device receives the graft message, it immediately puts the interface on which the graft was received into the forwarding state so that the multicast traffic begins flowing to the receiver.

PIM-SM

PIM-SM uses shared trees and shortest-path-trees (SPTs) to distribute multicast traffic to multicast receivers in the network. In PIM-SM, a router or multilayer switch assumes that other routers or switches do not forward multicast packets for a group, unless there is an explicit request for the traffic (join message). When a host joins a multicast group using IGMP, its directly connected PIM-SM device sends PIM join messages toward the root, also known as the rendezvous point (RP). This join message travels router-by-router toward the root, constructing a branch of the shared tree as it goes.

The RP keeps track of multicast receivers. It also registers sources through register messages received from the source’s first-hop router (designated router [DR]) to complete the shared tree path from the source to the receiver. When using a shared tree, sources must send their traffic to the RP so that the traffic reaches all receivers.

Prune messages are sent up the distribution tree to prune multicast group traffic. This action permits branches of the shared tree or SPT that were created with explicit join messages to be torn down when they are no longer needed.

When the number of PIM-enabled interfaces exceeds the hardware capacity and PIM-SM is enabled with the SPT threshold is set to **infinity**, the switch does not create (source, group (S, G)) entries in the multicast routing table for the some directly connected interfaces if they are not already in the table. The switch might not correctly forward traffic from these interfaces.

Multicast Source Discovery Protocol (MSDP)

Multicast Source Discovery Protocol (MSDP) is used for inter-domain source discovery when PIM SM is used. Each PIM administrative domain has its own RP. In order for the RP in one domain to signal new sources to the RP in the other domain, MSDP is used.

When RP in a domain receives a PIM register message for a new source, with MSDP configured it sends a new source-active (SA) message to all its MSDP peers in other domains. Each intermediate MSDP peer floods this SA message away from the originating RP. The MSDP peers install this SA message in their MSDP sa-cache. If the RPs in other domains have any join requests for the group in the SA message (indicated by the presence of a (*,G) entry with non empty outgoing interface list), the domain is interested in the group, and the RP triggers an (S,G) join toward the source.

PIM Stub Routing

The PIM stub routing feature, available in all of the switch software images, reduces resource usage by moving routed traffic closer to the end user.

The PIM stub routing feature supports multicast routing between the distribution layer and the access layer. It supports two types of PIM interfaces: uplink PIM interfaces and PIM passive interfaces. A routed interface configured with the PIM passive mode does not pass or forward PIM control traffic, it only passes and forwards IGMP traffic.

In a network using PIM stub routing, the only allowable route for IP traffic to the user is through a switch that is configured with PIM stub routing. PIM passive interfaces are connected to Layer 2 access domains, such as VLANs, or to interfaces that are connected to other Layer 2 devices. Only directly connected multicast (IGMP) receivers and sources are allowed in the Layer 2 access domains. The PIM passive interfaces do not send or process any received PIM control packets.

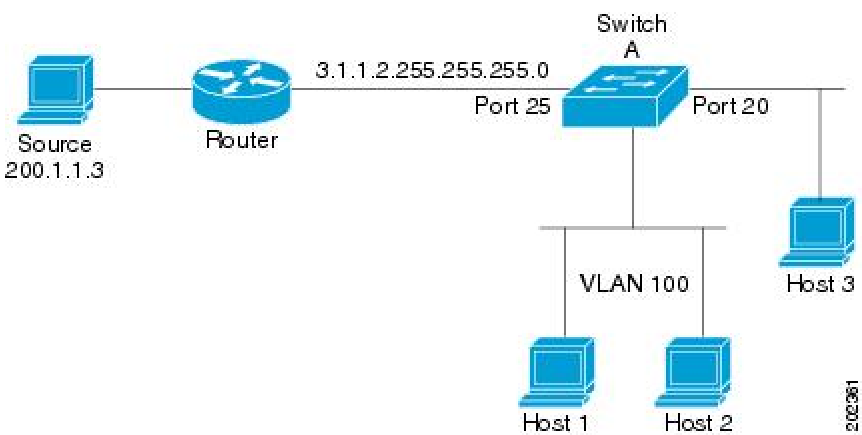
When using PIM stub routing, you should configure the distribution and remote routers to use IP multicast routing and configure only the switch as a PIM stub router. The switch does not route transit traffic between distribution routers. You also need to configure a routed uplink port on the switch. The switch uplink port cannot be used with SVIs. If you need PIM for an SVI uplink port, you should upgrade to the IP Services feature set.

You must also configure EIGRP stub routing when configuring PIM stub routing on the switch. For information about this procedure, refer to the *Catalyst 3850 IP Routing Configuration Guide*.

The redundant PIM stub router topology is not supported. The redundant topology exists when there is more than one PIM router forwarding multicast traffic to a single access domain. PIM messages are blocked, and the PIM asset and designated router election mechanisms are not supported on the PIM passive interfaces. Only the nonredundant access router topology is supported by the PIM stub feature. By using a nonredundant topology, the PIM passive interface assumes that it is the only interface and designated router on that access domain.

In the following figure, Switch A routed uplink port 25 is connected to the router and PIM stub routing is enabled on the VLAN 100 interfaces and on Host 3. This configuration allows the directly connected hosts to receive traffic from multicast source 200.1.1.3.

Figure 27: PIM Stub Router Configuration



Related Topics

Enabling PIM Stub Routing , on page 862
Example: Enabling PIM Stub Routing, on page 887

IGMP Helper

PIM stub routing moves routed traffic closer to the end user and reduces network traffic. You can also reduce traffic by configuring a stub router (switch) with the IGMP helper feature.

You can configure a stub router (switch) with the **ip igmp helper help-address** interface configuration command to enable the switch to send reports to the next-hop interface. Hosts that are not directly connected to a downstream router can then join a multicast group sourced from an upstream network. The IGMP packets from a host wanting to join a multicast stream are forwarded upstream to the next-hop device when this feature is configured. When the upstream central router receives the helper IGMP reports or leaves, it adds or removes the interfaces from its outgoing interface list for that group.

For complete syntax and usage information for the **ip igmp helper-address** command, see the *IP Multicast Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*.

Auto-RP

The PIM-SM protocols require the presence of a rendezvous point (RP) in the network. An RP acts as the meeting place for sources and receivers of multicast data. If a static RP configuration is used, then the configuration needs to be applied on all the routers in the multicast network. To automate this process, the Auto-RP protocol was devised.

This Cisco proprietary feature eliminates the need to manually configure the RP information in every router and multilayer switch in the network. For Auto-RP to work, you configure a Cisco router or multilayer switch as the mapping agent. It uses IP multicast to learn which routers or switches in the network are possible candidate RPs to receive candidate RP announcements. Candidate RPs periodically send multicast RP-announce messages to a particular group or group range to announce their availability.

Mapping agents listen to these candidate RP announcements and use the information to create entries in their group-to-RP mapping caches. Only one mapping cache entry is created for any group-to-RP range received, even if multiple candidate RPs are sending RP announcements for the same range. As the RP-announce messages arrive, the mapping agent selects the router or switch with the highest IP address as the active RP and stores this RP address in the group-to-RP mapping cache.

Mapping agents periodically multicast the contents of their group-to-RP mapping caches. Thus, all routers and switches automatically discover which RP to use for the groups that they support. If a router or switch fails to receive RP-discovery messages and the group-to-RP mapping information expires, it changes to a statically configured RP that was defined with the **ip pim rp-address** global configuration command. If no statically configured RP exists, the router or switch changes the group to dense-mode operation.

Multiple RPs serve different group ranges or serve as hot backups of each other.

Related Topics

Setting Up Auto-RP in a New Internetwork , on page 867

Example: Configuring Auto-RP, on page 888

Auto-RP Benefits

Auto-RP uses IP multicast to automate the distribution of group-to-RP mappings to all Cisco routers and multilayer switches in a PIM network. Auto-RP has these benefits:

- Easy to use multiple RPs within a network to serve different group ranges.
- Provides load splitting among different RPs and arrangement of RPs according to the location of group participants.
- Avoids inconsistent, manual RP configurations on every router and multilayer switch in a PIM network, which can cause connectivity problems.

PIM v2 BSR

PIMv2 BSR (Bootstrap Router) is another method to distribute group-to-RP mapping information to all PIM routers and multilayer switches in the network. It eliminates the need to manually configure RP information in every router and switch in the network. However, instead of using IP multicast to distribute group-to-RP mapping information, BSR uses hop-by-hop flooding of special BSR messages to distribute the mapping information.

The BSR is elected from a set of candidate routers and switches in the domain that have been configured to function as BSRs. The election mechanism is similar to the root-bridge election mechanism used in bridged LANs. The BSR election is based on the BSR priority of the device contained in the BSR messages that are sent hop-by-hop through the network. Each BSR device examines the message and forwards out all interfaces only the message that has either a higher BSR priority than its BSR priority or the same BSR priority, but with a higher BSR IP address. Using this method, the BSR is elected.

The elected BSR sends BSR messages with a TTL of 1. Neighboring PIMv2 routers or multilayer switches receive the BSR message and multicast it out all other interfaces (except the one on which it was received) with a TTL of 1. In this way, BSR messages travel hop-by-hop throughout the PIM domain. Because BSR messages contain the IP address of the current BSR, the flooding mechanism enables candidate RPs to automatically learn which device is the elected BSR.

Candidate RPs send candidate RP advertisements showing the group range for which they are responsible to the BSR, which stores this information in its local candidate-RP cache. The BSR periodically advertises the contents of this cache in BSR messages to all other PIM devices in the domain. These messages travel hop-by-hop through the network to all routers and switches, which store the RP information in the BSR message in their local RP cache. The routers and switches select the same RP for a given group because they all use a common RP hashing algorithm.

Related Topics

Configuring Candidate BSRs , on page 878

Example: Configuring Candidate BSRs, on page 889

Multicast Forwarding and Reverse Path Check

With unicast routing, routers and multilayer switches forward traffic through the network along a single path from the source to the destination host whose IP address appears in the destination address field of the IP packet. Each router and switch along the way makes a unicast forwarding decision, using the destination IP address in the packet, by looking up the destination address in the unicast routing table and forwarding the packet through the specified interface to the next hop toward the destination.

With multicasting, the source is sending traffic to an arbitrary group of hosts represented by a multicast group address in the destination address field of the IP packet. To decide whether to forward or drop an incoming multicast packet, the router or multilayer switch uses a reverse path forwarding (RPF) check on the packet as follows:

1. The router or multilayer switch examines the source address of the arriving multicast packet to decide whether the packet arrived on an interface that is on the reverse path back to the source.
2. If the packet arrives on the interface leading back to the source, the RPF check is successful and the packet is forwarded to all interfaces in the outgoing interface list (which might not be all interfaces on the router).
3. If the RPF check fails, the packet is discarded.

Some multicast routing protocols, such as DVMRP, maintain a separate multicast routing table and use it for the RPF check. However, PIM uses the unicast routing table to perform the RPF check.

Note DVMRP is not supported on the switch.

The following figure shows port 2 receiving a multicast packet from source 151.10.3.21. The following table shows that the port on the reverse path to the source is port 1, not port 2. Because the RPF check fails, the multilayer switch discards the packet. Another multicast packet from source 151.10.3.21 is received on port

1, and the routing table shows this port is on the reverse path to the source. Because the RPF check passes, the switch forwards the packet to all port in the outgoing port list

Figure 28: RPF Check

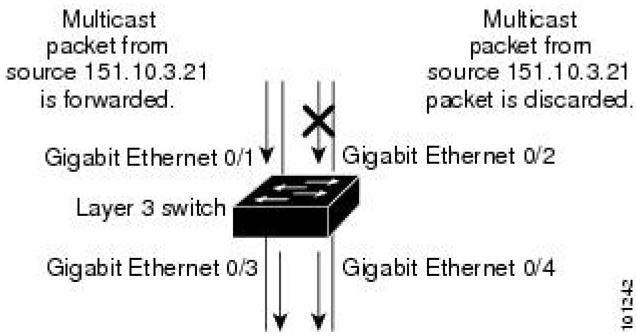


Table 103: Routing Table Example for an RPF Check

Network	Port
151.10.0.0/16	Gigabit Ethernet 1/0/1
198.14.32.0/32	Gigabit Ethernet 1/0/3
204.1.16.0/24	Gigabit Ethernet 1/0/4

PIM uses both source trees and RP-rooted shared trees to forward datagrams. The RPF check is performed differently for each:

- If a PIM router or multilayer switch has a source-tree state (that is, an (S, G) entry is present in the multicast routing table), it performs the RPF check against the IP address of the source of the multicast packet.
- If a PIM router or multilayer switch has a shared-tree state (and no explicit source-tree state), it performs the RPF check on the RP address (which is known when members join the group).

Sparse-mode PIM uses the RPF lookup function to decide where it needs to send joins and prunes:

- (S, G) joins (which are source-tree states) are sent toward the source.
- (*,G) joins (which are shared-tree states) are sent toward the RP.

DVMRP and dense-mode PIM use only source trees and use RPF.

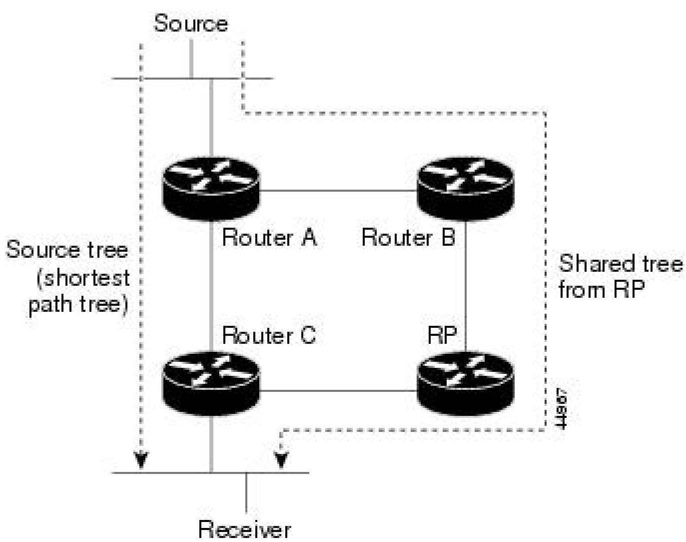
Note DVMRP is not supported on the switch.

PIM Shared Tree and Source Tree

By default, members of a group receive data from senders to the group across a single data-distribution tree rooted at the RP.

The following figure shows this type of shared-distribution tree. Data from senders is delivered to the RP for distribution to group members joined to the shared tree.

Figure 29: Shared Tree and Source Tree (Shortest-Path Tree)



If the data rate warrants, leaf routers (routers without any downstream connections) on the shared tree can use the data distribution tree rooted at the source. This type of distribution tree is called a shortest-path tree or source tree. By default, the software switches to a source tree upon receiving the first data packet from a source.

This process describes the move from a shared tree to a source tree:

1. A receiver joins a group; leaf Router C sends a join message toward the RP.
2. The RP puts a link to Router C in its outgoing interface list.
3. A source sends data; Router A encapsulates the data in a register message and sends it to the RP.
4. The RP forwards the data down the shared tree to Router C and sends a join message toward the source. At this point, data might arrive twice at Router C, once encapsulated and once natively.
5. When data arrives natively (unencapsulated) at the RP, it sends a register-stop message to Router A.
6. By default, reception of the first data packet prompts Router C to send a join message toward the source.
7. When Router C receives data on (S, G), it sends a prune message for the source up the shared tree.
8. The RP deletes the link to Router C from the outgoing interface of (S, G). The RP triggers a prune message toward the source.

Join and prune messages are sent for sources and RPs. They are sent hop-by-hop and are processed by each PIM device along the path to the source or RP. Register and register-stop messages are not sent hop-by-hop.

They are sent by the designated router that is directly connected to a source and are received by the RP for the group.

Multiple sources sending to groups use the shared tree.

You can configure the PIM device to stay on the shared tree. You can configure the PIM device to stay on the shared tree. For more information, see Delaying the Use of PIM Shortest-Path Tree , on page 882.

Default PIM Routing Configuration

This table displays the default PIM routing configuration for the switch.

Table 104: Default Multicast Routing Configuration

Feature	Default Setting
Multicast routing	Disabled on all interfaces.
PIM version	Version 2.
PIM mode	No mode is defined.
PIM stub routing	None configured.
PIM RP address	None configured.
PIM domain border	Disabled.
PIM multicast boundary	None.
Candidate BSRs	Disabled.
Candidate RPs	Disabled.
Shortest-path tree threshold rate	0 kb/s.
PIM router query message interval	30 seconds.

How to Configure PIM

Enabling PIM Stub Routing

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip pim passive**
4. **end**
5. **show ip pim interface**
6. **show ip igmp groups detail**
7. **show ip mroute**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Specifies the interface on which you want to enable PIM stub routing, and enters interface configuration mode. The specified interface must be one of the following: • A routed port—A physical port that has been configured as a Layer 3 port by entering the **no switchport** interface configuration command. You will also need to enable IP PIM sparse-dense-mode on the interface, and join the interface as a statically connected member to an IGMP static group. For a configuration example, see Example: Interface Configuration as a Routed Port, on page 834 • An SVI—A VLAN interface created by using the **interface vlan** *vlan-id* global configuration command. You will also need to enable IP PIM sparse-dense-mode on the VLAN, join the VLAN as a statically connected member to an IGMP static group, and then enable IGMP snooping on the VLAN, the IGMP static group, and physical interface. For a configuration example, see Example: Interface Configuration as an SVI, on page 835 These interfaces must have IP addresses assigned to them.
Step 3	**ip pim passive** **Example:** `Switch(config-if)# ip pim passive`	Configures the PIM stub feature on the interface.

	Command or Action	Purpose
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show ip pim interface** **Example:** `Switch# show ip pim interface`	(Optional) Displays the PIM stub that is enabled on each interface.
Step 6	**show ip igmp groups detail** **Example:** `Switch# show ip igmp groups detail`	(Optional) Displays the interested clients that have joined the specific multicast source group.
Step 7	**show ip mroute** **Example:** `Switch# show ip mroute`	(Optional) Displays the IP multicast routing table.
Step 8	**show running-config** **Example:** `Switch# show running-config`	(Optional) Verifies your entries.
Step 9	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

PIM Stub Routing, on page 856

Example: Enabling PIM Stub Routing, on page 887

Configuring a Rendezvous Point

You must have a rendezvous point (RP), if the interface is in sparse-dense mode and if you want to handle the group as a sparse group. You can use several methods, as described in these sections:

- Manual assignment

  For information about this procedure, see Manually Assigning an RP to Multicast Groups , on page 865.

- As a standalone, Cisco-proprietary protocol separate from PIMv1

  For information about these procedures, see the following sections:

  - Setting Up Auto-RP in a New Internetwork , on page 867
  - Adding Auto-RP to an Existing Sparse-Mode Cloud , on page 870
  - Preventing Join Messages to False RPs , on page 872
  - Filtering Incoming RP Announcement Messages , on page 872

- Using a standards track protocol in the Internet Engineering Task Force (IETF)

  For information about this procedure, see Configuring PIMv2 BSR, on page 874.

Note You can use Auto-RP, BSR, or a combination of both, depending on the PIM version that you are running and the types of routers in your network. For information about working with different PIM versions in your network, see PIMv1 and PIMv2 Interoperability, on page 854.

Manually Assigning an RP to Multicast Groups

If the rendezvous point (RP) for a group is learned through a dynamic mechanism (such as Auto-RP or BSR), you need not perform this task for that RP.

Senders of multicast traffic announce their existence through register messages received from the source first-hop router (designated router) and forwarded to the RP. Receivers of multicast packets use RPs to join a multicast group by using explicit join messages.

Note RPs are not members of the multicast group; they serve as a *meeting place* for multicast sources and group members.

You can configure a single RP for multiple groups defined by an access list. If there is no RP configured for a group, the multilayer switch responds to the group as dense and uses the dense-mode PIM techniques.

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **ip pim rp-address** *ip-address* [*access-list-number*] [**override**]
3. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip pim rp-address** *ip-address* [*access-list-number*] [**override**] **Example:** `Switch(config)# ip pim rp-address 10.1.1.1 20 override`	Configures the address of a PIM RP. By default, no PIM RP address is configured. You must configure the IP address of RPs on all routers and multilayer switches (including the RP). **Note** If there is no RP configured for a group, the switch treats the group as dense, using the dense-mode PIM techniques. A PIM device can be an RP for more than one group. Only one RP address can be used at a time within a PIM domain. The access list conditions specify for which groups the device is an RP. • For *ip-address*, enter the unicast address of the RP in dotted-decimal notation. • (Optional) For *access-list-number*, enter an IP standard access list number from 1 to 99. If no access list is configured, the RP is used for all groups. • (Optional) The **override** keyword indicates that if there is a conflict between the RP configured with this command and one learned by Auto-RP or BSR, the RP configured with this command prevails.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] **Example:** `Switch(config)# access-list 25 permit 10.5.0.1 255.224.0.0`	Creates a standard access list, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number specified in Step 2. • The **deny** keyword denies access if the conditions are matched. • The **permit** keyword permits access if the conditions are matched. • For *source*, enter the multicast group address for which the RP should be used. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. The access list is always terminated by an implicit deny statement for everything.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Setting Up Auto-RP in a New Internetwork

If you are setting up Auto-RP in a new internetwork, you do not need a default RP because you configure all the interfaces for sparse-dense mode.

Note Omit Step 3 in the following procedure, if you want to configure a PIM router as the RP for the local group.

SUMMARY STEPS

1. **show running-config**
2. **configure terminal**
3. **ip pim send-rp-announce** *interface-id* **scope** *ttl* **group-list** *access-list-number* **interval** *seconds*
4. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
5. **ip pim send-rp-discovery scope** *ttl*
6. **end**
7. **show running-config**
8. **show ip pim rp mapping**
9. **show ip pim rp**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show running-config** **Example:** `Switch# show running-config`	Verifies that a default RP is already configured on all PIM devices and the RP in the sparse-mode network. It was previously configured with the **ip pim rp-address** global configuration command. **Note** This step is not required for spare-dense-mode environments. The selected RP should have good connectivity and be available across the network. Use this RP for the global groups (for example, 224.x.x.x and other global groups). Do not reconfigure the group address range that this RP serves. RPs dynamically discovered through Auto-RP take precedence over statically configured RPs. Assume that it is desirable to use a second RP for the local groups.
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 3	**ip pim send-rp-announce** *interface-id* **scope** *ttl* **group-list** *access-list-number* **interval** *seconds* **Example:** `Switch(config)# ip pim send-rp-announce gigabitethernet 1/0/5 scope 20 group-list 10 interval 120`	Configures another PIM device to be the candidate RP for local groups. • For *interface-id*, enter the interface type and number that identifies the RP address. Valid interfaces include physical ports, port channels, and VLANs. • For **scope** *ttl*, specify the time-to-live value in hops. Enter a hop count that is high enough so that the RP-announce messages reach all mapping agents in the network. There is no default setting. The range is 1 to 255. • For **group-list** *access-list-number*, enter an IP standard access list number from 1 to 99. If no access list is configured, the RP is used for all groups. • For **interval** *seconds*, specify how often the announcement messages must be sent. The default is 60 seconds. The range is 1 to 16383.
Step 4	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] **Example:** `Switch(config)# access-list 10 permit 10.10.0.0`	Creates a standard access list, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number specified in Step 3. • The **deny** keyword denies access if the conditions are matched. • The **permit** keyword permits access if the conditions are matched. • For *source*, enter the multicast group address range for which the RP should be used. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore.

	Command or Action	Purpose
		Note Recall that the access list is always terminated by an implicit deny statement for everything.
Step 5	**ip pim send-rp-discovery scope** *ttl* **Example:** `Switch(config)# ip pim send-rp-discovery scope 50`	Finds a switch whose connectivity is not likely to be interrupted, and assign it the role of RP-mapping agent. For **scope** *ttl*, specify the time-to-live value in hops to limit the RP discovery packets. All devices within the hop count from the source device receive the Auto-RP discovery messages. These messages tell other devices which group-to-RP mapping to use to avoid conflicts (such as overlapping group-to-RP ranges). There is no default setting. The range is 1 to 255.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 7	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 8	**show ip pim rp mapping** **Example:** `Switch# show ip pim rp mapping`	Displays active RPs that are cached with associated multicast routing entries.
Step 9	**show ip pim rp** **Example:** `Switch# show ip pim rp`	Displays the information cached in the routing table.
Step 10	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Auto-RP, on page 857

Example: Configuring Auto-RP, on page 888

Adding Auto-RP to an Existing Sparse-Mode Cloud

This section contains suggestions for the initial deployment of Auto-RP into an existing sparse-mode cloud to minimize disruption of the existing multicast infrastructure.

This procedure is optional.

SUMMARY STEPS

1. **show running-config**
2. **configure terminal**
3. **ip pim send-rp-announce** *interface-id* **scope** *ttl* **group-list** *access-list-number* **interval** *seconds*
4. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
5. **ip pim send-rp-discovery scope** *ttl*
6. **end**
7. **show running-config**
8. **show ip pim rp mapping**
9. **show ip pim rp**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show running-config** **Example:** `Switch# show running-config`	Verifies that a default RP is already configured on all PIM devices and the RP in the sparse-mode network. It was previously configured with the **ip pim rp-address** global configuration command. **Note** This step is not required for spare-dense-mode environments. The selected RP should have good connectivity and be available across the network. Use this RP for the global groups (for example, 224.x.x.x and other global groups). Do not reconfigure the group address range that this RP serves. RPs dynamically discovered through Auto-RP take precedence over statically configured RPs. Assume that it is desirable to use a second RP for the local groups.
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 3	**ip pim send-rp-announce** *interface-id* **scope** *ttl* **group-list** *access-list-number* **interval** *seconds* **Example:** `Switch(config)# ip pim`	Configures another PIM device to be the candidate RP for local groups. • For *interface-id*, enter the interface type and number that identifies the RP address. Valid interfaces include physical ports, port channels, and VLANs.

	Command or Action	Purpose
	send-rp-announce gigabitethernet 1/0/5 scope 20 group-list 10 interval 120	• For **scope** *ttl*, specify the time-to-live value in hops. Enter a hop count that is high enough so that the RP-announce messages reach all mapping agents in the network. There is no default setting. The range is 1 to 255. • For **group-list** *access-list-number*, enter an IP standard access list number from 1 to 99. If no access list is configured, the RP is used for all groups. • For **interval** *seconds*, specify how often the announcement messages must be sent. The default is 60 seconds. The range is 1 to 16383.
Step 4	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] **Example:** `Switch(config)# access-list 10 permit  224.0.0.0 15.255.255.255`	Creates a standard access list, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number specified in Step 3. • The **deny** keyword denies access if the conditions are matched. • The **permit** keyword permits access if the conditions are matched. • For *source*, enter the multicast group address range for which the RP should be used. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 5	**ip pim send-rp-discovery scope** *ttl* **Example:** `Switch(config)# ip pim send-rp-discovery scope 50`	Finds a switch whose connectivity is not likely to be interrupted, and assigns it the role of RP-mapping agent. For **scope** *ttl*, specify the time-to-live value in hops to limit the RP discovery packets. All devices within the hop count from the source device receive the Auto-RP discovery messages. These messages tell other devices which group-to-RP mapping to use to avoid conflicts (such as overlapping group-to-RP ranges). There is no default setting. The range is 1 to 255. **Note** To remove the switch as the RP-mapping agent, use the **no ip pim send-rp-discovery** global configuration command.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 7	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.

	Command or Action	Purpose
Step 8	**show ip pim rp mapping** **Example:** `Switch#` **`show ip pim rp mapping`**	Displays active RPs that are cached with associated multicast routing entries.
Step 9	**show ip pim rp** **Example:** `Switch#` **`show ip pim rp`**	Displays the information cached in the routing table.
Step 10	**copy running-config startup-config** **Example:** `Switch#` **`copy running-config startup-config`**	(Optional) Saves your entries in the configuration file.

Preventing Join Messages to False RPs

Determine whether the **ip pim accept-rp** command was previously configured throughout the network by using the **show running-config** privileged EXEC command. If the **ip pim accept-rp** command is not configured on any device, this problem can be addressed later. In those routers or multilayer switches already configured with the **ip pim accept-rp** command, you must enter the command again to accept the newly advertised RP.

To accept all RPs advertised with Auto-RP and reject all other RPs by default, use the **ip pim accept-rp auto-rp** global configuration command.

This procedure is optional.

Related Topics

Example: Preventing Join Messages to False RPs, on page 889

Filtering Incoming RP Announcement Messages

You can add configuration commands to the mapping agents to prevent a maliciously configured router from masquerading as a candidate RP and causing problems.

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **ip pim rp-announce-filter rp-list** *access-list-number* **group-list** *access-list-number*
3. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip pim rp-announce-filter rp-list** *access-list-number* **group-list** *access-list-number* **Example:** `Switch(config)# ip pim rp-announce-filter rp-list 10 group-list 14`	Filters incoming RP announcement messages. Enter this command on each mapping agent in the network. Without this command, all incoming RP-announce messages are accepted by default. For **rp-list** *access-list-number*, configure an access list of candidate RP addresses that, if permitted, is accepted for the group ranges supplied in the **group-list** *access-list-number* variable. If this variable is omitted, the filter applies to all multicast groups. If more than one mapping agent is used, the filters must be consistent across all mapping agents to ensure that no conflicts occur in the group-to-RP mapping information.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] **Example:** `Switch(config)# access-list 10 permit 10.8.1.0 255.255.224.0`	Creates a standard access list, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number specified in Step 2. • The **deny** keyword denies access if the conditions are matched. • The **permit** keyword permits access if the conditions are matched. • Create an access list that specifies from which routers and multilayer switches the mapping agent accepts candidate RP announcements (rp-list ACL). • Create an access list that specifies the range of multicast groups from which to accept or deny (group-list ACL). • For *source*, enter the multicast group address range for which the RP should be used.

	Command or Action	Purpose
		• (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. The access list is always terminated by an implicit deny statement for everything.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Example: Filtering Incoming RP Announcement Messages, on page 888

Configuring PIMv2 BSR

The process for configuring PIMv2 BSR may involve the following optional tasks:

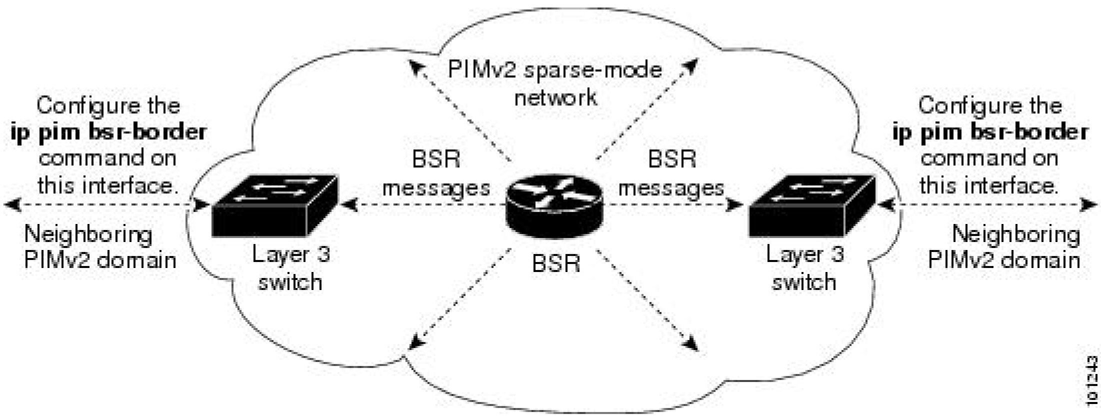
- Defining the PIM domain border
- Defining the IP multicast boundary
- Configuring candidate BSRs
- Configuring candidate RPs

Defining the PIM Domain Border

As IP multicast becomes more widespread, the chance of one PIMv2 domain bordering another PIMv2 domain increases. Because two domains probably do not share the same set of RPs, BSR, candidate RPs, and candidate BSRs, you need to constrain PIMv2 BSR messages from flowing into or out of the domain. Allowing messages to leak across the domain borders could adversely affect the normal BSR election mechanism and elect a single BSR across all bordering domains and comingle candidate RP advertisements, resulting in the election of RPs in the wrong domain.

This figure displays how you can configure the PIM domain border by using the **ip pim bsr-border** command.

Figure 30: Constraining PIMv2 BSR Messages



This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip pim bsr-border**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Specifies the interface to be configured, and enters interface configuration mode.
Step 3	**ip pim bsr-border**	Defines a PIM bootstrap message boundary for the PIM domain.

	Command or Action	Purpose
	Example: `Switch(config-if)# ip pim bsr-border`	Enter this command on each interface that connects to other bordering PIM domains. This command instructs the switch to neither send nor receive PIMv2 BSR messages on this interface. **Note** To remove the PIM border, use the **no ip pim bsr-border** interface configuration command.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Defining the IP Multicast Boundary

You define a multicast boundary to prevent Auto-RP messages from entering the PIM domain. You create an access list to deny packets destined for 224.0.1.39 and 224.0.1.40, which carry Auto-RP information.

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **access-list** *access-list-number* **deny** *source* [*source-wildcard*]
3. **interface** *interface-id*
4. **ip multicast boundary** *access-list-number*
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**access-list** *access-list-number* **deny** *source* [*source-wildcard*]
	
	Example:
	`Switch(config)#`
	`access-list 12 deny 224.0.1.39`
	`access-list 12 deny 224.0.1.40`
		Creates a standard access list, repeating the command as many times as necessary.
		• For *access-list-number*, the range is 1 to 99.
		• The **deny** keyword denies access if the conditions are matched.
		• For *source*, enter multicast addresses 224.0.1.39 and 224.0.1.40, which carry Auto-RP information.
		• (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore.
		
		The access list is always terminated by an implicit deny statement for everything.

Step 3	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Specifies the interface to be configured, and enters interface configuration mode.
Step 4	**ip multicast boundary** *access-list-number* **Example:** `Switch(config-if)# ip multicast boundary 12`	Configures the boundary, specifying the access list you created in Step 2.
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.

	Command or Action	Purpose
Step 7	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Example: Defining the IP Multicast Boundary to Deny Auto-RP Information, on page 888

Configuring Candidate BSRs

You can configure one or more candidate BSRs. The devices serving as candidate BSRs should have good connectivity to other devices and be in the backbone portion of the network.

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **ip pim bsr-candidate** *interface-id hash-mask-length* [*priority*]
3. **end**
4. **show running-config**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip pim bsr-candidate** *interface-id hash-mask-length* [*priority*] **Example:** `Switch(config)# ip pim bsr-candidate gigabitethernet 1/0/3 28 100`	Configures your switch to be a candidate BSR. • For *interface-id*, enter the interface on this switch from which the BSR address is derived to make it a candidate. This interface must be enabled with PIM. Valid interfaces include physical ports, port channels, and VLANs. • For *hash-mask-length*, specify the mask length (32 bits maximum) that is to be ANDed with the group address before the hash function is called. All groups with the same seed hash correspond to the same

	Command or Action	Purpose
		RP. For example, if this value is 24, only the first 24 bits of the group addresses matter. • (Optional) For *priority*, enter a number from 0 to 255. The BSR with the larger priority is preferred. If the priority values are the same, the device with the highest IP address is selected as the BSR. The default is 0.
Step 3	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 4	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

PIM v2 BSR, on page 858

Example: Configuring Candidate BSRs, on page 889

Configuring the Candidate RPs

You can configure one or more candidate RPs. Similar to BSRs, the RPs should also have good connectivity to other devices and be in the backbone portion of the network. An RP can serve the entire IP multicast address space or a portion of it. Candidate RPs send candidate RP advertisements to the BSR. When deciding which devices should be RPs, consider these options:

- In a network of Cisco routers and multilayer switches where only Auto-RP is used, any device can be configured as an RP.
- In a network that includes only Cisco PIMv2 routers and multilayer switches and with routers from other vendors, any device can be used as an RP.
- In a network of Cisco PIMv1 routers, Cisco PIMv2 routers, and routers from other vendors, configure only Cisco PIMv2 routers and multilayer switches as RPs.

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **ip pim rp-candidate** *interface-id* [**group-list** *access-list-number*]
3. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip pim rp-candidate** *interface-id* [**group-list** *access-list-number*] **Example:** `Switch(config)# ip pim rp-candidate gigabitethernet 1/0/5 group-list 10`	Configures your switch to be a candidate RP. • For *interface-id*, specify the interface whose associated IP address is advertised as a candidate RP address. Valid interfaces include physical ports, port channels, and VLANs. • (Optional) For **group-list** *access-list-number*, enter an IP standard access list number from 1 to 99. If no group-list is specified, the switch is a candidate RP for all groups.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] **Example:** `Switch(config)# access-list 10 permit 239.0.0.0 0.255.255.255`	Creates a standard access list, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number specified in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, enter the number of the network or host from which the packet is being sent. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. The access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Example: `Switch(config-if)# end`	
Step 5	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file

Related Topics

Example: Configuring Candidate RPs, on page 889

Configuring Auto-RP and BSR for the Network

If there are only Cisco devices in your network (no routers from other vendors), there is no need to configure a BSR. Configure Auto-RP in a network that is running both PIMv1 and PIMv2.

If you have non-Cisco PIMv2 routers that need to interoperate with Cisco PIMv1 routers and multilayer switches, both Auto-RP and a BSR are required. We recommend that a Cisco PIMv2 router or multilayer switch be both the Auto-RP mapping agent and the BSR.

If you must have one or more BSRs, we have these recommendations:

- Configure the candidate BSRs as the RP-mapping agents for Auto-RP. For information about these procedures, see:
  - Configuring a Rendezvous Point, on page 864
  - Configuring Candidate BSRs , on page 878
- For group prefixes advertised through Auto-RP, the PIMv2 BSR mechanism should not advertise a subrange of these group prefixes served by a different set of RPs. In a mixed PIMv1 and PIMv2 domain, backup RPs should serve the same group prefixes. This prevents the PIMv2 DRs from selecting a different RP from those PIMv1 DRs, due to the longest match lookup in the RP-mapping database.

Before You Begin

Beginning in privileged EXEC mode, follow these steps to verify the consistency of group-to-RP mappings. This procedure is optional.

SUMMARY STEPS

1. **show ip pim rp** [*hostname* or *IP address* | **mapping** [*hostname* or *IP address* | **elected** | **in-use**] | **metric** [*hostname* or *IP address*]]
2. **show ip pim rp-hash** *group*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show ip pim rp** [*hostname* or *IP address* | **mapping** [*hostname* or *IP address* | **elected** | **in-use**] | **metric** [*hostname* or *IP address*]] **Example:** `Switch# show ip pim rp mapping`	On any Cisco device, displays available RP mappings and metrics: • (Optional) For the *hostname*, specify the IP name of the group about which to display RPs. • (Optional) For the *IP address*, specify the IP address of the group about which to display RPs. • (Optional) Use the **mapping** keyword to display all group-to-RP mappings of which the Cisco device is aware (either configured or learned from Auto-RP). • (Optional) Use the **metric** keyword to display the RP RPF metric.
Step 2	**show ip pim rp-hash** *group* **Example:** `Switch# show ip pim rp-hash 239.1.1.1`	On a PIMv2 router or multilayer switch, confirms that the same RP is the one that a PIMv1 system chooses. For *group*, enter the group address for which to display RP information.

Delaying the Use of PIM Shortest-Path Tree

The change from shared to source tree happens when the first data packet arrives at the last-hop router. This change occurs because the **ip pim spt-threshold** global configuration command controls that timing.

The shortest-path tree requires more memory than the shared tree but reduces delay. You might want to postpone its use. Instead of allowing the leaf router to immediately move to the shortest-path tree, you can specify that the traffic must first reach a threshold.

You can configure when a PIM leaf router should join the shortest-path tree for a specified group. If a source sends at a rate greater than or equal to the specified kbps rate, the multilayer switch triggers a PIM join message toward the source to construct a source tree (shortest-path tree). If the traffic rate from the source drops below the threshold value, the leaf router switches back to the shared tree and sends a prune message toward the source.

You can specify to which groups the shortest-path tree threshold applies by using a group list (a standard access list). If a value of 0 is specified or if the group list is not used, the threshold applies to all groups.

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
3. **ip pim spt-threshold** {*kbps* | **infinity**} [**group-list** *access-list-number*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] **Example:** `Switch(config)# access-list 16 permit 225.0.0.0 0.255.255.255`	Creates a standard access list. • For *access-list-number*, the range is 1 to 99. • The **deny** keyword denies access if the conditions are matched. • The **permit** keyword permits access if the conditions are matched. • For *source*, specify the multicast group to which the threshold will apply. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. The access list is always terminated by an implicit deny statement for everything.
Step 3	**ip pim spt-threshold** {*kbps* | **infinity**} [**group-list** *access-list-number*] **Example:** `Switch(config)# ip pim spt-threshold infinity group-list 16`	Specifies the threshold that must be reached before moving to shortest-path tree (spt). • For *kbps*, specify the traffic rate in kilobits per second. The default is 0 kbps. **Note** Because of switch hardware limitations, 0 kbps is the only valid entry even though the range is 0 to 4294967. • Specify **infinity** if you want all sources for the specified group to use the shared tree, never switching to the source tree. • (Optional) For **group-list** *access-list-number*, specify the access list created in Step 2. If the value is 0 or if the group list is not used, the threshold applies to all groups.

	Command or Action	Purpose
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Modifying the PIM Router-Query Message Interval

PIM routers and multilayer switches send PIM router-query messages to find which device will be the designated router (DR) for each LAN segment (subnet). The DR is responsible for sending IGMP host-query messages to all hosts on the directly connected LAN.

With PIM DM operation, the DR has meaning only if IGMPv1 is in use. IGMPv1 does not have an IGMP querier election process, so the elected DR functions as the IGMP querier. With PIM-SM operation, the DR is the device that is directly connected to the multicast source. It sends PIM register messages to notify the RP that multicast traffic from a source needs to be forwarded down the shared tree. In this case, the DR is the device with the highest IP address.

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip pim query-interval** *seconds*
4. **end**
5. **show ip igmp interface** [*interface-id*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Specifies the interface to be configured, and enters interface configuration mode.
Step 3	**ip pim query-interval** *seconds* **Example:** `Switch(config-if)# ip pim query-interval 45`	Configures the frequency at which the switch sends PIM router-query messages. The default is 30 seconds. The range is 1 to 65535.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show ip igmp interface** [*interface-id*] **Example:** `Switch# show ip igmp interface`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring PIM

Use the privileged EXEC commands in the following table to monitor your PIM configurations.

Table 105: PIM Monitoring Commands

Command	Purpose
show ip pim all-vrfs tunnel [**tunnel** *tunnel_number* | **verbose**]	Displays all VRFs.
show ip pim autorp	Displays global auto-RP information.
show ip pim boundary	Displays information about mroutes filtered by administratively scoped IPv4 multicast boundaries configured on an interface.
show ip pim interface	Displays information about interfaces configured for Protocol Independent Multicast (PIM).
show ip pim neighbor	Displays the PIM neighbor information.
show ip pim tunnel [**tunnel** | **verbose**]	Displays information about Protocol Independent Multicast (PIM) tunnel interfaces
show ip pim vrf { **word** { **all-vrfs** | **autorp** | **boundary** | **bsr-router** | **interface** | **mdt** | **neighbor** | **rp** | **rp-hash** | **tunnel** } }	Displays the VPN routing/forwarding instance.

Monitoring RP Mapping

Use the privileged EXEC commands in the following table to monitor RP mapping.

Table 106: RP Mapping Monitoring Commands

Command	Purpose
show ip pim bsr	Displays information about the elected BSR.
show ip pim bsr-router	Displays information about the BSRv2.
show ip pim rp [*hostname* or *IP address* | **mapping** [*hostname* or *IP address* | **elected** [*hostname* or *IP address*] | **in-use** [*hostname* or *IP address*]] | **metric** [*hostname* or *IP address*]]	Displays how the switch learns of the RP (through the BSR or the Auto-RP mechanism).
show ip pim rp-hash *hostname* or *IP group address*	Displays the RP that was selected for the specified group.

Troubleshooting PIMv1 and PIMv2 Interoperability Problems

When debugging interoperability problems between PIMv1 and PIMv2, check these in the order shown:

1. Verify RP mapping with the **show ip pim rp-hash** privileged EXEC command, making sure that all systems agree on the same RP for the same group.
2. Verify interoperability between different versions of DRs and RPs. Make sure that the RPs are interacting with the DRs properly (by responding with register-stops and forwarding decapsulated data packets from registers).

Configuration Examples for PIM

Example: Enabling PIM Stub Routing

In this example, IP multicast routing is enabled, Switch A PIM uplink port 25 is configured as a routed uplink port with **spare-dense-mode** enabled. PIM stub routing is enabled on the VLAN 100 interfaces and on Gigabit Ethernet port 20.

```
Switch(config)# ip multicast-routing distributed
Switch(config)# interface GigabitEthernet3/0/25
Switch(config-if)# no switchport
Switch(config-if)# ip address 3.1.1.2 255.255.255.0
Switch(config-if)# ip pim sparse-dense-mode
Switch(config-if)# exit
Switch(config)# interface vlan100
Switch(config-if)# ip pim passive
Switch(config-if)# exit
Switch(config)# interface GigabitEthernet3/0/20
Switch(config-if)# ip pim passive
Switch(config-if)# exit
Switch(config)# interface vlan100
Switch(config-if)# ip address 100.1.1.1 255.255.255.0
Switch(config-if)# ip pim passive
Switch(config-if)# exit
Switch(config)# interface GigabitEthernet3/0/20
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.1.1.1 255.255.255.0
Switch(config-if)# ip pim passive
Switch(config-if)# end
```

Related Topics

Enabling PIM Stub Routing , on page 862

PIM Stub Routing, on page 856

Example: Verifying PIM Stub Routing

To verify that PIM stub is enabled for each interface, use the **show ip pim interface** privileged EXEC command:

```
Switch# show ip pim interface
Address Interface Ver/ Nbr Query DR DR
```

```
Mode Count Intvl Prior
3.1.1.2 GigabitEthernet3/0/25 v2/SD 1 30 1 3.1.1.2
100.1.1.1 Vlan100 v2/P 0 30 1 100.1.1.1
10.1.1.1 GigabitEthernet3/0/20 v2/P 0 30 1 10.1.1.1
```

Example: Manually Assigning an RP to Multicast Groups

This example shows how to configure the address of the RP to 147.106.6.22 for multicast group 225.2.2.2 only:

```
Switch(config)# access-list 1 permit 225.2.2.2 0.0.0.0
Switch(config)# ip pim rp-address 147.106.6.22 1
```

Example: Configuring Auto-RP

This example shows how to send RP announcements out all PIM-enabled interfaces for a maximum of 31 hops. The IP address of port 1 is the RP. Access list 5 describes the group for which this switch serves as RP:

```
Switch(config)# ip pim send-rp-announce gigabitethernet1/0/1 scope 31 group-list 5
Switch(config)# access-list 5 permit 224.0.0.0 15.255.255.255
```

Related Topics

Setting Up Auto-RP in a New Internetwork , on page 867

Auto-RP, on page 857

Example: Defining the IP Multicast Boundary to Deny Auto-RP Information

This example shows a portion of an IP multicast boundary configuration that denies Auto-RP information:

```
Switch(config)# access-list 1 deny 224.0.1.39
Switch(config)# access-list 1 deny 224.0.1.40
Switch(config)# access-list 1 permit all
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip multicast boundary 1
```

Related Topics

Defining the IP Multicast Boundary , on page 876

Example: Filtering Incoming RP Announcement Messages

This example shows a sample configuration on an Auto-RP mapping agent that is used to prevent candidate RP announcements from being accepted from unauthorized candidate RPs:

```
Switch(config)# ip pim rp-announce-filter rp-list 10 group-list 20
Switch(config)# access-list 10 permit host 172.16.5.1
Switch(config)# access-list 10 permit host 172.16.2.1
Switch(config)# access-list 20 deny 239.0.0.0 0.0.255.255
Switch(config)# access-list 20 permit 224.0.0.0 15.255.255.255
```

The mapping agent accepts candidate RP announcements from only two devices, 172.16.5.1 and 172.16.2.1. The mapping agent accepts candidate RP announcements from these two devices only for multicast groups that fall in the group range of 224.0.0.0 to 239.255.255.255. The mapping agent does not accept candidate RP announcements from any other devices in the network. Furthermore, the mapping agent does not accept candidate RP announcements from 172.16.5.1 or 172.16.2.1 if the announcements are for any groups in the 239.0.0.0 through 239.255.255.255 range. This range is the administratively scoped address range.

Related Topics

Filtering Incoming RP Announcement Messages , on page 872

Example: Preventing Join Messages to False RPs

If all interfaces are in sparse mode, use a default-configured RP to support the two well-known groups 224.0.1.39 and 224.0.1.40. Auto-RP uses these two well-known groups to collect and distribute RP-mapping information. When this is the case and the **ip pim accept-rp auto-rp** command is configured, another **ip pim accept-rp** command accepting the RP must be configured as follows:

```
Switch(config)# ip pim accept-rp 172.10.20.1 1
Switch(config)# access-list 1 permit 224.0.1.39
Switch(config)# access-list 1 permit 224.0.1.40
```

Related Topics

Preventing Join Messages to False RPs , on page 872

Example: Configuring Candidate BSRs

This example shows how to configure a candidate BSR, which uses the IP address 172.21.24.18 on a port as the advertised BSR address, uses 30 bits as the hash-mask-length, and has a priority of 10.

```
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# ip address 172.21.24.18 255.255.255.0
Switch(config-if)# ip pim sparse-dense-mode
Switch(config-if)# ip pim bsr-candidate gigabitethernet1/0/2 30 10
```

Related Topics

Configuring Candidate BSRs , on page 878

PIM v2 BSR, on page 858

Example: Configuring Candidate RPs

This example shows how to configure the switch to advertise itself as a candidate RP to the BSR in its PIM domain. Standard access list number 4 specifies the group prefix associated with the RP that has the address identified by a port. That RP is responsible for the groups with the prefix 239.

```
Switch(config)# ip pim rp-candidate gigabitethernet1/0/2 group-list 4
Switch(config)# access-list 4 permit 239.0.0.0 0.255.255.255
```

Related Topics

Configuring the Candidate RPs , on page 879

Where to Go Next for PIM

You can configure the following:

- IGMP
- Wireless Multicast
- SSM
- IP Multicast Routing
- Service Discovery Gateway

Additional References

Related Documents

Related Topic	Document Title
PIM is defined in RFC 4601 and in these Internet Engineering Task Force (IETF) Internet drafts.	• *Protocol Independent Multicast (PIM): Motivation and Architecture* • *Protocol Independent Multicast (PIM), Dense Mode Protocol Specification* • *Protocol Independent Multicast (PIM), Sparse Mode Protocol Specification* • *draft-ietf-idmr-igmp-v2-06.txt, Internet Group Management Protocol, Version 2* • *draft-ietf-pim-v2-dm-03.txt, PIM Version 2 Dense Mode*
For complete syntax and usage information for the commands used in this chapter.	*IP Multicast Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*
IGMP Helper command syntax and usage information.	*IP Multicast Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*

Related Topic	Document Title
Multicast Source Discovery Protocol (MSDP)	*IP Routing: Protocol-Independent Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*
Enhanced Interior Gateway Routing Protocol (EIGRP) stub routing	*IP Routing: EIGRP Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*
Open Shortest Path First (OSPF) stub routing	*IP Routing: OSPF Configuration Guide, Cisco IOS XE 3SE (Catalyst 3850 Switches)*
Platform-independent configuration information	• *IP Multicast: PIM Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)* • *IP Multicast: IGMP Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)* • *IP Multicast: Multicast Optimization Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*

Standards and RFCs

Standard/RFC	Title
RFC 4601	*Protocol-Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification*

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for PIM

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 41

Configuring SSM

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for Configuring SSM

The following are the prerequisites for configuring source-specific multicast (SSM) and SSM mapping:

- Before you configure SSM mapping, you must perform the following tasks:
  - Enable IP multicast routing. For information about this procedure, see Configuring Basic IP Multicast Routing , on page 914.

- Enable PIM sparse mode. For information about this procedure, see How to Configure PIM, on page 862.
- Configure SSM. For information about this procedure, see Configuring SSM , on page 899.

- Before you configure static SSM mapping, you must configure access control lists (ACLs) that define the group ranges to be mapped to source addresses.
- Before you can configure and use SSM mapping with DNS look ups, you must be able to add records to a running DNS server. If you do not already have a DNS server running, you need to install one.

Note You can use a product such as *Cisco Network Registrar* to add records to a running DNS server.

Restrictions for Configuring SSM

The following are the restrictions for configuring SSM:

- To run SSM with IGMPv3, SSM must be supported in the Cisco IOS router, the host where the application is running, and the application itself.
- The SSM mapping feature does not have all the benefits of full SSM. Because SSM mapping takes a group join from a host and identifies this group with an application associated with one or more sources, it can only support one such application per group. Full SSM applications can still share the same group as in SSM mapping.
- Enable IGMPv3 carefully on the last hop router when you rely solely on SSM mapping as a transition solution for full SSM. When you enable both SSM mapping and IGMPv3 and the hosts already support IGMPv3 (but not SSM), the hosts send IGMPv3 group reports. SSM mapping does not support these IGMPv3 group reports, and the router does not correctly associate sources with these reports.
- Existing applications in a network predating SSM do not work within the SSM range unless they are modified to support (S, G) channel subscriptions. Therefore, enabling SSM in a network can cause problems for existing applications if they use addresses within the designated SSM range.
- IGMPv3 uses new membership report messages that might not be correctly recognized by older IGMP snooping switches.
- Address management is still necessary to some degree when SSM is used with Layer 2 switching mechanisms. Cisco Group Management Protocol (CGMP), IGMP snooping, or Router-Port Group Management Protocol (RGMP) support only group-specific filtering, not (S, G) channel-specific filtering. If different receivers in a switched network request different (S, G) channels sharing the same group, they do not benefit from these existing mechanisms. Instead, both receivers receive all (S, G) channel traffic and filter out the unwanted traffic on input.

  Because SSM can re-use the group addresses in the SSM range for many independent applications, this situation can lead to decreased traffic filtering in a switched network. For this reason, it is important to use random IP addresses from the SSM range for an application to minimize the chance for re-use of a single address within the SSM range between different applications. For example, an application service providing a set of television channels should, even with SSM, use a different group for each television (S, G) channel. This setup guarantees that multiple receivers to different channels within the same application service never experience traffic aliasing in networks that include Layer 2 switches.

- In PIM-SSM, the last hop router continues to periodically send (S, G) join messages if appropriate (S, G) subscriptions are on the interfaces. Therefore, as long as receivers send (S, G) subscriptions, the shortest path tree (SPT) state from the receivers to the source is maintained, even if the source does not send traffic for longer periods of time (or even never).

  The opposite situation occurs with PIM-SM, where (S, G) state is maintained only if the source is sending traffic and receivers are joining the group. If a source stops sending traffic for more than 3 minutes in PIM-SM, the (S, G) state is deleted and only reestablished after packets from the source arrive again through the RPT (rendezvous point tree). Because no mechanism in PIM-SSM notifies a receiver that a source is active, the network must maintain the (S, G) state in PIM-SSM as long as receivers are requesting receipt of that channel.

Information About SSM

The source-specific multicast (SSM) feature is an extension of IP multicast in which datagram traffic is forwarded to receivers from only those multicast sources that the receivers have explicitly joined. For multicast groups configured for SSM, only SSM distribution trees (no shared trees) are created.

This section describes how to configure source-specific multicast (SSM). For a complete description of the SSM commands in this section, refer to the *IP Multicast Command Reference*. To locate documentation for other commands that appear in this chapter, use the command reference master index, or search online.

SSM Components Overview

SSM is a datagram delivery model that best supports one-to-many applications, also known as broadcast applications. SSM is a core networking technology for the Cisco implementation of IP multicast solutions targeted for audio and video broadcast application environments. The switch supports the following components that support SSM implementation:

- Protocol independent multicast source-specific mode (PIM-SSM)

  PIM-SSM is the routing protocol that supports the implementation of SSM and is derived from PIM sparse mode (PIM-SM).

- Internet Group Management Protocol version 3 (IGMPv3)

SSM and Internet Standard Multicast (ISM)

The current IP multicast infrastructure in the Internet and many enterprise intranets is based on the PIM-SM protocol and Multicast Source Discovery Protocol (MSDP). These protocols have the limitations of the Internet Standard Multicast (ISM) service model. For example, with ISM, the network must maintain knowledge about which hosts in the network are actively sending multicast traffic.

The ISM service consists of the delivery of IP datagrams from any source to a group of receivers called the multicast host group. The datagram traffic for the multicast host group consists of datagrams with an arbitrary IP unicast source address (S) and the multicast group address (G) as the IP destination address. Systems receive this traffic by becoming members of the host group. Membership in a host group simply requires signaling the host group through IGMP version 1, 2, or 3.

In SSM, delivery of datagrams is based on (S, G) channels. In both SSM and ISM, no signaling is required to become a source. However, in SSM, receivers must subscribe or unsubscribe to (S, G) channels to receive or not receive traffic from specific sources. In other words, receivers can receive traffic only from (S, G) channels to which they are subscribed, whereas in ISM, receivers need not know the IP addresses of sources from which they receive their traffic. The proposed standard approach for channel subscription signaling uses IGMP and includes modes membership reports, which are supported only in IGMP version 3.

SSM IP Address Range

SSM can coexist with the ISM service by applying the SSM delivery model to a configured subset of the IP multicast group address range. Cisco IOS software allows SSM configuration for the IP multicast address range of 224.0.0.0 through 239.255.255.255. When an SSM range is defined, existing IP multicast receiver applications do not receive any traffic when they try to use an address in the SSM range (unless the application is modified to use an explicit (S, G) channel subscription).

SSM Operations

An established network, in which IP multicast service is based on PIM-SM, can support SSM services. SSM can also be deployed alone in a network without the full range of protocols required for interdomain PIM-SM (for example, MSDP, Auto-RP, or bootstrap router [BSR]) if only SSM service is needed.

If SSM is deployed in a network already configured for PIM-SM, only the last-hop routers support SSM. Routers that are not directly connected to receivers do not require support for SSM. In general, these not-last-hop routers must only run PIM-SM in the SSM range and might need additional access control configuration to suppress MSDP signalling, registering, or PIM-SM shared tree operations from occurring within the SSM range.

Use the **ip pim ssm** global configuration command to configure the SSM range and to enable SSM. This configuration has the following effects:

- For groups within the SSM range, (S, G) channel subscriptions are accepted through IGMPv3 include-mode membership reports.
- PIM operations within the SSM range of addresses change to PIM-SSM, a mode derived from PIM-SM. In this mode, only PIM (S, G) join and prune messages are generated by the router, and no (S, G) rendezvous point tree (RPT) or (*, G) RPT messages are generated. Incoming messages related to RPT operations are ignored or rejected, and incoming PIM register messages are immediately answered with register-stop messages. PIM-SSM is backward-compatible with PIM-SM unless a router is a last-hop router. Therefore, routers that are not last-hop routers can run PIM-SM for SSM groups (for example, if they do not yet support SSM).
- No MSDP source-active (SA) messages within the SSM range are accepted, generated, or forwarded.

SSM Mapping

In a typical set-top box (STB) deployment, each TV channel uses one separate IP multicast group and has one active server host sending the TV channel. A single server can send multiple TV channels, but each to a different group. In this network environment, if a router receives an IGMPv1 or IGMPv2 membership report for a particular group, the report addresses the well-known TV server for the TV channel associated with the multicast group.

When SSM mapping is configured, if a router receives an IGMPv1 or IGMPv2 membership report for a particular group, the router translates this report into one or more channel memberships for the well-known sources associated with this group.

When the router receives an IGMPv1 or IGMPv2 membership report for a group, the router uses SSM mapping to determine one or more source IP addresses for the group. SSM mapping then translates the membership report as an IGMPv3 report and continues as if it had received an IGMPv3 report. The router then sends PIM joins and continues to be joined to these groups as long as it continues to receive the IGMPv1 or IGMPv2 membership reports, and the SSM mapping for the group remains the same.

SSM mapping enables the last hop router to determine the source addresses either by a statically configured table on the router or through a DNS server. When the statically configured table or the DNS mapping changes, the router leaves the current sources associated with the joined groups.

Static SSM Mapping

With static SSM mapping, you can configure the last hop router to use a static map to determine the sources that are sending to groups. Static SSM mapping requires that you configure ACLs to define group ranges. After configuring the ACLs to define group ranges, you can then map the groups permitted by those ACLs to sources by using the **ip igmp ssm-map static** global configuration command.

You can configure static SSM mapping in smaller networks when a DNS is not needed or to locally override DNS mappings. When configured, static SSM mappings take precedence over DNS mappings.

Related Topics

Configuring Static SSM Mapping , on page 900

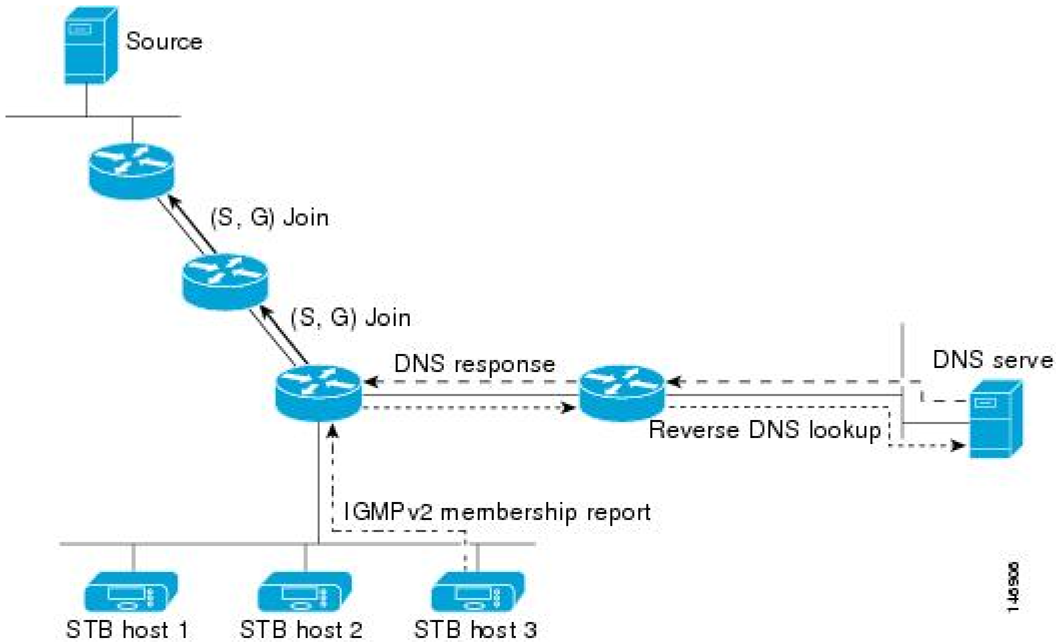
Configuring Static Traffic Forwarding with SSM Mapping , on page 903

DNS-Based SSM Mapping

You can use DNS-based SSM mapping to configure the last hop router to perform a reverse DNS lookup to determine sources sending to groups. When DNS-based SSM mapping is configured, the router constructs a domain name that includes the group address and performs a reverse lookup into the DNS. The router looks up IP address resource records and uses them as the source addresses associated with this group. SSM mapping supports up to 20 sources for each group. The router joins all sources configured for a group.

The following figure displays DNS-based SSM mapping.

Figure 31: DNS-Based SSM Mapping



The SSM mapping mechanism that enables the last hop router to join multiple sources for a group can provide source redundancy for a TV broadcast. In this context, the last hop router provides redundancy using SSM mapping to simultaneously join two video sources for the same TV channel. However, to prevent the last hop router from duplicating the video traffic, the video sources must use a server-side switchover mechanism. One video source is active, and the other backup video source is passive. The passive source waits until an active source failure is detected before sending the video traffic for the TV channel. Thus, the server-side switchover mechanism ensures that only one of the servers is actively sending video traffic for the TV channel.

To look up one or more source addresses for a group that includes G1, G2, G3, and G4, you must configure these DNS records on the DNS server:

```
G4.G3.G2.G1 [multicast-domain] [timeout] IN A source-address-1
 IN A source-address-2
 IN A source-address-n
```

See your DNS server documentation for more information about configuring DNS resource records.

Related Topics

Configuring DNS-Based SSM Mapping , on page 901

How to Configure SSM

For a complete description of the source-specific multicast (SSM) commands in this section, see the *IP Multicast Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*. To locate documentation for other commands that appear in this chapter, use the command reference master index, or search online.

Configuring SSM

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **ip pim ssm** [**default** | **range** *access-list*]
3. **interface** *type number*
4. **ip pim** {**sparse-mode** | **sparse-dense-mode**}
5. **ip igmp version 3**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip pim ssm** [**default** | **range** *access-list*] **Example:** `Switch(config)# ip pim ssm range 20`	Defines the SSM range of IP multicast addresses.
Step 3	**interface** *type number* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Selects an interface that is connected to hosts on which IGMPv3 can be enabled, and enters the interface configuration mode.
Step 4	**ip pim** {**sparse-mode** | **sparse-dense-mode**} **Example:** `Switch(config-if)# ip pim sparse-dense-mode`	Enables PIM on an interface. You must use either sparse mode or sparse-dense mode.
Step 5	**ip igmp version 3** **Example:** `Switch(config-if)# ip igmp version 3`	Enables IGMPv3 on this interface. The default version of IGMP is set to Version 2.

Configuring Source Specific Multicast Mapping

The Source Specific Multicast (SSM) mapping feature supports SSM transition when supporting SSM on the end system is impossible or unwanted due to administrative or technical reasons. You can use SSM mapping to leverage SSM for video delivery to legacy STBs that do not support IGMPv3 or for applications that do not use the IGMPv3 host stack.

Configuring Static SSM Mapping

The following procedure describes how to configure static SSM mapping.

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp ssm-map enable**
3. **no ip igmp ssm-map query dns**
4. **ip igmp ssm-map static** *access-list source-address*
5. Repeat Step 4 to configure additional static SSM mappings, if required.
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp ssm-map enable** **Example:** `Switch(config)# ip igmp ssm-map enable`	Enables SSM mapping for groups in the configured SSM range. **Note** By default, this command enables DNS-based SSM mapping.
Step 3	**no ip igmp ssm-map query dns** **Example:** `Switch(config)# no ip igmp ssm-map dns`	(Optional) Disables DNS-based SSM mapping. **Note** Disable DNS-based SSM mapping if you only want to rely on static SSM mapping. By default, the **ip igmp ssm-map** global configuration command enables DNS-based SSM mapping.
Step 4	**ip igmp ssm-map static** *access-list source-address*	Configures static SSM mapping. The ACL supplied for *access-list* defines the groups to be mapped to the source IP address entered for the *source-address*.

	Command or Action	Purpose
	Example: `Switch(config)# ip igmp ssm-map static 11 172.16.8.11`	**Note** You can configure additional static SSM mappings. If additional SSM mappings are configured and the router receives an IGMPv1 or IGMPv2 membership report for a group in the SSM range, the switch determines the source addresses associated with the group by using each configured **ip igmp ssm-map static** command. The switch associates up to 20 sources per group.
Step 5	Repeat Step 4 to configure additional static SSM mappings, if required.	—
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 7	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 8	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Static SSM Mapping, on page 897

Configuring DNS-Based SSM Mapping

To configure DNS-based SSM mapping, you need to create a DNS server zone or add records to an existing zone. If the routers that are using DNS-based SSM mapping are also using DNS for other purposes, you should use a normally configured DNS server. If DNS-based SSM mapping is the only DNS implementation being used on the router, you can configure a false DNS setup with an empty root zone or a root zone that points back to itself.

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp ssm-map enable**
3. **ip igmp ssm-map query dns**
4. **ip domain multicast** *domain-prefix*
5. **ip name-server** *server-address1* [*server-address2... server-address6*]
6. Repeat Step 5 to configure additional DNS servers for redundancy, if required.
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp ssm-map enable** **Example:** `Switch(config)# ip igmp ssm-map enable`	Enables SSM mapping for groups in a configured SSM range.
Step 3	**ip igmp ssm-map query dns** **Example:** `Switch(config)# ip igmp ssm-map query dns`	(Optional) Enables DNS-based SSM mapping. By default, the **ip igmp ssm-map** command enables DNS-based SSM mapping. Only the **no** form of this command is saved to the running configuration. **Note** Use this command to reenable DNS-based SSM mapping if DNS-based SSM mapping is disabled.
Step 4	**ip domain multicast** *domain-prefix* **Example:** `Switch(config)# ip domain multicast ssm-map.cisco.com`	(Optional) Changes the domain prefix used by the switch for DNS-based SSM mapping. By default, the switch uses the *ip-addr.arpa* domain prefix.
Step 5	**ip name-server** *server-address1* [*server-address2... server-address6*] **Example:** `Switch(config)# ip name-server`	Specifies the address of one or more name servers to use for name and address resolution.

	Command or Action	Purpose
	`172.16.1.111 172.16.1.2`	
Step 6	Repeat Step 5 to configure additional DNS servers for redundancy, if required.	—
Step 7	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 8	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 9	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

DNS-Based SSM Mapping, on page 897

Configuring Static Traffic Forwarding with SSM Mapping

Use static traffic forwarding with SSM mapping to statically forward SSM traffic for certain groups.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type number*
3. **ip igmp static-group** *group-address* **source ssm-map**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *type number* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Selects an interface on which to statically forward traffic for a multicast group using SSM mapping, and enters interface configuration mode. **Note** Static forwarding of traffic with SSM mapping works with either DNS-based SSM mapping or statically configured SSM mapping.
Step 3	**ip igmp static-group** *group-address* **source ssm-map** **Example:** `Switch(config-if)# ip igmp static-group 239.1.2.1 source ssm-map`	Configures SSM mapping to statically forward a (S, G) channel from the interface. Use this command if you want to statically forward SSM traffic for certain groups. Use DNS-based SSM mapping to determine the source addresses of the channels.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Static SSM Mapping, on page 897

Monitoring SSM

Use the privileged EXEC commands in the following table to monitor SSM.

Table 107: Commands for Monitoring SSM

Command	Purpose
show ip igmp groups detail	Displays the (S, G) channel subscription through IGMPv3.
show ip mroute	Displays whether a multicast group supports SSM service or whether a source-specific host report was received.

Monitoring SSM Mapping

Use the privileged EXEC commands in the following table to monitor SSM mapping.

Table 108: SSM Mapping Monitoring Commands

Command	Purpose
show ip igmp ssm-mapping	Displays information about SSM mapping.
show ip igmp ssm-mapping *group-address*	Displays the sources that SSM mapping uses for a particular group.
show ip igmp groups [*group-name* | *group-address* | *interface-type interface-number*] [**detail**]	Displays the multicast groups with receivers that are directly connected to the router and that were learned through IGMP.
show host	Displays the default domain name, the style of name lookup service, a list of name server hosts, and the cached list of hostnames and addresses.
debug ip igmp *group-address*	Displays the IGMP packets received and sent and IGMP host-related events.

Where to Go Next for SSM

You can configure the following:

- IGMP
- Wireless Multicast

- PIM
- IP Multicast Routing
- Service Discovery Gateway

Additional References

Related Documents

Related Topic	Document Title
SSM and other available commands	*IP Multicast Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*
Platform-independent configuration information	• *IP Multicast: PIM Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)* • *IP Multicast: IGMP Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)* • *IP Multicast: Multicast Optimization Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*

Standards and RFCs

Standard/RFC	Title
RFC 4601	*Protocol-Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification*

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for SSM

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 42

Configuring IP Multicast Routing

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for Configuring IP Multicast Routing

The following are the prerequisites for configuring IP multicast routing:

- To use the IP multicast routing feature on the switch, the switch or active switch must be running the IP Services feature set.

- You must enable IP multicast routing and configure the PIM version and PIM mode on the switch. After performing these tasks, the switch can then forward multicast packets and can populate its multicast routing table.
- To participate in IP multicasting, the multicast hosts, routers, and multilayer switch must have IGMP operating.

Restrictions for Configuring IP Multicast Routing

The following are the restrictions for configuring IP multicast routing:

- IP multicast routing is not supported on switches running the LAN Base feature set.
- Layer 3 IPv6 multicast routing is not supported on the switch.
- High-availability support for Layer 3 multicast routing is not supported.
- You cannot have a switch stack containing a mix of Catalyst 3850 and Catalyst 3650 switches.

Information About IP Multicast Routing

IP multicasting is an efficient way to use network resources, especially for bandwidth-intensive services such as audio and video. IP multicast routing enables a host (source) to send packets to a group of hosts (receivers) anywhere within the IP network by using a special form of IP address called the IP multicast group address.

The sending host inserts the multicast group address into the IP destination address field of the packet, and IP multicast routers and multilayer switches forward incoming IP multicast packets out all interfaces that lead to members of the multicast group. Any host, regardless of whether it is a member of a group, can send to a group. However, only the members of a group receive the message.

Note For complete syntax and usage information for the commands used in this chapter, see the *IP Multicast Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*. For information on configuring the Multicast Source Discovery Protocol (MSDP), see the *Catalyst 3850 Routing Configuration Guide*.

Cisco's Implementation of IP Multicast Routing

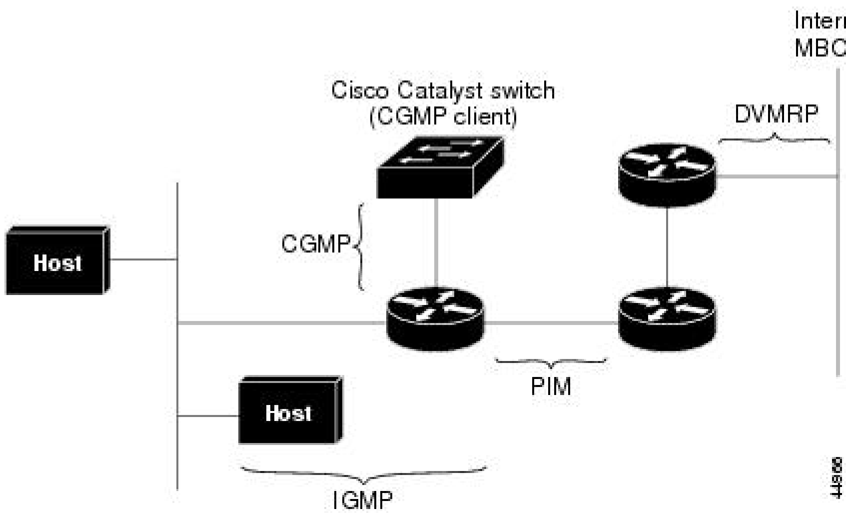
Cisco IOS software supports the following protocols to implement IP multicast routing:

- Internet Group Management Protocol (IGMP) is used among hosts on a LAN and the routers (and multilayer switches) on that LAN to track the multicast groups of which hosts are members.
- Protocol-Independent Multicast (PIM) protocol is used among routers and multilayer switches to track which multicast packets to forward to each other and to their directly connected LANs.

Note The switch does not support the Distance Vector Multicast Routing Protocol (DVMRP) nor the Cisco Group Management Protocol (CGMP).

The following figure shows where the Cisco-supported protocols for the switch operate within the IP multicast environment.

Figure 32: IP Multicast Routing Protocols



According to IPv4 multicast standards, the MAC destination multicast address begins with 0100:5e and is appended by the last 23 bits of the IP address. For example, if the IP destination address is 239.1.1.39, the MAC destination address is 0100:5e01:0127.

A multicast packet is unmatched when the destination IPv4 address does not match the destination MAC address. The switch forwards the unmatched packet in hardware based upon the MAC address table. If the destination MAC address is not in the MAC address table, the switch floods the packet to the all port in the same VLAN as the receiving port.

Related Topics

Configuring Basic IP Multicast Routing , on page 914

Multicast Forwarding Information Base Overview

The switch uses the Multicast Forwarding Information Base (MFIB) architecture and the Multicast Routing Information Base (MRIB) for IP multicast.

The MFIB architecture provides both modularity and separation between the multicast control plane (Protocol Independent Multicast [PIM] and Internet Group Management Protocol [IGMP]) and the multicast forwarding plane (MFIB). This architecture is used in Cisco IOS IPv6 multicast implementations.

MFIB itself is a multicast routing protocol independent forwarding engine; that is, it does not depend on PIM or any other multicast routing protocol. It is responsible for:

- Forwarding multicast packets
- Registering with the MRIB to learn the entry and interface flags set by the control plane
- Handling data-driven events that must be sent to the control plane
- Maintaining counts, rates, and bytes of received, dropped, and forwarded multicast packets

The MRIB is the communication channel between MRIB clients. Examples of MRIB clients are PIM, IGMP, the multicast routing (mroute) table, and the MFIB.

Related Topics

Configuring IP Multicast Forwarding , on page 916

Multicast Group Concept

Multicast is based on the concept of a group. An arbitrary group of receivers expresses an interest in receiving a particular data stream. This group does not have any physical or geographical boundaries. The hosts can be located anywhere on the Internet. Hosts that are interested in receiving data flowing to a particular group must join the group using IGMP. Hosts must be a member of the group to receive the data stream.

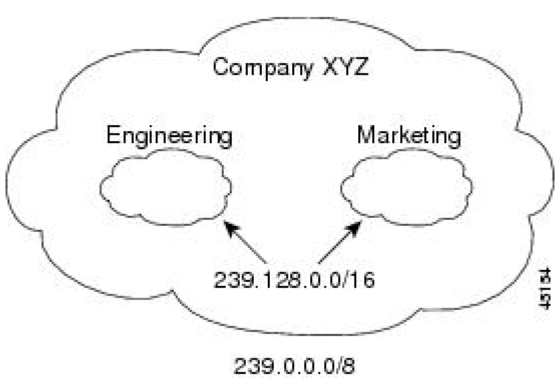
Multicast Boundaries

Administratively-scoped boundaries can be used to limit the forwarding of multicast traffic outside of a domain or subdomain. This approach uses a special range of multicast addresses, called administratively-scoped addresses, as the boundary mechanism. If you configure an administratively-scoped boundary on a routed interface, multicast traffic whose multicast group addresses fall in this range cannot enter or exit this interface, which provides a firewall for multicast traffic in this address range.

Note Multicast boundaries and TTL thresholds control the scoping of multicast domains; however, TTL thresholds are not supported by the switch. You should use multicast boundaries instead of TTL thresholds to limit the forwarding of multicast traffic outside of a domain or a subdomain.

The following figure shows that Company XYZ has an administratively-scoped boundary set for the multicast address range 239.0.0.0/8 on all routed interfaces at the perimeter of its network. This boundary prevents any multicast traffic in the range 239.0.0.0 through 239.255.255.255 from entering or leaving the network. Similarly, the engineering and marketing departments have an administratively-scoped boundary of 239.128.0.0/16 around the perimeter of their networks. This boundary prevents multicast traffic in the range of 239.128.0.0 through 239.128.255.255 from entering or leaving their respective networks.

Figure 33: Administratively-Scoped Boundaries



You can define an administratively-scoped boundary on a routed interface for multicast group addresses. A standard access list defines the range of addresses affected. When a boundary is defined, no multicast data packets are allowed to flow across the boundary from either direction. The boundary allows the same multicast group address to be reused in different administrative domains.

The IANA has designated the multicast address range 239.0.0.0 to 239.255.255.255 as the administratively-scoped addresses. This range of addresses can then be reused in domains administered by different organizations. The addresses would be considered local, not globally unique.

Related Topics

Configuring an IP Multicast Boundary , on page 922

Example: Configuring an IP Multicast Boundary, on page 928

Multicast Routing and Switch Stacks

For all multicast routing protocols, the entire stack appears as a single router to the network and operates as a single multicast router.

In a switch stack, the active switch performs these functions:

- It is responsible for completing the IP multicast routing functions of the stack. It fully initializes and runs the IP multicast routing protocols.
- It builds and maintains the multicast routing table for the entire stack.
- It is responsible for distributing the multicast routing table to all stack members.

The stack members perform these functions:

- They act as multicast routing standby devices and are ready to take over if there is a active switch failure.

  If the active switch fails, all stack members delete their multicast routing tables. The newly elected active switch starts building the routing tables and distributes them to the stack members.
- They do not build multicast routing tables. Instead, they use the multicast routing table that is distributed by the active switch.

Default Multicast Routing Configuration

This table describes the default multicast routing configuration for the switch.

Table 109: Default Multicast Routing Configuration

Feature	Default Setting
Multicast routing	Disabled on all interfaces.

How to Configure Basic IP Multicast Routing

Configuring Basic IP Multicast Routing

You must enable IP multicast routing and configure the PIM version and mode. After performing these tasks, the software can then forward multicast packets, and the switch can populate its multicast routing table.

You can configure an interface to be in PIM dense mode, sparse mode, or sparse-dense mode. The switch populates its multicast routing table and forwards multicast packets it receives from its directly connected LANs according to the mode setting. You must enable PIM in one of these modes for an interface to perform IP multicast routing.

Enabling PIM on an interface also enables IGMP operation on that interface.

Note If you enable PIM on multiple interfaces, when most of these interfaces are not on the outgoing interface list, and IGMP snooping is disabled, the outgoing interface might not be able to sustain line rate for multicast traffic because of the extra replication.

In populating the multicast routing table, dense-mode interfaces are always added to the table. Sparse-mode interfaces are added to the table only when periodic join messages are received from downstream devices or when there is a directly connected member on the interface.

When forwarding from a LAN, sparse-mode operation occurs if there is a rendezvous point (RP) known for the group. An RP acts as the meeting place for sources and receivers of multicast data. If an RP exists, the packets are encapsulated and sent toward the RP. When no RP is known, the packet is flooded in a dense-mode fashion. If the multicast traffic from a specific source is sufficient, the receiver's first-hop router might send join messages toward the source to build a source-based distribution tree.

By default, multicast routing is disabled, and there is no default mode setting.

This procedure is required.

SUMMARY STEPS

1. **configure terminal**
2. **ip multicast-routing**
3. **interface** *interface-id*
4. **ip pim** {**dense-mode** | **sparse-mode** | **sparse-dense-mode**}
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip multicast-routing** **Example:** `Switch(config)# ip multicast-routing`	Enables IP multicast routing. IP multicast routing is supported with Multicast Forwarding Information Base (MFIB) and Multicast Routing Information Base (MRIB).
Step 3	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Specifies the Layer 3 interface on which you want to enable multicast routing, and enters interface configuration mode. The specified interface must be one of the following: • A routed port—A physical port that has been configured as a Layer 3 port by entering the **no switchport** interface configuration command. You will also need to enable IP PIM sparse-dense-mode on the interface, and join the interface as a statically connected member to an IGMP static group. For a configuration example, see Example: Interface Configuration as a Routed Port, on page 834 • An SVI—A VLAN interface created by using the **interface vlan** *vlan-id* global configuration command. You will also need to enable IP PIM sparse-dense-mode on the VLAN, join the VLAN as a statically connected member to an IGMP static group, and then enable IGMP snooping on the VLAN, the IGMP static group, and physical interface. For a configuration example, see Example: Interface Configuration as an SVI, on page 835 These interfaces must have IP addresses assigned to them.
Step 4	**ip pim** {**dense-mode** | **sparse-mode** | **sparse-dense-mode**} **Example:** `Switch(config-if)# ip pim sparse-dense-mode`	Enables a PIM mode on the interface. By default, no mode is configured. The keywords have these meanings: • **dense-mode**—Enables dense mode of operation. • **sparse-mode**—Enables sparse mode of operation. If you configure sparse mode, you must also configure an RP. • **sparse-dense-mode**—Causes the interface to be treated in the mode in which the group belongs. Sparse-dense mode is the recommended setting. • **state-refresh**—PM dense mode state-refresh configuration.

	Command or Action	Purpose
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Cisco's Implementation of IP Multicast Routing, on page 910

Configuring IP Multicast Forwarding

You can use the following procedure to configure IPv4 Multicast Forwarding Information Base (MFIB) interrupt-level IP multicast forwarding of incoming packets or outgoing packets on the switch.

Note After you have enabled IP multicast routing by using the **ip multicast-routing** command, IPv4 multicast forwarding is enabled. Because IPv4 multicast forwarding is enabled by default, you can use the **no** form of the **ip mfib** command to disable IPv4 multicast forwarding.

SUMMARY STEPS

1. **configure terminal**
2. **ip mfib**
3. **exit**
4. **show running-config**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip mfib** **Example:** `Switch(config)# ip mfib`	Enables IP multicast forwarding.
Step 3	**exit** **Example:** `Switch(config)# exit`	Returns to privileged EXEC mode.
Step 4	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Multicast Forwarding Information Base Overview , on page 911

Configuring a Static Multicast Route (mroute)

You can use the following procedure to configure static mroutes. Static mroutes are similar to unicast static routes but differ in the following ways:

- Static mroutes are used to calculate RPF information, not to forward traffic.
- Static mroutes cannot be redistributed.

Static mroutes are strictly local to the switch on which they are defined. Because Protocol Independent Multicast (PIM) does not have its own routing protocol, there is no mechanism to distribute static mroutes throughout the network. Consequently, the administration of static mroutes tends to be more complicated than the administration of unicast static routes.

When static mroutes are configured, they are stored on the switch in a separate table referred to as the static mroute table. When configured, the **ip mroute** command enters a static mroute into the static mroute table for the source address or source address range specified for the source-address and mask arguments. Sources that match the source address or that fall in the source address range specified for the source-address argument will RPF to either the interface associated with the IP address specified for the *rpf-address* argument or the local interface on the switch specified for the *interface-type* and *interface-number* arguments. If an IP address is specified for the *rpf-address* argument, a recursive lookup is done from the unicast routing table on this address to find the directly connected neighbor.

If there are multiple static mroutes configured, the switch performs a longest-match lookup of the mroute table. When the mroute with the longest match (of the source-address) is found, the search terminates and the information in the matching static mroute is used. The order in which the static mroutes are configured is not important.

The administrative distance of an mroute may be specified for the optional distance argument. If a value is not specified for the distance argument, the distance of the mroute defaults to zero. If the static mroute has the same distance as another RPF source, the static mroute will take precedence. There are only two exceptions to this rule: directly connected routes and the default unicast route.

SUMMARY STEPS

1. **configure terminal**
2. **ip mroute** [**vrf** *vrf-name*] *source-address mask* { **fallback-lookup** {**global** | **vrf** *vrf-name* }[*protocol*] {*rpf-address* | *interface-type interface-number*}} [**distance**]
3. **exit**
4. **show running-config**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip mroute** [**vrf** *vrf-name*] *source-address mask* { **fallback-lookup** {**global** | **vrf** *vrf-name* }[*protocol*] {*rpf-address* | *interface-type interface-number*}} [**distance**] **Example:** `Switch(configure)# ip mroute 10.1.1.1 255.255.255.255 10.2.2.2`	The source IP address 10.1.1.1 is configured to be reachable through the interface associated with IP address 10.2.2.2.

	Command or Action	Purpose
Step 3	**exit** **Example:** `Switch(config)# exit`	Returns to privileged EXEC mode.
Step 4	**show running-config** **Example:** `Switch# show running-config`	(Optional) Verifies your entries.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring sdr Listener Support

The MBONE is the small subset of Internet routers and hosts that are interconnected and capable of forwarding IP multicast traffic. Other multimedia content is often broadcast over the MBONE. Before you can join a multimedia session, you need to know what multicast group address and port are being used for the session, when the session is going to be active, and what sort of applications (audio, video, and so forth) are required on your workstation. The MBONE Session Directory Version 2 (sdr) tool provides this information. This freeware application can be downloaded from several sites on the World Wide Web, one of which is http://www.video.ja.net/mice/index.html.

SDR is a multicast application that listens to a well-known multicast group address and port for Session Announcement Protocol (SAP) multicast packets from SAP clients, which announce their conference sessions. These SAP packets contain a session description, the time the session is active, its IP multicast group addresses, media format, contact person, and other information about the advertised multimedia session. The information in the SAP packet is displayed in the SDR Session Announcement window.

Enabling sdr Listener Support

By default, the switch does not listen to session directory advertisements.

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip sap listen**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Specifies the interface to be enabled for sdr, and enters interface configuration mode.
Step 3	**ip sap listen** **Example:** `Switch(config-if)# ip sap listen`	Enables the switch software to listen to session directory announcements.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Limiting How Long an sdr Cache Entry Exists

By default, entries are never deleted from the sdr cache. You can limit how long the entry remains active so that if a source stops advertising SAP information, old advertisements are not unnecessarily kept.

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **ip sap cache-timeout** *minutes*
3. **end**
4. **show running-config**
5. **show ip sap**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip sap cache-timeout** *minutes* **Example:** `Switch(config)# ip sap cache-timeout 30`	Limits how long a Session Announcement Protocol (SAP) cache entry stays active in the cache. By default, entries are never deleted from the cache. For *minutes*, the range is 1 to 1440 minutes (24 hours).
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.

	Command or Action	Purpose
Step 5	**show ip sap** **Example:** `Switch# show ip sap`	Displays the SAP cache.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring an IP Multicast Boundary

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **access-list** {*access-list-number 1-99* | *access-list-number 100-199* | *access-list-number 1300-1999* |*access-list-number 2000-2699* | **dynamic-extended** | **rate-limit**}
3. **interface** *interface-id*
4. **ip multicast boundary** *access-list-number*
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**access-list** {*access-list-number 1-99* | *access-list-number 100-199* | *access-list-number 1300-1999* |*access-list-number 2000-2699* | **dynamic-extended** | **rate-limit**}	Creates a standard access list, repeating the command as many times as necessary. • For *access-list-number*, the ranges are as follows: ◦ access-list-number 1—99 (IP standard access list) ◦ access-list-number 100—199 (IP extended access list)

	Command or Action	Purpose
	Example: `Switch(config)# access-list 99 permit any`	◦ access-list-number 1300—1999 (IP standard access list - expanded range) ◦ access-list-number 2000—2699 (IP extended access list - expanded range) • The **dynamic-extended** keyword extends the dynamic ACL absolute timer. • The **rate-limit** keyword permits a simple rate-limit specific access list. The access list is always terminated by an implicit deny statement for everything.
Step 3	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitEthernet1/0/1`	Specifies the interface to be configured, and enters interface configuration mode. The specified interface must be one of the following: • A routed port—A physical port that has been configured as a Layer 3 port by entering the **no switchport** interface configuration command. You will also need to enable IP PIM sparse-dense-mode on the interface, and join the interface as a statically connected member to an IGMP static group. For a configuration example, see Example: Interface Configuration as a Routed Port, on page 834 • An SVI—A VLAN interface created by using the **interface vlan** *vlan-id* global configuration command. You will also need to enable IP PIM sparse-dense-mode on the VLAN, join the VLAN as a statically connected member to an IGMP static group, and then enable IGMP snooping on the VLAN, the IGMP static group, and physical interface. For a configuration example, see Example: Interface Configuration as an SVI, on page 835 These interfaces must have IP addresses assigned to them.
Step 4	**ip multicast boundary** *access-list-number* **Example:** `Switch(config-if)# ip multicast boundary 99`	Configures the boundary, specifying the access list you created in Step 2. Additional command options include: • For *access-list-number*, the ranges are as follows: ◦ access-list-number 1—99 (IP standard access list) ◦ access-list-number 100—199 (IP extended access list) ◦ access-list-number 1300—1999 (IP standard access list - expanded range) ◦ access-list-number 2000—2699 (IP extended access list - expanded range) • *Word*—IP named access list. • **filter-autorp**—Filter AutoRP packet contents. • **in**—Restrict (s,g) creation when this interface is the RPF. • **out**—Restrict interface addition to outgoing list.

	Command or Action	Purpose
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

What to Do Next

Proceed to the other supported IP multicast routing procedures.

Related Topics

Multicast Boundaries, on page 912

Example: Configuring an IP Multicast Boundary, on page 928

Monitoring and Maintaining IP Multicast Routing

Clearing Caches, Tables, and Databases

You can remove all contents of a particular cache, table, or database. Clearing a cache, table, or database might be necessary when the contents of the particular structure are or suspected to be invalid.

You can use any of the privileged EXEC commands in the following table to clear IP multicast caches, tables, and databases.

Table 110: Commands for Clearing Caches, Tables, and Databases

Command	Purpose
clear ip igmp group {**group** [*hostname* | *IP address*] | **vrf** *name* **group** [*hostname* | *IP address*] }	Deletes entries from the IGMP cache.

Command	Purpose
clear ip mfib { **counters** [*group* | *source*] | **global counters** [*group* | *source*] | **vrf *** }	Clears all active IPv4 Multicast Forwarding Information Base (MFIB) traffic counters.
clear ip mrm {**status-report** [*source*] }	IP multicast routing clear commands.
clear ip mroute { ***** | [*hostname* | *IP address*] | **vrf** *name* **group** [*hostname* | *IP address*] }	Deletes entries from the IP multicast routing table.
clear ip msdp { **peer** | **sa-cache** | **statistics** | **vrf** }	Clears the Multicast Source Discovery Protocol (MSDP) cache.
clear ip multicast { **limit** | **redundancy statistics** }	Clears the IP multicast information.
clear ip pim { **df** [**int** | **rp** *rp address*] | **interface** | **rp-mapping** [*rp address*] | **vrf** *vpn name* { **df** | **interface** | **rp-mapping** }	Clears the PIM cache.
clear ip sap [*group-address* | "*session-name*"]	Deletes the Session Directory Protocol Version 2 cache or an sdr cache entry.

Displaying System and Network Statistics

You can display specific statistics, such as the contents of IP routing tables, caches, and databases.

Note This release does not support per-route statistics.

You can display information to learn resource usage and solve network problems. You can also display information about node reachability and discover the routing path that packets of your device are taking through the network.

You can use any of the privileged EXEC commands in the following table to display various routing statistics.

Table 111: Commands for Displaying System and Network Statistics

Command	Purpose
ping [*group-name* | *group-address*]	Sends an ICMP Echo Request to a multicast group address.
show ip igmp filter	Displays IGMP filter information.
show ip igmp groups [*type-number* | *detail*]	Displays the multicast groups that are directly connected to the switch and that were learned through IGMP.

Command	Purpose
show ip igmp interface [*type number*]	Displays multicast-related information about an interface.
show ip igmp membership [*name/group address* | **all** | **tracked**]	Displays IGMP membership information for forwarding.
show ip igmp profile [profile_number]	Displays IGMP profile information.
show ip igmp ssm-mapping [*hostname/IP address*]	Displays IGMP SSM mapping information.
show ip igmp static-group {**class-map** [**interface** [type]]	Displays static group information.
show ip igmp vrf	Displays the selected VPN Routing/Forwarding instance by name.
show ip mfib [*type number*]	Displays the IP multicast forwarding information base.
show ip mrib { **client** | **route** | **vrf** }	Displays the multicast routing information base.
show ip mrm { **interface** | **manager** | **status-report** }	Displays the IP multicast routing monitor information.
show ip mroute [*group-name* | *group-address*] [*source*] [**count** | **interface** | **proxy** | **pruned** | **summary** | **verbose**]	Displays the contents of the IP multicast routing table.
show ip msdp { **count** | **peer** | **rpf-peer** | **sa-cache** | **summary** | **vrf** }	Displays the Multicast Source Discovery Protocol (MSDP) information.
show ip multicast [**interface** | **limit** | **mpls** | **redundancy** | **vrf**]	Displays global multicast information.
show ip pim interface [*type number*] [**count** | **detail** | **df** | **stats**]	Displays information about interfaces configured for PIM. This command is available in all software images.
show ip pim all-vrfs { **tunnel** }	Display all VRFs.
show ip pim autorp	Display global auto-RP information.
show ip pim boundary [*type number*]	Displays boundary information.
show ip pim bsr-router	Display bootstrap router information (version 2).
show ip pim interface [*type number*]	Displays PIM interface information.

Command	Purpose
show ip pim mdt [**bgp**]	Displays multicast tunnel information.
show ip pim neighbor [*type number*]	Lists the PIM neighbors discovered by the switch. This command is available in all software images.
show ip pim rp [*group-name* | *group-address*]	Displays the RP routers associated with a sparse-mode multicast group. This command is available in all software images.
show ip pim rp-hash [*group-name* | *group-address*]	Displays the RP to be chosen based upon the group selected.
show ip pim tunnel [*tunnel* | *verbose*]	Displays the registered tunnels.
show ip pim vrf *name*	Displays VPN routing and forwarding instances.
show ip rpf {*source-address* | *name*}	Displays how the switch is doing Reverse-Path Forwarding (that is, from the unicast routing table, DVMRP routing table, or static mroutes). Command parameters include: • *Host name* or *IP address*—IP name or group address. • **Select**—Group-based VRF select information. • **vrf**—Selects VPN Routing/Forwarding instance.
show ip sap [*group* | "*session-name*" | **detail**]	Displays the Session Announcement Protocol (SAP) Version 2 cache. Command parameters include: • *A.B.C.D*—IP group address. • *WORD*—Session name (in double quotes). • **detail**—Session details.

Monitoring IP Multicast Routing

You can use the privileged EXEC commands in the following table to monitor IP multicast routers, packets, and paths.

Table 112: Commands for Monitoring IP Multicast Routing

Command	Purpose
mrinfo { [*hostname* | *address*] | **vrf** }	Queries a multicast router or multilayer switch about which neighboring multicast devices are peering with it.
mstat { [*hostname* | *address*] | **vrf** }	Displays IP multicast packet rate and loss information.
mtrace { [*hostname* | *address*] | **vrf** }	Traces the path from a source to a destination branch for a multicast distribution tree for a given group.

Configuration Examples for IP Multicast Routing

Example: Configuring an IP Multicast Boundary

This example shows how to set up a boundary for all administratively-scoped addresses:

```
Switch(config)# access-list 1 deny 239.0.0.0 0.255.255.255
Switch(config)# access-list 1 permit 224.0.0.0 15.255.255.255
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip multicast boundary 1
```

Related Topics

Configuring an IP Multicast Boundary , on page 922

Multicast Boundaries, on page 912

Example: Responding to mrinfo Requests

The software answers mrinfo requests sent by mrouted systems and Cisco routers and multilayer switches. The software returns information about neighbors through DVMRP tunnels and all the routed interfaces. This information includes the metric (always set to 1), the configured TTL threshold, the status of the interface, and various flags. You can also use the **mrinfo** privileged EXEC command to query the router or switch itself, as in this example:

```
Switch# mrinfo
  171.69.214.27 (mm1-7kd.cisco.com) [version cisco 11.1] [flags: PMS]:
  171.69.214.27 -> 171.69.214.26 (mm1-r7kb.cisco.com) [1/0/pim/querier]
  171.69.214.27 -> 171.69.214.25 (mm1-45a.cisco.com) [1/0/pim/querier]
  171.69.214.33 -> 171.69.214.34 (mm1-45c.cisco.com) [1/0/pim]
  171.69.214.137 -> 0.0.0.0 [1/0/pim/querier/down/leaf]
  171.69.214.203 -> 0.0.0.0 [1/0/pim/querier/down/leaf]
  171.69.214.18 -> 171.69.214.20 (mm1-45e.cisco.com) [1/0/pim]
  171.69.214.18 -> 171.69.214.19 (mm1-45c.cisco.com) [1/0/pim]
  171.69.214.18 -> 171.69.214.17 (mm1-45a.cisco.com) [1/0/pim]
```

Where to Go Next for IP Multicast

You can configure the following:

- IGMP
- Wireless Multicast
- PIM
- SSM
- Service Discovery Gateway

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	*IP Multicast Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*
For information on configuring the Multicast Source Discovery Protocol (MSDP).	*Catalyst 3850 Routing Configuration Guide*
Platform-independent configuration information	• *IP Multicast: PIM Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)* • *IP Multicast: IGMP Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)* • *IP Multicast: Multicast Optimization Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*

Standards and RFCs

Standard/RFC	Title
RFC 1112	*Host Extensions for IP Multicasting*
RFC 2236	*Internet Group Management Protocol, Version 2*
RFC 4601	*Protocol-Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification*

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for IP Multicast

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

PART VII

Security

CHAPTER 43

Preventing Unauthorized Access

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Preventing Unauthorized Access

You can prevent unauthorized users from reconfiguring your switch and viewing configuration information. Typically, you want network administrators to have access to your switch while you restrict access to users who dial from outside the network through an asynchronous port, connect from outside the network through a serial port, or connect through a terminal or workstation from within the local network.

To prevent unauthorized access into your switch, you should configure one or more of these security features:

- At a minimum, you should configure passwords and privileges at each switch port. These passwords are locally stored on the switch. When users attempt to access the switch through a port or line, they must enter the password specified for the port or line before they can access the switch.
- For an additional layer of security, you can also configure username and password pairs, which are locally stored on the switch. These pairs are assigned to lines or ports and authenticate each user before that user can access the switch. If you have defined privilege levels, you can also assign a specific privilege level (with associated rights and privileges) to each username and password pair.

- If you want to use username and password pairs, but you want to store them centrally on a server instead of locally, you can store them in a database on a security server. Multiple networking devices can then use the same database to obtain user authentication (and, if necessary, authorization) information.
- You can also enable the login enhancements feature, which logs both failed and unsuccessful login attempts. Login enhancements can also be configured to block future login attempts after a set number of unsuccessful attempts are made. For more information, see the Cisco IOS Login Enhancements documentation.

Related Topics

Configuring Username and Password Pairs, on page 944
TACACS+ and Switch Access, on page 953
Setting a Telnet Password for a Terminal Line, on page 942

CHAPTER 44

Controlling Switch Access with Passwords and Privilege Levels

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Restrictions for Controlling Switch Access with Passwords and Privileges

The following are the restrictions for controlling switch access with passwords and privileges:

- Disabling password recovery will not work if you have set the switch to boot up manually by using the **boot manual** global configuration command. This command produces the boot loader prompt (*switch:*) after the switch is power cycled.

Related Topics

Disabling Password Recovery, on page 941
Password Recovery, on page 937

Information About Passwords and Privilege Levels

Default Password and Privilege Level Configuration

A simple way of providing terminal access control in your network is to use passwords and assign privilege levels. Password protection restricts access to a network or network device. Privilege levels define what commands users can enter after they have logged into a network device.

This table shows the default password and privilege level configuration.

Table 113: Default Password and Privilege Levels

Feature	Default Setting
Enable password and privilege level	No password is defined. The default is level 15 (privileged EXEC level). The password is not encrypted in the configuration file.
Enable secret password and privilege level	No password is defined. The default is level 15 (privileged EXEC level). The password is encrypted before it is written to the configuration file.
Line password	No password is defined.

Additional Password Security

To provide an additional layer of security, particularly for passwords that cross the network or that are stored on a Trivial File Transfer Protocol (TFTP) server, you can use either the **enable password** or **enable secret** global configuration commands. Both commands accomplish the same thing; that is, you can establish an encrypted password that users must enter to access privileged EXEC mode (the default) or any privilege level you specify.

We recommend that you use the **enable secret** command because it uses an improved encryption algorithm.

If you configure the **enable secret** command, it takes precedence over the **enable password** command; the two commands cannot be in effect simultaneously.

If you enable password encryption, it applies to all passwords including username passwords, authentication key passwords, the privileged command password, and console and virtual terminal line passwords.

Related Topics

Protecting Enable and Enable Secret Passwords with Encryption, on page 939
Example: Protecting Enable and Enable Secret Passwords with Encryption, on page 949

Password Recovery

By default, any end user with physical access to the switch can recover from a lost password by interrupting the boot process while the switch is powering on and then by entering a new password.

The password-recovery disable feature protects access to the switch password by disabling part of this functionality. When this feature is enabled, the end user can interrupt the boot process only by agreeing to set the system back to the default configuration. With password recovery disabled, you can still interrupt the boot process and change the password, but the configuration file (config.text) and the VLAN database file (vlan.dat) are deleted.

If you disable password recovery, we recommend that you keep a backup copy of the configuration file on a secure server in case the end user interrupts the boot process and sets the system back to default values. Do not keep a backup copy of the configuration file on the switch. If the switch is operating in VTP transparent mode, we recommend that you also keep a backup copy of the VLAN database file on a secure server. When the switch is returned to the default system configuration, you can download the saved files to the switch by using the Xmodem protocol.

To re-enable password recovery, use the **service password-recovery** global configuration command.

Related Topics

Disabling Password Recovery, on page 941
Restrictions for Controlling Switch Access with Passwords and Privileges, on page 935

Terminal Line Telnet Configuration

When you power-up your switch for the first time, an automatic setup program runs to assign IP information and to create a default configuration for continued use. The setup program also prompts you to configure your switch for Telnet access through a password. If you did not configure this password during the setup program, you can configure it when you set a Telnet password for a terminal line. For more information on doing this, see Related Topics.

Related Topics

Setting a Telnet Password for a Terminal Line, on page 942
Example: Setting a Telnet Password for a Terminal Line, on page 949

Username and Password Pairs

You can configure username and password pairs, which are locally stored on the switch. These pairs are assigned to lines or ports and authenticate each user before that user can access the switch. If you have defined privilege levels, you can also assign a specific privilege level (with associated rights and privileges) to each username and password pair.

Related Topics

Configuring Username and Password Pairs, on page 944

Privilege Levels

Cisco switches (and other devices) use privilege levels to provide password security for different levels of switch operation. By default, the Cisco IOS software operates in two modes (privilege levels) of password security: user EXEC (Level 1) and privileged EXEC (Level 15). You can configure up to 16 hierarchical levels of commands for each mode. By configuring multiple passwords, you can allow different sets of users to have access to specified commands.

Privilege Levels on Lines

Users can override the privilege level you set using the **privilege level** line configuration command by logging in to the line and enabling a different privilege level. They can lower the privilege level by using the **disable** command. If users know the password to a higher privilege level, they can use that password to enable the higher privilege level. You might specify a high level or privilege level for your console line to restrict line usage.

For example, if you want many users to have access to the **clear line** command, you can assign it level 2 security and distribute the level 2 password fairly widely. But if you want more restricted access to the **configure** command, you can assign it level 3 security and distribute that password to a more restricted group of users.

Command Privilege Levels

When you set a command to a privilege level, all commands whose syntax is a subset of that command are also set to that level. For example, if you set the **show ip traffic** command to level 15, the **show** commands and **show ip** commands are automatically set to privilege level 15 unless you set them individually to different levels.

Related Topics

Setting the Privilege Level for a Command, on page 945
Example: Setting the Privilege Level for a Command, on page 949
Changing the Default Privilege Level for Lines, on page 946
Logging into and Exiting a Privilege Level, on page 947

How to Control Switch Access with Passwords and Privilege Levels

Setting or Changing a Static Enable Password

The enable password controls access to the privileged EXEC mode. Beginning in privileged EXEC mode, follow these steps to set or change a static enable password:

SUMMARY STEPS

1. **configure terminal**
2. **enable password** *password*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**enable password** *password* **Example:** `Switch(config)# enable password secret321`	Defines a new password or changes an existing password for access to privileged EXEC mode. By default, no password is defined. For *password*, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. It can contain the question mark (?) character if you precede the question mark with the key combination Crtl-v when you create the password; for example, to create the password abc?123, do this: Enter **abc**. Enter **Crtl-v**. Enter **?123**. When the system prompts you to enter the enable password, you need not precede the question mark with the Ctrl-v; you can simply enter abc?123 at the password prompt.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Related Topics

Example: Setting or Changing a Static Enable Password, on page 948

Protecting Enable and Enable Secret Passwords with Encryption

Beginning in privileged EXEC mode, follow these steps to establish an encrypted password that users must enter to access privileged EXEC mode (the default) or any privilege level you specify:

SUMMARY STEPS

1. **configure terminal**
2. Use one of the following:
   - **enable password** [**level** *level*] {*password* | *encryption-type encrypted-password*}
   - **enable secret** [**level** *level*] {*password* | *encryption-type encrypted-password*}
3. **service password-encryption**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	Use one of the following:
	• **enable password** [**level** *level*] {*password* | *encryption-type encrypted-password*}
	• **enable secret** [**level** *level*] {*password* | *encryption-type encrypted-password*}
	
	Example:
	`Switch(config)# enable password example102`
	or
	`Switch(config)# enable secret level 1 password secret123sample`
		• Defines a new password or changes an existing password for access to privileged EXEC mode.
		• Defines a secret password, which is saved using a nonreversible encryption method.
		◦ (Optional) For *level*, the range is from 0 to 15. Level 1 is normal user EXEC mode privileges. The default level is 15 (privileged EXEC mode privileges).
		◦ For *password*, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined.
		◦ (Optional) For *encryption-type*, only type 5, a Cisco proprietary encryption algorithm, is available. If you specify an encryption type, you must provide an encrypted password—an encrypted password that you copy from another switch configuration.
		
		Note If you specify an encryption type and then enter a clear text password, you can not re-enter privileged EXEC mode. You cannot recover a lost encrypted password by any method.

	Command or Action	Purpose
Step 3	**service password-encryption** **Example:** `Switch(config)# service password-encryption`	(Optional) Encrypts the password when the password is defined or when the configuration is written. Encryption prevents the password from being readable in the configuration file.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Related Topics

Additional Password Security, on page 936

Example: Protecting Enable and Enable Secret Passwords with Encryption, on page 949

Disabling Password Recovery

Beginning in privileged EXEC mode, follow these steps to disable password recovery to protect the security of your switch:

Before You Begin

If you disable password recovery, we recommend that you keep a backup copy of the configuration file on a secure server in case the end user interrupts the boot process and sets the system back to default values. Do not keep a backup copy of the configuration file on the switch. If the switch is operating in VTP transparent mode, we recommend that you also keep a backup copy of the VLAN database file on a secure server. When the switch is returned to the default system configuration, you can download the saved files to the switch by using the Xmodem protocol.

SUMMARY STEPS

1. **configure terminal**
2. **no service password-recovery**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**no service password-recovery** **Example:** `Switch(config)# no service password-recovery`	Disables password recovery. This setting is saved in an area of the flash memory that is accessible by the boot loader and the Cisco IOS image, but it is not part of the file system and is not accessible by any user.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

What to Do Next

To re-enable password recovery, use the **service password-recovery** global configuration command.

Related Topics

Password Recovery, on page 937

Restrictions for Controlling Switch Access with Passwords and Privileges, on page 935

Setting a Telnet Password for a Terminal Line

Beginning in user EXEC mode, follow these steps to set a Telnet password for the connected terminal line:

Before You Begin

Attach a PC or workstation with emulation software to the switch console port, or attach a PC to the Ethernet management port.

The default data characteristics of the console port are 9600, 8, 1, no parity. You might need to press the Return key several times to see the command-line prompt.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line vty 0 15**
4. **password** *password*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch> enable`	**Note** If a password is required for access to privileged EXEC mode, you will be prompted for it. Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 3	**line vty 0 15** **Example:** `Switch(config)# line vty 0 15`	Configures the number of Telnet sessions (lines), and enters line configuration mode. There are 16 possible sessions on a command-capable switch. The 0 and 15 mean that you are configuring all 16 possible Telnet sessions.
Step 4	**password** *password* **Example:** `Switch(config-line)# password abcxyz543`	Sets a Telnet password for the line or lines. For *password*, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined.
Step 5	**end** **Example:** `Switch(config-line)# end`	Returns to privileged EXEC mode.

Related Topics

Preventing Unauthorized Access, on page 933
Terminal Line Telnet Configuration, on page 937

Example: Setting a Telnet Password for a Terminal Line, on page 949

Configuring Username and Password Pairs

Beginning in privileged EXEC mode, follow these steps to configure username and password pairs:

SUMMARY STEPS

1. **configure terminal**
2. **username** *name* [**privilege** *level*] {**password** *encryption-type password*}
3. Use one of the following:
   - **line console 0**
   - **line vty 0 15**
4. **login local**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**username** *name* [**privilege** *level*] {**password** *encryption-type password*} **Example:** `Switch(config)# username adamsample privilege 1 password secret456`	Sets the username, privilege level, and password for each user. • For *name*, specify the user ID as one word. Spaces and quotation marks are not allowed. • (Optional) For *level*, specify the privilege level the user has after gaining access. The range is 0 to 15. Level 15 gives privileged EXEC mode access. Level 1 gives user EXEC mode access. • For *encryption-type*, enter 0 to specify that an unencrypted password will follow. Enter 7 to specify that a hidden password will follow. • For *password*, specify the password the user must enter to gain access to the switch. The password must be from 1 to 25 characters, can contain embedded spaces, and must be the last option specified in the **username** command.
Step 3	Use one of the following: • **line console 0** • **line vty 0 15**	Enters line configuration mode, and configures the console port (line 0) or the VTY lines (line 0 to 15).

	Command or Action	Purpose

	Example:
	`Switch(config)# line console 0`
	or
	`Switch(config)# line vty 15`	

Step 4	**login local** **Example:** `Switch(config-line)# login local`	Enables local password checking at login time. Authentication is based on the username specified in Step 2.
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Related Topics

Preventing Unauthorized Access, on page 933
Username and Password Pairs, on page 937

Setting the Privilege Level for a Command

Beginning in privileged EXEC mode, follow these steps to set the privilege level for a command:

SUMMARY STEPS

1. **configure terminal**
2. **privilege** *mode* **level** *level command*
3. **enable password level** *level password*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	**privilege** *mode* **level** *level command* **Example:** `Switch(config)# privilege exec level 14 configure`	Sets the privilege level for a command. • For *mode*, enter **configure** for global configuration mode, **exec** for EXEC mode, **interface** for interface configuration mode, or **line** for line configuration mode. • For *level*, the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. Level 15 is the level of access permitted by the **enable** password. • For *command*, specify the command to which you want to restrict access.
Step 3	**enable password level** *level password* **Example:** `Switch(config)# enable password level 14 SecretPswd14`	Specifies the password to enable the privilege level. • For *level*, the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. • For *password*, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Related Topics

Privilege Levels, on page 938

Example: Setting the Privilege Level for a Command, on page 949

Changing the Default Privilege Level for Lines

Beginning in privileged EXEC mode, follow these steps to change the default privilege level for the specified line:

SUMMARY STEPS

1. **configure terminal**
2. **line vty** *line*
3. **privilege level** *level*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**line vty** *line* **Example:** `Switch(config)# line vty 10`	Selects the virtual terminal line on which to restrict access.
Step 3	**privilege level** *level* **Example:** `Switch(config)# privilege level 15`	Changes the default privilege level for the line. For *level*, the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. Level 15 is the level of access permitted by the **enable** password.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

What to Do Next

Users can override the privilege level you set using the **privilege level** line configuration command by logging in to the line and enabling a different privilege level. They can lower the privilege level by using the **disable** command. If users know the password to a higher privilege level, they can use that password to enable the higher privilege level. You might specify a high level or privilege level for your console line to restrict line usage.

Related Topics

Privilege Levels, on page 938

Logging into and Exiting a Privilege Level

Beginning in user EXEC mode, follow these steps to log into a specified privilege level and exit a specified privilege level.

SUMMARY STEPS

1. **enable** *level*
2. **disable** *level*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** *level* **Example:** `Switch> enable 15`	Logs in to a specified privilege level. Following the example, Level 15 is privileged EXEC mode. For *level*, the range is 0 to 15.
Step 2	**disable** *level* **Example:** `Switch# disable 1`	Exits to a specified privilege level. Following the example, Level 1 is user EXEC mode. For *level*, the range is 0 to 15.

Related Topics

Privilege Levels, on page 938

Monitoring Switch Access

Table 114: Commands for Displaying DHCP Information

show privilege	Displays the privilege level configuration.

Configuration Examples for Setting Passwords and Privilege Levels

Example: Setting or Changing a Static Enable Password

This example shows how to change the enable password to *l1u2c3k4y5*. The password is not encrypted and provides access to level 15 (traditional privileged EXEC mode access):

```
Switch(config)# enable password l1u2c3k4y5
```

Related Topics

Setting or Changing a Static Enable Password, on page 938

Example: Protecting Enable and Enable Secret Passwords with Encryption

This example shows how to configure the encrypted password *1FaD0$Xyti5Rkls3LoyxzS8* for privilege level 2:

```
Switch(config)# enable secret level 2 5 $1$FaD0$Xyti5Rkls3LoyxzS8
```

Related Topics

Protecting Enable and Enable Secret Passwords with Encryption, on page 939
Additional Password Security, on page 936

Example: Setting a Telnet Password for a Terminal Line

This example shows how to set the Telnet password to *let45me67in89*:

```
Switch(config)# line vty 10
Switch(config-line)# password let45me67in89
```

Related Topics

Setting a Telnet Password for a Terminal Line, on page 942
Terminal Line Telnet Configuration, on page 937

Example: Setting the Privilege Level for a Command

This example shows how to set the **configure** command to privilege level 14 and define *SecretPswd14* as the password users must enter to use level 14 commands:

```
Switch(config)# privilege exec level 14 configure
Switch(config)# enable password level 14 SecretPswd14
```

Related Topics

Setting the Privilege Level for a Command, on page 945
Privilege Levels, on page 938

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

CHAPTER 45

Configuring TACACS+

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for Controlling Switch Access with Terminal Access Controller Access Control System Plus (TACACS+)

The following are the prerequisites for set up and configuration of switch access with Terminal Access Controller Access Control System Plus (TACACS+) (must be performed in the order presented):

1. Configure the switches with the TACACS+ server addresses.
2. Set an authentication key.
3. Configure the key from Step 2 on the TACACS+ servers.

4 Enable AAA.

5 Create a login authentication method list.

6 Apply the list to the terminal lines.

7 Create an authorization and accounting method list.

The following are the prerequisites for controlling switch access with TACACS+:

- You must have access to a configured TACACS+ server to configure TACACS+ features on your switch. Also, you must have access to TACACS+ services maintained in a database on a TACACS+ daemon typically running on a LINUX or Windows workstation.
- We recommend a redundant connection between a switch stack and the TACACS+ server. This is to help ensure that the TACACS+ server remains accessible in case one of the connected stack members is removed from the switch stack.
- You need a system running the TACACS+ daemon software to use TACACS+ on your switch.
- To use TACACS+, it must be enabled.
- Authorization must be enabled on the switch to be used.
- Users must first successfully complete TACACS+ authentication before proceeding to TACACS+ authorization.
- To use any of the AAA commands listed in this section or elsewhere, you must first enable AAA with the **aaa new-model** command.
- At a minimum, you must identify the host or hosts maintaining the TACACS+ daemon and define the method lists for TACACS+ authentication. You can optionally define method lists for TACACS+ authorization and accounting.
- The method list defines the types of authentication to be performed and the sequence in which they are performed; it must be applied to a specific port before any of the defined authentication methods are performed. The only exception is the default method list (which, by coincidence, is named *default*). The default method list is automatically applied to all ports except those that have a named method list explicitly defined. A defined method list overrides the default method list.
- Use TACACS+ for privileged EXEC access authorization if authentication was performed by using TACACS+.
- Use the local database if authentication was not performed by using TACACS+.

Related Topics

TACACS+ Overview, on page 953
TACACS+ Operation, on page 955
How to Configure TACACS+, on page 957
Method List Description, on page 956
Configuring TACACS+ Login Authentication, on page 959
TACACS+ Login Authentication, on page 956
Configuring TACACS+ Authorization for Privileged EXEC Access and Network Services, on page 961
TACACS+ Authorization for Privileged EXEC Access and Network Services, on page 956

Information About TACACS+

TACACS+ and Switch Access

This section describes TACACS+. TACACS+ provides detailed accounting information and flexible administrative control over the authentication and authorization processes. It is facilitated through authentication, authorization, accounting (AAA) and can be enabled only through AAA commands.

The switch supports TACACS+ for IPv6. Information is in the “TACACS+ Over an IPv6 Transport” section of the “Implementing ADSL for IPv6” chapter in the *Cisco IOS XE IPv6 Configuration Guide, Release 2*.

For information about configuring this feature, see the “Configuring TACACS+ over IPv6” section of the “Implementing ADSL for IPv6” chapter in the *Cisco IOS XE IPv6 Configuration Guide, Release 2*.

Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Security Command Reference, Release 12.4* and the *Cisco IOS IPv6 Command Reference*.

Related Topics

Preventing Unauthorized Access, on page 933
Configuring the Switch for Local Authentication and Authorization, on page 1007
SSH Servers, Integrated Clients, and Supported Versions, on page 1013

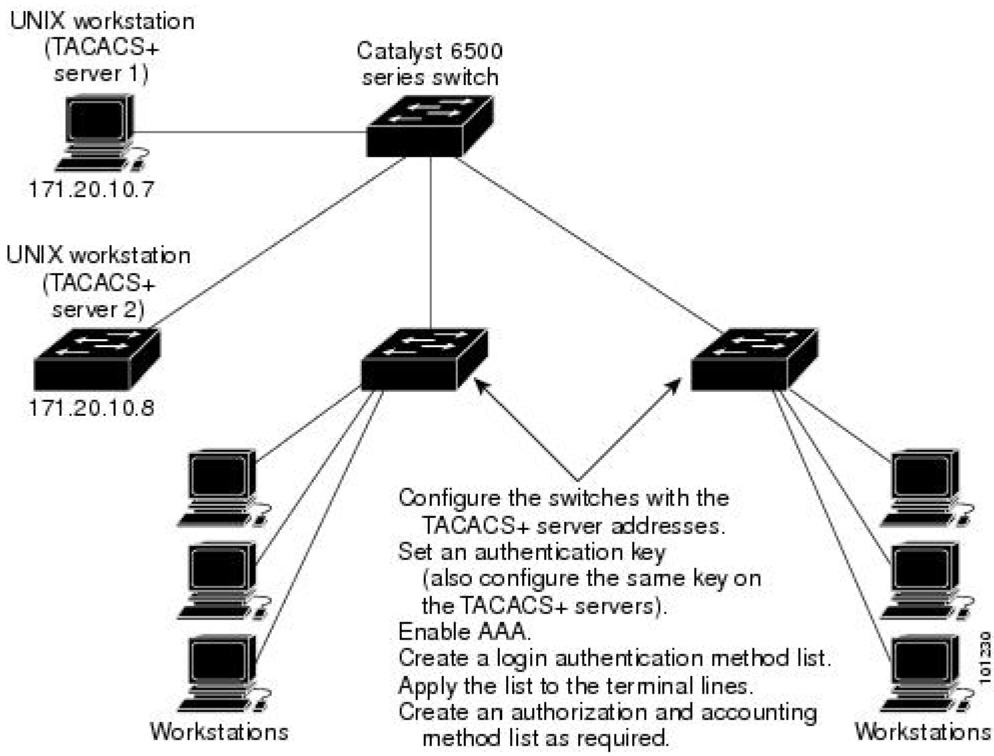
TACACS+ Overview

TACACS+ is a security application that provides centralized validation of users attempting to gain access to your switch.

TACACS+ provides for separate and modular authentication, authorization, and accounting facilities. TACACS+ allows for a single access control server (the TACACS+ daemon) to provide each service—authentication, authorization, and accounting—independently. Each service can be tied into its own database to take advantage of other services available on that server or on the network, depending on the capabilities of the daemon.

The goal of TACACS+ is to provide a method for managing multiple network access points from a single management service. Your switch can be a network access server along with other Cisco routers and access servers.

Figure 34: Typical TACACS+ Network Configuration



TACACS+, administered through the AAA security services, can provide these services:

- Authentication—Provides complete control of authentication through login and password dialog, challenge and response, and messaging support.

  The authentication facility can conduct a dialog with the user (for example, after a username and password are provided, to challenge a user with several questions, such as home address, mother's maiden name, service type, and social security number). The TACACS+ authentication service can also send messages to user screens. For example, a message could notify users that their passwords must be changed because of the company's password aging policy.

- Authorization—Provides fine-grained control over user capabilities for the duration of the user's session, including but not limited to setting autocommands, access control, session duration, or protocol support. You can also enforce restrictions on what commands a user can execute with the TACACS+ authorization feature.

- Accounting—Collects and sends information used for billing, auditing, and reporting to the TACACS+ daemon. Network managers can use the accounting facility to track user activity for a security audit or to provide information for user billing. Accounting records include user identities, start and stop times, executed commands (such as PPP), number of packets, and number of bytes.

The TACACS+ protocol provides authentication between the switch and the TACACS+ daemon, and it ensures confidentiality because all protocol exchanges between the switch and the TACACS+ daemon are encrypted.

Related Topics

Prerequisites for Controlling Switch Access with Terminal Access Controller Access Control System Plus (TACACS+), on page 951

TACACS+ Operation

When a user attempts a simple ASCII login by authenticating to a switch using TACACS+, this process occurs:

1. When the connection is established, the switch contacts the TACACS+ daemon to obtain a username prompt to show to the user. The user enters a username, and the switch then contacts the TACACS+ daemon to obtain a password prompt. The switch displays the password prompt to the user, the user enters a password, and the password is then sent to the TACACS+ daemon.

   TACACS+ allows a dialog between the daemon and the user until the daemon receives enough information to authenticate the user. The daemon prompts for a username and password combination, but can include other items, such as the user's mother's maiden name.

2. The switch eventually receives one of these responses from the TACACS+ daemon:

   - ACCEPT—The user is authenticated and service can begin. If the switch is configured to require authorization, authorization begins at this time.
   - REJECT—The user is not authenticated. The user can be denied access or is prompted to retry the login sequence, depending on the TACACS+ daemon.
   - ERROR—An error occurred at some time during authentication with the daemon or in the network connection between the daemon and the switch. If an ERROR response is received, the switch typically tries to use an alternative method for authenticating the user.
   - CONTINUE—The user is prompted for additional authentication information.

   After authentication, the user undergoes an additional authorization phase if authorization has been enabled on the switch. Users must first successfully complete TACACS+ authentication before proceeding to TACACS+ authorization.

3. If TACACS+ authorization is required, the TACACS+ daemon is again contacted, and it returns an ACCEPT or REJECT authorization response. If an ACCEPT response is returned, the response contains data in the form of attributes that direct the EXEC or NETWORK session for that user and the services that the user can access:

   - Telnet, Secure Shell (SSH), rlogin, or privileged EXEC services
   - Connection parameters, including the host or client IP address, access list, and user timeouts

Related Topics

Prerequisites for Controlling Switch Access with Terminal Access Controller Access Control System Plus (TACACS+), on page 951

Method List Description

A method list defines the sequence and methods to be used to authenticate, to authorize, or to keep accounts on a user. You can use method lists to designate one or more security protocols to be used, thus ensuring a backup system if the initial method fails. The software uses the first method listed to authenticate, to authorize, or to keep accounts on users; if that method does not respond, the software selects the next method in the list. This process continues until there is successful communication with a listed method or the method list is exhausted.

Related Topics

How to Configure TACACS+, on page 957
Prerequisites for Controlling Switch Access with Terminal Access Controller Access Control System Plus (TACACS+), on page 951

TACACS+ Configuration Options

You can configure the switch to use a single server or AAA server groups to group existing server hosts for authentication. You can group servers to select a subset of the configured server hosts and use them for a particular service. The server group is used with a global server-host list and contains the list of IP addresses of the selected server hosts.

Related Topics

Identifying the TACACS+ Server Host and Setting the Authentication Key, on page 957

TACACS+ Login Authentication

A method list describes the sequence and authentication methods to be queried to authenticate a user. You can designate one or more security protocols to be used for authentication, thus ensuring a backup system for authentication in case the initial method fails. The software uses the first method listed to authenticate users; if that method fails to respond, the software selects the next authentication method in the method list. This process continues until there is successful communication with a listed authentication method or until all defined methods are exhausted. If authentication fails at any point in this cycle—meaning that the security server or local username database responds by denying the user access—the authentication process stops, and no other authentication methods are attempted.

Related Topics

Configuring TACACS+ Login Authentication, on page 959
Prerequisites for Controlling Switch Access with Terminal Access Controller Access Control System Plus (TACACS+), on page 951

TACACS+ Authorization for Privileged EXEC Access and Network Services

AAA authorization limits the services available to a user. When AAA authorization is enabled, the switch uses information retrieved from the user’s profile, which is located either in the local user database or on the security server, to configure the user’s session. The user is granted access to a requested service only if the information in the user profile allows it.

Related Topics

Configuring TACACS+ Authorization for Privileged EXEC Access and Network Services, on page 961
Prerequisites for Controlling Switch Access with Terminal Access Controller Access Control System Plus (TACACS+), on page 951

TACACS+ Accounting

The AAA accounting feature tracks the services that users are accessing and the amount of network resources that they are consuming. When AAA accounting is enabled, the switch reports user activity to the TACACS+ security server in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the security server. This data can then be analyzed for network management, client billing, or auditing.

Related Topics

Starting TACACS+ Accounting, on page 962

Default TACACS+ Configuration

TACACS+ and AAA are disabled by default.

To prevent a lapse in security, you cannot configure TACACS+ through a network management application. When enabled, TACACS+ can authenticate users accessing the switch through the CLI.

Note Although TACACS+ configuration is performed through the CLI, the TACACS+ server authenticates HTTP connections that have been configured with a privilege level of 15.

How to Configure TACACS+

This section describes how to configure your switch to support TACACS+.

Related Topics

Method List Description, on page 956

Prerequisites for Controlling Switch Access with Terminal Access Controller Access Control System Plus (TACACS+), on page 951

Identifying the TACACS+ Server Host and Setting the Authentication Key

Beginning in privileged EXEC mode, follow these steps to identify the TACACS+ server host and set the authentication key:

SUMMARY STEPS

1. **configure terminal**
2. **tacacs-server host** *hostname*
3. **aaa new-model**
4. **aaa group server tacacs+** *group-name*
5. **server** *ip-address*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**tacacs-server host** *hostname* **Example:** `Switch(config)# tacacs-server host yourserver`	Identifies the IP host or hosts maintaining a TACACS+ server. Enter this command multiple times to create a list of preferred hosts. The software searches for hosts in the order in which you specify them. For *hostname*, specify the name or IP address of the host.
Step 3	**aaa new-model** **Example:** `Switch(config)# aaa new-model`	Enables AAA.
Step 4	**aaa group server tacacs+** *group-name* **Example:** `Switch(config)# aaa group server tacacs+ your_server_group`	(Optional) Defines the AAA server-group with a group name. This command puts the switch in a server group subconfiguration mode.
Step 5	**server** *ip-address* **Example:** `Switch(config)# server 10.1.2.3`	(Optional) Associates a particular TACACS+ server with the defined server group. Repeat this step for each TACACS+ server in the AAA server group. Each server in the group must be previously defined in Step 2.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Related Topics

TACACS+ Configuration Options, on page 956

Configuring TACACS+ Login Authentication

Beginning in privileged EXEC mode, follow these steps to configure TACACS+ login authentication:

Before You Begin

To configure AAA authentication, you define a named list of authentication methods and then apply that list to various ports.

Note To secure the switch for HTTP access by using AAA methods, you must configure the switch with the **ip http authentication aaa** global configuration command. Configuring AAA authentication does not secure the switch for HTTP access by using AAA methods.

For more information about the **ip http authentication** command, see the *Cisco IOS Security Command Reference, Release 12.4*.

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **aaa authentication login** {**default** | *list-name*} *method1* [*method2...*]
4. **line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*]
5. **login authentication** {**default** | *list-name*}
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**aaa new-model** **Example:** `Switch(config)# aaa new-model`	Enables AAA.

	Command or Action	Purpose
Step 3	**aaa authentication login** {**default** | *list-name*} *method1* [*method2...*] **Example:** `Switch(config)# aaa authentication login default tacacs+ local`	Creates a login authentication method list. • To create a default list that is used when a named list is *not* specified in the **login authentication** command, use the **default** keyword followed by the methods that are to be used in default situations. The default method list is automatically applied to all ports. • For *list-name*, specify a character string to name the list you are creating. • For *method1...*, specify the actual method the authentication algorithm tries. The additional methods of authentication are used only if the previous method returns an error, not if it fails. Select one of these methods: • *enable*—Use the enable password for authentication. Before you can use this authentication method, you must define an enable password by using the **enable** *password* global configuration command. • *group tacacs+*—Uses TACACS+ authentication. Before you can use this authentication method, you must configure the TACACS+ server. For more information, see the Identifying the TACACS+ Server Host and Setting the Authentication Key, on page 957. • *line* —Use the line password for authentication. Before you can use this authentication method, you must define a line password. Use the **password** *password* line configuration command. • *local*—Use the local username database for authentication. You must enter username information in the database. Use the **username** *password* global configuration command. • *local-case*—Use a case-sensitive local username database for authentication. You must enter username information in the database by using the **username** *name* **password** global configuration command. • *none*—Do not use any authentication for login.
Step 4	**line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*] **Example:** `Switch(config)# line 2 4`	Enters line configuration mode, and configures the lines to which you want to apply the authentication list.
Step 5	**login authentication** {**default** | *list-name*} **Example:** `Switch(config-line)# login`	Applies the authentication list to a line or set of lines. • If you specify **default**, use the default list created with the **aaa authentication login** command. • For *list-name*, specify the list created with the **aaa authentication login** command.

	Command or Action	Purpose
	`authentication default`	
Step 6	**end** **Example:** `Switch(config-line)# end`	Returns to privileged EXEC mode.

Related Topics

TACACS+ Login Authentication, on page 956

Prerequisites for Controlling Switch Access with Terminal Access Controller Access Control System Plus (TACACS+), on page 951

Configuring TACACS+ Authorization for Privileged EXEC Access and Network Services

You can use the **aaa authorization** global configuration command with the **tacacs+** keyword to set parameters that restrict a user's network access to privileged EXEC mode.

The **aaa authorization exec tacacs+ local** command sets these authorization parameters:

- Use TACACS+ for privileged EXEC access authorization if authentication was performed by using TACACS+.
- Use the local database if authentication was not performed by using TACACS+.

Note Authorization is bypassed for authenticated users who log in through the CLI even if authorization has been configured.

SUMMARY STEPS

1. **configure terminal**
2. **aaa authorization network tacacs+**
3. **aaa authorization exec tacacs+**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch#` **`configure terminal`**	Enters the global configuration mode.
Step 2	**aaa authorization network tacacs+** **Example:** `Switch(config)#` **`aaa authorization network tacacs+`**	Configures the switch for user TACACS+ authorization for all network-related service requests.
Step 3	**aaa authorization exec tacacs+** **Example:** `Switch(config)#` **`aaa authorization exec tacacs+`**	Configures the switch for user TACACS+ authorization if the user has privileged EXEC access. The **exec** keyword might return user profile information (such as **autocommand** information).
Step 4	**end** **Example:** `Switch(config)#` **`end`**	Returns to privileged EXEC mode.

Related Topics

TACACS+ Authorization for Privileged EXEC Access and Network Services, on page 956

Prerequisites for Controlling Switch Access with Terminal Access Controller Access Control System Plus (TACACS+), on page 951

Starting TACACS+ Accounting

Beginning in privileged EXEC mode, follow these steps to start TACACS+ Accounting:

SUMMARY STEPS

1. **configure terminal**
2. **aaa accounting network start-stop tacacs+**
3. **aaa accounting exec start-stop tacacs+**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**aaa accounting network start-stop tacacs+** **Example:** `Switch(config)# aaa accounting network start-stop tacacs+`	Enables TACACS+ accounting for all network-related service requests.
Step 3	**aaa accounting exec start-stop tacacs+** **Example:** `Switch(config)# aaa accounting exec start-stop tacacs+`	Enables TACACS+ accounting to send a start-record accounting notice at the beginning of a privileged EXEC process and a stop-record at the end.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

What to Do Next

To establish a session with a router if the AAA server is unreachable, use the **aaa accounting system guarantee-first** command. It guarantees system accounting as the first record, which is the default condition. In some situations, users might be prevented from starting a session on the console or terminal connection until after the system reloads, which can take more than 3 minutes.

To establish a console or Telnet session with the router if the AAA server is unreachable when the router reloads, use the **no aaa accounting system guarantee-first** command.

Related Topics

TACACS+ Accounting, on page 957

Establishing a Session with a Router if the AAA Server is Unreachable

To establishing a session with a router if the AAA server is unreachable, use the **aaa accounting system guarantee-first** command. It guarantees system accounting as the first record, which is the default condition. In some situations, users might be prevented from starting a session on the console or terminal connection until after the system reloads, which can take more than 3 minutes.

To establish a console or Telnet session with the router if the AAA server is unreachable when the router reloads, use the **no aaa accounting system guarantee-first** command.

Monitoring TACACS+

Table 115: Commands for Displaying TACACS+ Information

Command	Purpose
show tacacs	Displays TACACS+ server statistics.

Additional References

Related Documents

Related Topic	Document Title
Configuring Identity Control policies and Identity Service templates for Session Aware networking.	Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/en/US/docs/ios-xml/ios/san/configuration/xe-3se/3850/san-xe-3se-3850-book.html
Configuring RADIUS, TACACS+, Secure Shell, 802.1X and AAA.	Securing User Services Configuration Guide Library, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/en/US/docs/ios-xml/ios/security/config_library/xe-3se/3850/secuser-xe-3se-3850-library.html

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

CHAPTER 46

Configuring RADIUS

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for Controlling Switch Access with RADIUS

This section lists the prerequisites for controlling Catalyst switch access with RADIUS.

General:

- RADIUS and AAA must be enabled to use any of the configuration commands in this chapter.
- RADIUS is facilitated through AAA and can be enabled only through AAA commands.

- At a minimum, you must identify the host or hosts that run the RADIUS server software and define the method lists for RADIUS authentication. You can optionally define method lists for RADIUS authorization and accounting.
- You should have access to and should configure a RADIUS server before configuring RADIUS features on your switch.
- The RADIUS host is normally a multiuser system running RADIUS server software from Cisco (Cisco Secure Access Control Server Version 3.0), Livingston, Merit, Microsoft, or another software provider. For more information, see the RADIUS server documentation.
- To use the Change-of-Authorization (CoA) interface, a session must already exist on the switch. CoA can be used to identify a session and enforce a disconnect request. The update affects only the specified session.
- A redundant connection between a switch stack and the RADIUS server is recommended. This is to help ensure that the RADIUS server remains accessible in case one of the connected stack members is removed from the switch stack.

For RADIUS operation:

- Users must first successfully complete RADIUS authentication before proceeding to RADIUS authorization, if it is enabled.

Related Topics

RADIUS and Switch Access, on page 969
RADIUS Operation, on page 970

Restrictions for Controlling Switch Access with RADIUS

This topic covers restrictions for controlling switch access with RADIUS.

General:

- To prevent a lapse in security, you cannot configure RADIUS through a network management application.

RADIUS is not suitable in the following network security situations:

- Multiprotocol access environments. RADIUS does not support AppleTalk Remote Access (ARA), NetBIOS Frame Control Protocol (NBFCP), NetWare Asynchronous Services Interface (NASI), or X.25 PAD connections.
- Switch-to-switch or router-to-router situations. RADIUS does not provide two-way authentication. RADIUS can be used to authenticate from one device to a non-Cisco device if the non-Cisco device requires authentication.
- Networks using a variety of services. RADIUS generally binds a user to one service model.

Related Topics

RADIUS Overview, on page 969

Information about RADIUS

RADIUS and Switch Access

This section describes how to enable and configure RADIUS. RADIUS provides detailed accounting information and flexible administrative control over the authentication and authorization processes.

The switch supports RADIUS for IPv6. Information is in the "RADIUS Over IPv6" section of the "Implementing ADSL for IPv6" chapter in the *Cisco IOS XE IPv6 Configuration Guide, Release 2*. For information about configuring this feature, see the "Configuring the NAS" section in the "Implementing ADSL for IPv6" chapter in the *Cisco IOS XE IPv6 Configuration Guide, Release 2*.

Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Security Command Reference, Release 12.4* and the *Cisco IOS IPv6 Command Reference*.

Related Topics

Prerequisites for Controlling Switch Access with RADIUS, on page 967

Configuring the Switch for Local Authentication and Authorization, on page 1007

SSH Servers, Integrated Clients, and Supported Versions, on page 1013

RADIUS Overview

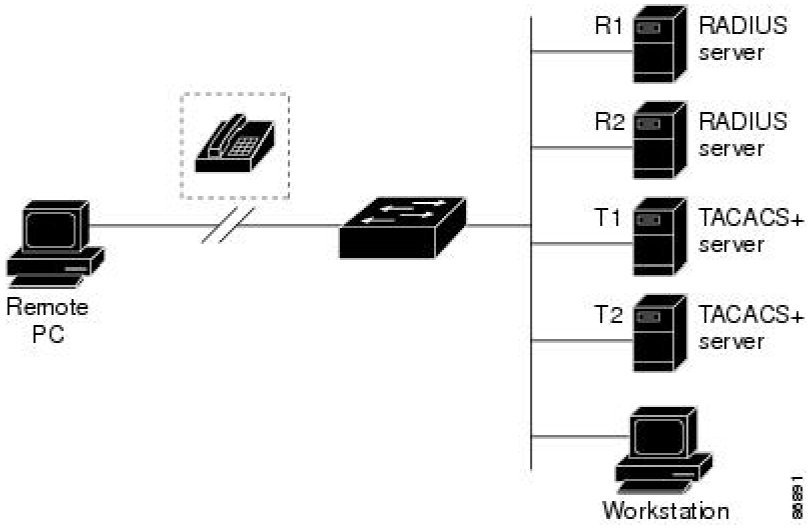
RADIUS is a distributed client/server system that secures networks against unauthorized access. RADIUS clients run on supported Cisco routers and switches. Clients send authentication requests to a central RADIUS server, which contains all user authentication and network service access information.

Use RADIUS in these network environments that require access security:

- Networks with multiple-vendor access servers, each supporting RADIUS. For example, access servers from several vendors use a single RADIUS server-based security database. In an IP-based network with multiple vendors' access servers, dial-in users are authenticated through a RADIUS server that has been customized to work with the Kerberos security system.
- Turnkey network security environments in which applications support the RADIUS protocol, such as in an access environment that uses a *smart card* access control system. In one case, RADIUS has been used with Enigma's security cards to validates users and to grant access to network resources.
- Networks already using RADIUS. You can add a Cisco switch containing a RADIUS client to the network. This might be the first step when you make a transition to a TACACS+ server. See Figure 2: Transitioning from RADIUS to TACACS+ Services below.
- Network in which the user must only access a single service. Using RADIUS, you can control user access to a single host, to a single utility such as Telnet, or to the network through a protocol such as IEEE 802.1x. For more information about this protocol, see Chapter 11, "Configuring IEEE 802.1x Port-Based Authentication."
- Networks that require resource accounting. You can use RADIUS accounting independently of RADIUS authentication or authorization. The RADIUS accounting functions allow data to be sent at the start and

end of services, showing the amount of resources (such as time, packets, bytes, and so forth) used during the session. An Internet service provider might use a freeware-based version of RADIUS access control and accounting software to meet special security and billing needs.

Figure 35: Transitioning from RADIUS to TACACS+ Services



Related Topics

Restrictions for Controlling Switch Access with RADIUS, on page 968

RADIUS Operation

When a user attempts to log in and authenticate to a switch that is access controlled by a RADIUS server, these events occur:

1. The user is prompted to enter a username and password.
2. The username and encrypted password are sent over the network to the RADIUS server.
3. The user receives one of the following responses from the RADIUS server:
   - ACCEPT—The user is authenticated.
   - REJECT—The user is either not authenticated and is prompted to re-enter the username and password, or access is denied.
   - CHALLENGE—A challenge requires additional data from the user.
   - CHALLENGE PASSWORD—A response requests the user to select a new password.

The ACCEPT or REJECT response is bundled with additional data that is used for privileged EXEC or network authorization. The additional data included with the ACCEPT or REJECT packets includes these items:

- Telnet, SSH, rlogin, or privileged EXEC services

- Connection parameters, including the host or client IP address, access list, and user timeouts

Related Topics

Prerequisites for Controlling Switch Access with RADIUS, on page 967

RADIUS Change of Authorization

This section provides an overview of the RADIUS interface including available primitives and how they are used during a Change of Authorization (CoA).

- Change-of-Authorization Requests
- CoA Request Response Code
- CoA Request Commands
- Session Reauthentication
- Stacking Guidelines for Session Termination

A standard RADIUS interface is typically used in a pulled model where the request originates from a network attached device and the response come from the queried servers. Catalyst switches support the RADIUS Change of Authorization (CoA) extensions defined in RFC 5176 that are typically used in a pushed model and allow for the dynamic reconfiguring of sessions from external authentication, authorization, and accounting (AAA) or policy servers.

The switch supports these per-session CoA requests:

- Session reauthentication
- Session termination
- Session termination with port shutdown
- Session termination with port bounce

This feature is integrated with the Cisco Identity Services Engine, and the Cisco Secure Access Control Server (ACS) 5.1.

The RADIUS interface is enabled by default on Catalyst switches. However, some basic configuration is required for the following attributes:

- Security and Password—refer to the "Preventing Unauthorized Access to Your Switch" section in this guide.
- Accounting—refer to the "Starting RADIUS Accounting" section in the Configuring Switch-Based Authentication chapter in this guide.

Change-of-Authorization Requests

Change of Authorization (CoA) requests, as described in RFC 5176, are used in a push model to allow for session identification, host reauthentication, and session termination. The model is comprised of one request (CoA-Request) and two possible response codes:

- CoA acknowledgment (ACK) [CoA-ACK]

- CoA non-acknowledgment (NAK) [CoA-NAK]

The request is initiated from a CoA client (typically a RADIUS or policy server) and directed to the switch that acts as a listener.

RFC 5176 Compliance

The Disconnect Request message, which is also referred to as Packet of Disconnect (POD), is supported by the switch for session termination.

This table shows the IETF attributes are supported for this feature.

Table 116: Supported IETF Attributes

Attribute Number	Attribute Name
24	State
31	Calling-Station-ID
44	Acct-Session-ID
80	Message-Authenticator
101	Error-Cause

This table shows the possible values for the Error-Cause attribute.

Table 117: Error-Cause Values

Value	Explanation
201	Residual Session Context Removed
202	Invalid EAP Packet (Ignored)
401	Unsupported Attribute
402	Missing Attribute
403	NAS Identification Mismatch
404	Invalid Request
405	Unsupported Service
406	Unsupported Extension
407	Invalid Attribute Value
501	Administratively Prohibited

Value	Explanation
502	Request Not Routable (Proxy)
503	Session Context Not Found
504	Session Context Not Removable
505	Other Proxy Processing Error
506	Resources Unavailable
507	Request Initiated
508	Multiple Session Selection Unsupported

CoA Request Response Code

The CoA Request response code can be used to convey a command to the switch.

Related Topics

CoA Request Commands, on page 974

Session Identification

For disconnect and CoA requests targeted at a particular session, the switch locates the session based on one or more of the following attributes:

- Calling-Station-Id (IETF attribute #31 which contains the host MAC address)
- Audit-Session-Id (Cisco VSA)
- Acct-Session-Id (IETF attribute #44)

Unless all session identification attributes included in the CoA message match the session, the switch returns a Disconnect-NAK or CoA-NAK with the "Invalid Attribute Value" error-code attribute.

If more than one session identification attribute is included in the message, all the attributes must match the session or the switch returns a Disconnect- negative acknowledgment (NAK) or CoA-NAK with the error code "Invalid Attribute Value."

The packet format for a CoA Request code as defined in RFC 5176 consists of the fields: Code, Identifier, Length, Authenticator, and Attributes in Type:Length:Value (TLV) format.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Code      |  Identifier   |            Length             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                         Authenticator                         |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Attributes ...
```

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
```

The attributes field is used to carry Cisco vendor-specific attributes (VSAs).

Related Topics

CoA Disconnect-Request, on page 975
CoA Request: Disable Host Port, on page 976
CoA Request: Bounce-Port, on page 976

CoA ACK Response Code

If the authorization state is changed successfully, a positive acknowledgment (ACK) is sent. The attributes returned within CoA ACK will vary based on the CoA Request and are discussed in individual CoA Commands.

CoA NAK Response Code

A negative acknowledgment (NAK) indicates a failure to change the authorization state and can include attributes that indicate the reason for the failure. Use **show** commands to verify a successful CoA.

CoA Request Commands

Table 118: CoA Commands Supported on the Switch

Command [12]	Cisco VSA
Reauthenticate host	Cisco:Avpair="subscriber:command=reauthenticate"
Terminate session	This is a standard disconnect request that does not require a VSA.
Bounce host port	Cisco:Avpair="subscriber:command=bounce-host-port"
Disable host port	Cisco:Avpair="subscriber:command=disable-host-port"

12 All CoA commands must include the session identifier between the switch and the CoA client.

Related Topics

CoA Request Response Code, on page 973

Session Reauthentication

The AAA server typically generates a session reauthentication request when a host with an unknown identity or posture joins the network and is associated with a restricted access authorization profile (such as a guest VLAN). A reauthentication request allows the host to be placed in the appropriate authorization group when its credentials are known.

To initiate session authentication, the AAA server sends a standard CoA-Request message which contains a Cisco VSA in this form: *Cisco:Avpair="subscriber:command=reauthenticate*" and one or more session identification attributes.

The current session state determines the switch response to the message. If the session is currently authenticated by IEEE 802.1x, the switch responds by sending an EAPoL (Extensible Authentication Protocol over Lan) -RequestId message to the server.

If the session is currently authenticated by MAC authentication bypass (MAB), the switch sends an access-request to the server, passing the same identity attributes used for the initial successful authentication.

If session authentication is in progress when the switch receives the command, the switch terminates the process, and restarts the authentication sequence, starting with the method configured to be attempted first.

If the session is not yet authorized, or is authorized via guest VLAN, or critical VLAN, or similar policies, the reauthentication message restarts the access control methods, beginning with the method configured to be attempted first. The current authorization of the session is maintained until the reauthentication leads to a different authorization result.

Session Reauthentication in a Switch Stack

When a switch stack receives a session reauthentication message:

- It checkpoints the need for a re-authentication before returning an acknowledgment (ACK).
- It initiates reauthentication for the appropriate session.
- If authentication completes with either success or failure, the signal that triggered the reauthentication is removed from the stack member.
- If the stack master fails before authentication completes, reauthentication is initiated after stack master switch-over based on the original command (which is subsequently removed).
- If the stack master fails before sending an ACK, the new stack master treats the re-transmitted command as a new command.

Session Termination

There are three types of CoA requests that can trigger session termination. A CoA Disconnect-Request terminates the session, without disabling the host port. This command causes re-initialization of the authenticator state machine for the specified host, but does not restrict that host's access to the network.

To restrict a host's access to the network, use a CoA Request with the Cisco:Avpair="subscriber:command=disable-host-port" VSA. This command is useful when a host is known to be causing problems on the network, and you need to immediately block network access for the host. When you want to restore network access on the port, re-enable it using a non-RADIUS mechanism.

When a device with no supplicant, such as a printer, needs to acquire a new IP address (for example, after a VLAN change), terminate the session on the host port with port-bounce (temporarily disable and then re-enable the port).

CoA Disconnect-Request

This command is a standard Disconnect-Request. Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes. If the session cannot be located, the switch returns a Disconnect-NAK message with the "Session Context Not Found" error-code attribute. If the session is located, the switch terminates the session. After the session has been completely removed, the switch returns a Disconnect-ACK.

If the switch fails-over to a standby switch before returning a Disconnect-ACK to the client, the process is repeated on the new active switch when the request is re-sent from the client. If the session is not found following re-sending, a Disconnect-ACK is sent with the "Session Context Not Found" error-code attribute.

Related Topics

Session Identification, on page 973

CoA Request: Disable Host Port

This command is carried in a standard CoA-Request message that has this new VSA:

Cisco:Avpair="subscriber:command=disable-host-port"

Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes. If the session cannot be located, the switch returns a CoA-NAK message with the "Session Context Not Found" error-code attribute. If the session is located, the switch disables the hosting port and returns a CoA-ACK message.

If the switch fails before returning a CoA-ACK to the client, the process is repeated on the new active switch when the request is re-sent from the client. If the switch fails after returning a CoA-ACK message to the client but before the operation has completed, the operation is restarted on the new active switch.

Note A Disconnect-Request failure following command re-sending could be the result of either a successful session termination before change-over (if the Disconnect-ACK was not sent) or a session termination by other means (for example, a link failure) that occurred after the original command was issued and before the standby switch became active.

Related Topics

Session Identification, on page 973

CoA Request: Bounce-Port

This command is carried in a standard CoA-Request message that contains the following VSA:

Cisco:Avpair="subscriber:command=bounce-host-port"

Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes. If the session cannot be located, the switch returns a CoA-NAK message with the "Session Context Not Found" error-code attribute. If the session is located, the switch disables the hosting port for a period of 10 seconds, re-enables it (port-bounce), and returns a CoA-ACK.

If the switch fails before returning a CoA-ACK to the client, the process is repeated on the new active switch when the request is re-sent from the client. If the switch fails after returning a CoA-ACK message to the client but before the operation has completed, the operation is re-started on the new active switch.

Related Topics

Session Identification, on page 973

Stacking Guidelines for Session Termination

No special handling is required for CoA Disconnect-Request messages in a switch stack.

Stacking Guidelines for CoA-Request Bounce-Port

Because the **bounce-port** command is targeted at a session, not a port, if the session is not found, the command cannot be executed.

When the Auth Manager command handler on the stack master receives a valid **bounce-port** command, it checkpoints the following information before returning a CoA-ACK message:

- the need for a port-bounce
- the port-id (found in the local session context)

The switch initiates a port-bounce (disables the port for 10 seconds, then re-enables it).

If the port-bounce is successful, the signal that triggered the port-bounce is removed from the standby stack master.

If the stack master fails before the port-bounce completes, a port-bounce is initiated after stack master change-over based on the original command (which is subsequently removed).

If the stack master fails before sending a CoA-ACK message, the new stack master treats the re-sent command as a new command.

Stacking Guidelines for CoA-Request Disable-Port

Because the **disable-port** command is targeted at a session, not a port, if the session is not found, the command cannot be executed.

When the Auth Manager command handler on the stack master receives a valid **disable-port** command, it verifies this information before returning a CoA-ACK message:

- the need for a port-disable
- the port-id (found in the local session context)

The switch attempts to disable the port.

If the port-disable operation is successful, the signal that triggered the port-disable is removed from the standby stack master.

If the stack master fails before the port-disable operation completes, the port is disabled after stack master change-over based on the original command (which is subsequently removed).

If the stack master fails before sending a CoA-ACK message, the new stack master treats the re-sent command as a new command.

Default RADIUS Configuration

RADIUS and AAA are disabled by default.

To prevent a lapse in security, you cannot configure RADIUS through a network management application. When enabled, RADIUS can authenticate users accessing the switch through the CLI.

RADIUS Server Host

Switch-to-RADIUS-server communication involves several components:

- Hostname or IP address
- Authentication destination port
- Accounting destination port
- Key string
- Timeout period
- Retransmission value

You identify RADIUS security servers by their hostname or IP address, hostname and specific UDP port numbers, or their IP address and specific UDP port numbers. The combination of the IP address and the UDP port number creates a unique identifier, allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service. This unique identifier enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address.

If two different host entries on the same RADIUS server are configured for the same service—for example, accounting—the second host entry configured acts as a fail-over backup to the first one. Using this example, if the first host entry fails to provide accounting services, the %RADIUS-4-RADIUS_DEAD message appears, and then the switch tries the second host entry configured on the same device for accounting services. (The RADIUS host entries are tried in the order that they are configured.)

A RADIUS server and the switch use a shared secret text string to encrypt passwords and exchange responses. To configure RADIUS to use the AAA security commands, you must specify the host running the RADIUS server daemon and a secret text (key) string that it shares with the switch.

The timeout, retransmission, and encryption key values can be configured globally for all RADIUS servers, on a per-server basis, or in some combination of global and per-server settings.

Related Topics

Identifying the RADIUS Server Host, on page 980

Defining AAA Server Groups, on page 984

Configuring Settings for All RADIUS Servers, on page 988

Configuring RADIUS Login Authentication, on page 982

RADIUS Login Authentication

To configure AAA authentication, you define a named list of authentication methods and then apply that list to various ports. The method list defines the types of authentication to be performed and the sequence in which they are performed; it must be applied to a specific port before any of the defined authentication methods are performed. The only exception is the default method list. The default method list is automatically applied to all ports except those that have a named method list explicitly defined.

A method list describes the sequence and authentication methods to be queried to authenticate a user. You can designate one or more security protocols to be used for authentication, thus ensuring a backup system for authentication in case the initial method fails. The software uses the first method listed to authenticate users; if that method fails to respond, the software selects the next authentication method in the method list. This process continues until there is successful communication with a listed authentication method or until all defined methods are exhausted. If authentication fails at any point in this cycle—meaning that the security server or local username database responds by denying the user access—the authentication process stops, and no other authentication methods are attempted.

Related Topics

Configuring RADIUS Login Authentication, on page 982

AAA Server Groups

You can configure the switch to use AAA server groups to group existing server hosts for authentication. You select a subset of the configured server hosts and use them for a particular service. The server group is used with a global server-host list, which lists the IP addresses of the selected server hosts.

Server groups also can include multiple host entries for the same server if each entry has a unique identifier (the combination of the IP address and UDP port number), allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service. If you configure two different host entries on the same

RADIUS server for the same service, (for example, accounting), the second configured host entry acts as a fail-over backup to the first one.

Related Topics

Defining AAA Server Groups, on page 984

AAA Authorization

AAA authorization limits the services available to a user. When AAA authorization is enabled, the switch uses information retrieved from the user's profile, which is in the local user database or on the security server, to configure the user's session. The user is granted access to a requested service only if the information in the user profile allows it.

Related Topics

Configuring RADIUS Authorization for User Privileged Access and Network Services, on page 986

RADIUS Accounting

The AAA accounting feature tracks the services that users are using and the amount of network resources that they are consuming. When you enable AAA accounting, the switch reports user activity to the RADIUS security server in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the security server. You can then analyze the data for network management, client billing, or auditing.

Related Topics

Starting RADIUS Accounting, on page 987

Vendor-Specific RADIUS Attributes

The Internet Engineering Task Force (IETF) draft standard specifies a method for communicating vendor-specific information between the switch and the RADIUS server by using the vendor-specific attribute (attribute 26). Vendor-specific attributes (VSAs) allow vendors to support their own extended attributes not suitable for general use. The Cisco RADIUS implementation supports one vendor-specific option by using the format recommended in the specification. Cisco's vendor-ID is 9, and the supported option has vendor-type 1, which is named *cisco-avpair*. The value is a string with this format:

```
protocol : attribute sep value *
```

Protocol is a value of the Cisco protocol attribute for a particular type of authorization. *Attribute* and *value* are an appropriate attributevalue (AV) pair defined in the Cisco TACACS+ specification, and *sep* is = for mandatory attributes and is * for optional attributes. The full set of features available for TACACS+ authorization can then be used for RADIUS.

Other vendors have their own unique vendor-IDs, options, and associated VSAs. For more information about vendor-IDs and VSAs, see RFC 2138, "Remote Authentication Dial-In User Service (RADIUS)."

For a complete list of RADIUS attributes or more information about vendor-specific attribute 26, see the "RADIUS Attributes" appendix in the *Cisco IOS Security Configuration Guide*.

Related Topics

Configuring the Switch to Use Vendor-Specific RADIUS Attributes, on page 990

Vendor-Proprietary RADIUS Server Communication

Although an IETF draft standard for RADIUS specifies a method for communicating vendor-proprietary information between the switch and the RADIUS server, some vendors have extended the RADIUS attribute set in a unique way. Cisco IOS software supports a subset of vendor-proprietary RADIUS attributes.

As mentioned earlier, to configure RADIUS (whether vendor-proprietary or IETF draft-compliant), you must specify the host running the RADIUS server daemon and the secret text string it shares with the switch. You specify the RADIUS host and secret text string by using the **radius-server** global configuration commands.

Related Topics

Configuring the Switch for Vendor-Proprietary RADIUS Server Communication, on page 991

How to Configure RADIUS

Identifying the RADIUS Server Host

To apply these settings globally to all RADIUS servers communicating with the switch, use the three unique global configuration commands: **radius-server timeout**, **radius-server retransmit**, and **radius-server key**. To apply these values on a specific RADIUS server, use the **radius-server host** global configuration command.

You can configure the switch to use AAA server groups to group existing server hosts for authentication. For more information, see Related Topics below.

You also need to configure some settings on the RADIUS server. These settings include the IP address of the switch and the key string to be shared by both the server and the switch. For more information, see the RADIUS server documentation.

Before You Begin

If you configure both global and per-server functions (timeout, retransmission, and key commands) on the switch, the per-server timer, retransmission, and key value commands override global timer, retransmission, and key value commands. For information on configuring these settings on all RADIUS servers, see Related Topics below.

SUMMARY STEPS

1. **configure terminal**
2. **radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*] **Example:** `Switch(config)# radius-server host 172.29.36.49 auth-port 1612 key rad1`	Specifies the IP address or hostname of the remote RADIUS server host. • (Optional) For **auth-port** *port-number*, specify the UDP destination port for authentication requests. • (Optional) For **acct-port** *port-number*, specify the UDP destination port for accounting requests. • (Optional) For **timeout** *seconds*, specify the time interval that the switch waits for the RADIUS server to reply before resending. The range is 1 to 1000. This setting overrides the **radius-server timeout** global configuration command setting. If no timeout is set with the **radius-server host** command, the setting of the **radius-server timeout** command is used. • (Optional) For **retransmit** *retries*, specify the number of times a RADIUS request is resent to a server if that server is not responding or responding slowly. The range is 1 to 1000. If no retransmit value is set with the **radius-server host** command, the setting of the **radius-server retransmit** global configuration command is used. • (Optional) For **key** *string*, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. **Note** The key is a text string that must match the encryption key used on the RADIUS server. Always configure the key as the last item in the **radius-server host** command. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key. To configure the switch to recognize more than one host entry associated with a single IP address, enter this command as many times as necessary, making sure that each UDP port number is different. The switch software searches for hosts in the order in which you specify them. Set the timeout, retransmit, and encryption key values to use with the specific RADIUS host.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Related Topics

RADIUS Server Host, on page 977
Defining AAA Server Groups, on page 984
Configuring Settings for All RADIUS Servers, on page 988

Configuring RADIUS Login Authentication

Beginning in privileged EXEC mode, follow these steps to configure RADIUS login authentication:

Before You Begin

To secure the switch for HTTP access by using AAA methods, you must configure the switch with the **ip http authentication aaa** global configuration command. Configuring AAA authentication does not secure the switch for HTTP access by using AAA methods.

For more information about the **ip http authentication** command, see the *Cisco IOS Security Command Reference, Release 12.4*.

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **aaa authentication login** {**default** | *list-name*} *method1* [*method2...*]
4. **line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*]
5. **login authentication** {**default** | *list-name*}
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**aaa new-model** **Example:** `Switch(config)# aaa new-model`	Enables AAA.
Step 3	**aaa authentication login** {**default** | *list-name*} *method1* [*method2...*] **Example:** `Switch(config)# aaa authentication login default`	Creates a login authentication method list. • To create a default list that is used when a named list is *not* specified in the **login authentication** command, use the **default** keyword followed by the methods that are to be used in default situations. The default method list is automatically applied to all ports.

	Command or Action	Purpose
	`local`	• For *list-name*, specify a character string to name the list you are creating. • For *method1...*, specify the actual method the authentication algorithm tries. The additional methods of authentication are used only if the previous method returns an error, not if it fails. Select one of these methods: ◦ *enable*—Use the enable password for authentication. Before you can use this authentication method, you must define an enable password by using the **enable** *password* global configuration command. ◦ *group radius*—Use RADIUS authentication. Before you can use this authentication method, you must configure the RADIUS server. ◦ *line*—Use the line password for authentication. Before you can use this authentication method, you must define a line password. Use the **password** *password* line configuration command. ◦ *local*—Use the local username database for authentication. You must enter username information in the database. Use the **username** *name* **password** global configuration command. ◦ *local-case*—Use a case-sensitive local username database for authentication. You must enter username information in the database by using the **username** *password* global configuration command. ◦ *none*—Do not use any authentication for login.
Step 4	**line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*] **Example:** `Switch(config)# line 1 4`	Enters line configuration mode, and configure the lines to which you want to apply the authentication list.
Step 5	**login authentication** {**default** | *list-name*} **Example:** `Switch(config)# login authentication default`	Applies the authentication list to a line or set of lines. • If you specify **default**, use the default list created with the **aaa authentication login** command. • For *list-name*, specify the list created with the **aaa authentication login** command.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Related Topics

RADIUS Login Authentication, on page 978
RADIUS Server Host, on page 977

Defining AAA Server Groups

You use the **server** group server configuration command to associate a particular server with a defined group server. You can either identify the server by its IP address or identify multiple host instances or entries by using the optional **auth-port** and **acct-port** keywords.

Beginning in privileged EXEC mode, follow these steps to define AAA server groups:

SUMMARY STEPS

1. **configure terminal**
2. **radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]
3. **aaa new-model**
4. **aaa group server radius** *group-name*
5. **server** *ip-address*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*] **Example:** `Switch(config)# radius-server host 172.29.36.49 auth-port 1612 key rad1`	Specifies the IP address or hostname of the remote RADIUS server host. • (Optional) For **auth-port** *port-number*, specify the UDP destination port for authentication requests. • (Optional) For **acct-port** *port-number*, specify the UDP destination port for accounting requests. • (Optional) For **timeout** *seconds*, specify the time interval that the switch waits for the RADIUS server to reply before resending. The range is 1 to 1000. This setting overrides the **radius-server timeout** global configuration command setting. If no timeout is set with the **radius-server host** command, the setting of the **radius-server timeout** command is used. • (Optional) For **retransmit** *retries*, specify the number of times a RADIUS request is resent to a server if that server is not responding or responding slowly. The range is 1 to 1000. If no retransmit value is set with the

	Command or Action	Purpose
		radius-server host command, the setting of the **radius-server retransmit** global configuration command is used. • (Optional) For **key** *string*, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. **Note** The key is a text string that must match the encryption key used on the RADIUS server. Always configure the key as the last item in the **radius-server host** command. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key. To configure the switch to recognize more than one host entry associated with a single IP address, enter this command as many times as necessary, making sure that each UDP port number is different. The switch software searches for hosts in the order in which you specify them. Set the timeout, retransmit, and encryption key values to use with the specific RADIUS host.
Step 3	**aaa new-model** **Example:** `Switch(config)# aaa new-model`	Enables AAA.
Step 4	**aaa group server radius** *group-name* **Example:** `Switch(config)# aaa group server radius group1`	Defines the AAA server-group with a group name. This command puts the switch in a server group configuration mode.
Step 5	**server** *ip-address* **Example:** `Switch(config-sg-radius)# server 172.20.0.1 auth-port 1000 acct-port 1001`	Associates a particular RADIUS server with the defined server group. Repeat this step for each RADIUS server in the AAA server group. Each server in the group must be previously defined in Step 2.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Using Two Different RADIUS Group Servers

In this example, the switch is configured to recognize two different RADIUS group servers (*group1* and *group2*). Group1 has two different host entries on the same RADIUS server configured for the same services. The second host entry acts as a fail-over backup to the first entry.

```
Switch(config)# radius-server host 172.20.0.1 auth-port 1000 acct-port 1001
Switch(config)# radius-server host 172.10.0.1 auth-port 1645 acct-port 1646
Switch(config)# aaa new-model
Switch(config)# aaa group server radius group1
Switch(config-sg-radius)# server 172.20.0.1 auth-port 1000 acct-port 1001
Switch(config-sg-radius)# exit
Switch(config)# aaa group server radius group2
Switch(config-sg-radius)# server 172.20.0.1 auth-port 2000 acct-port 2001
Switch(config-sg-radius)# exit
```

Related Topics

Identifying the RADIUS Server Host, on page 980
RADIUS Server Host, on page 977
AAA Server Groups, on page 978

Configuring RADIUS Authorization for User Privileged Access and Network Services

Note Authorization is bypassed for authenticated users who log in through the CLI even if authorization has been configured.

Beginning in privileged EXEC mode, follow these steps to configure RADIUS authorization for user priviledged access and network services:

SUMMARY STEPS

1. **configure terminal**
2. **aaa authorization network radius**
3. **aaa authorization exec radius**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	**aaa authorization network radius** **Example:** `Switch(config)# aaa authorization network radius`	Configures the switch for user RADIUS authorization for all network-related service requests.
Step 3	**aaa authorization exec radius** **Example:** `Switch(config)# aaa authorization exec radius`	Configures the switch for user RADIUS authorization if the user has privileged EXEC access. The **exec** keyword might return user profile information (such as **autocommand** information).
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

What to Do Next

You can use the **aaa authorization** global configuration command with the **radius** keyword to set parameters that restrict a user's network access to privileged EXEC mode.

The **aaa authorization exec radius local** command sets these authorization parameters:

- Use RADIUS for privileged EXEC access authorization if authentication was performed by using RADIUS.
- Use the local database if authentication was not performed by using RADIUS.

Related Topics

AAA Authorization, on page 979

Starting RADIUS Accounting

Beginning in privileged EXEC mode, follow these steps to start RADIUS accounting:

SUMMARY STEPS

1. **configure terminal**
2. **aaa accounting network start-stop radius**
3. **aaa accounting exec start-stop radius**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**aaa accounting network start-stop radius** **Example:** `Switch(config)# aaa accounting network start-stop radius`	Enables RADIUS accounting for all network-related service requests.
Step 3	**aaa accounting exec start-stop radius** **Example:** `Switch(config)# aaa accounting exec start-stop radius`	Enables RADIUS accounting to send a start-record accounting notice at the beginning of a privileged EXEC process and a stop-record at the end.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

What to Do Next

To establishing a session with a router if the AAA server is unreachable, use the **aaa accounting system guarantee-first** command. This command guarantees system accounting as the first record, which is the default condition. In some situations, users might be prevented from starting a session on the console or terminal connection until after the system reloads, which can take more than 3 minutes.

To establish a console or Telnet session with the router if the AAA server is unreachable when the router reloads, use the **no aaa accounting system guarantee-first** command.

Related Topics

RADIUS Accounting, on page 979

Configuring Settings for All RADIUS Servers

Beginning in privileged EXEC mode, follow these steps to configure settings for all RADIUS servers:

SUMMARY STEPS

1. **configure terminal**
2. **radius-server key** *string*
3. **radius-server retransmit** *retries*
4. **radius-server timeout** *seconds*
5. **radius-server deadtime** *minutes*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**radius-server key** *string* **Example:** `Switch(config)# radius-server key your_server_key`	Specifies the shared secret text string used between the switch and all RADIUS servers. **Note** The key is a text string that must match the encryption key used on the RADIUS server. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.
Step 3	**radius-server retransmit** *retries* **Example:** `Switch(config)# radius-server retransmit 5`	Specifies the number of times the switch sends each RADIUS request to the server before giving up. The default is 3; the range 1 to 1000.
Step 4	**radius-server timeout** *seconds* **Example:** `Switch(config)# radius-server timeout 3`	Specifies the number of seconds a switch waits for a reply to a RADIUS request before resending the request. The default is 5 seconds; the range is 1 to 1000.
Step 5	**radius-server deadtime** *minutes* **Example:** `Switch(config)# radius-server deadtime 0`	When a RADIUS server is not responding to authentication requests, this command specifies a time to stop the request on that server. This avoids the wait for the request to timeout before trying the next configured server. The default is 0; the range is 1 to 1440 minutes.

	Command or Action	Purpose
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Related Topics

Identifying the RADIUS Server Host, on page 980
RADIUS Server Host, on page 977

Configuring the Switch to Use Vendor-Specific RADIUS Attributes

Beginning in privileged EXEC mode, follow these steps to configure the switch to use vendor-specific RADIUS attributes:

SUMMARY STEPS

1. **configure terminal**
2. **radius-server vsa send** [**accounting** | **authentication**]
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**radius-server vsa send** [**accounting** | **authentication**] **Example:** `Switch(config)# radius-server vsa send`	Enables the switch to recognize and use VSAs as defined by RADIUS IETF attribute 26. • (Optional) Use the **accounting** keyword to limit the set of recognized vendor-specific attributes to only accounting attributes. • (Optional) Use the **authentication** keyword to limit the set of recognized vendor-specific attributes to only authentication attributes. If you enter this command without keywords, both accounting and authentication vendor-specific attributes are used.

	Command or Action	Purpose
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Related Topics

Vendor-Specific RADIUS Attributes, on page 979

Configuring the Switch for Vendor-Proprietary RADIUS Server Communication

Beginning in privileged EXEC mode, follow these steps to configure the switch to use vendor-proprietary RADIUS server communication:

SUMMARY STEPS

1. **configure terminal**
2. **radius-server host** {*hostname* | *ip-address*} **non-standard**
3. **radius-server key** *string*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**radius-server host** {*hostname* | *ip-address*} **non-standard** **Example:** `Switch(config)# radius-server host 172.20.30.15 nonstandard`	Specifies the IP address or hostname of the remote RADIUS server host and identifies that it is using a vendor-proprietary implementation of RADIUS.
Step 3	**radius-server key** *string* **Example:** `Switch(config)# radius-server key`	Specifies the shared secret text string used between the switch and the vendor-proprietary RADIUS server. The switch and the RADIUS server use this text string to encrypt passwords and exchange responses.

	Command or Action	Purpose
	`rad124`	**Note** The key is a text string that must match the encryption key used on the RADIUS server. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

What to Do Next

This feature allows access and authentication requests to be evenly across all RADIUS servers in a server group. For more information, see the "RADIUS Server Load Balancing" chapter of the *Cisco IOS Security Configuration Guide, Release 12.4*.

Related Topics

Vendor-Proprietary RADIUS Server Communication, on page 980

Configuring CoA on the Switch

Beginning in privileged EXEC mode, follow these steps to configure CoA on a switch. This procedure is required.

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **aaa server radius dynamic-author**
4. **client** {*ip-address* | *name*} [**vrf** *vrfname*] [**server-key** *string*]
5. **server-key** [**0** | **7**] *string*
6. **port** *port-number*
7. **auth-type** {**any** | **all** | **session-key**}
8. **ignore session-key**
9. **ignore server-key**
10. **authentication command bounce-port ignore**
11. **authentication command disable-port ignore**
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**aaa new-model** **Example:** `Switch(config)# aaa new-model`	Enables AAA.
Step 3	**aaa server radius dynamic-author** **Example:** `Switch(config)# aaa server radius dynamic-author`	Configures the switch as an authentication, authorization, and accounting (AAA) server to facilitate interaction with an external policy server.
Step 4	**client** {*ip-address* | *name*} [**vrf** *vrfname*] [**server-key** *string*]	Enters dynamic authorization local server configuration mode and specifies a RADIUS client from which a device will accept CoA and disconnect requests.
Step 5	**server-key** [**0** | **7**] *string* **Example:** `Switch(config-sg-radius)# server-key your_server_key`	Configures the RADIUS key to be shared between a device and RADIUS clients.
Step 6	**port** *port-number* **Example:** `Switch(config-sg-radius)# port 25`	Specifies the port on which a device listens for RADIUS requests from configured RADIUS clients.
Step 7	**auth-type** {**any** | **all** | **session-key**} **Example:** `Switch(config-sg-radius)# auth-type any`	Specifies the type of authorization the switch uses for RADIUS clients. The client must match all the configured attributes for authorization.
Step 8	**ignore session-key**	(Optional) Configures the switch to ignore the session-key. For more information about the **ignore** command, see the *Cisco IOS Intelligent Services Gateway Command Reference* on Cisco.com.
Step 9	**ignore server-key**	(Optional) Configures the switch to ignore the server-key.

	Command or Action	Purpose
	Example: `Switch(config-sg-radius)# ignore server-key`	For more information about the **ignore** command, see the *Cisco IOS Intelligent Services Gateway Command Reference* on Cisco.com.
Step 10	**authentication command bounce-port ignore** **Example:** `Switch(config-sg-radius)# authentication command bounce-port ignore`	(Optional) Configures the switch to ignore a CoA request to temporarily disable the port hosting a session. The purpose of temporarily disabling the port is to trigger a DHCP renegotiation from the host when a VLAN change occurs and there is no supplicant on the endpoint to detect the change.
Step 11	**authentication command disable-port ignore** **Example:** `Switch(config-sg-radius)# authentication command disable-port ignore`	(Optional) Configures the switch to ignore a nonstandard command requesting that the port hosting a session be administratively shut down. Shutting down the port results in termination of the session. Use standard CLI or SNMP commands to re-enable the port.
Step 12	**end** **Example:** `Switch(config-sg-radius)# end`	Returns to privileged EXEC mode.

Monitoring CoA Functionality

Table 119: Privileged EXEC show Commands

Command	Purpose
show aaa attributes protocol radius	Displays AAA attributes of RADIUS commands.

Table 120: Global Troubleshooting Commands

Command	Purpose
debug radius	Displays information for troubleshooting RADIUS.
debug aaa coa	Displays information for troubleshooting CoA processing.

Command	Purpose
debug aaa pod	Displays information for troubleshooting POD packets.
debug aaa subsys	Displays information for troubleshooting POD packets.
debug cmdhd [**detail** | **error** | **events**]	Displays information for troubleshooting command headers.

For detailed information about the fields in these displays, see the command reference for this release.

Configuration Examples for Controlling Switch Access with RADIUS

Examples: Identifying the RADIUS Server Host

This example shows how to configure one RADIUS server to be used for authentication and another to be used for accounting:

```
Switch(config)# radius-server host 172.29.36.49 auth-port 1612 key rad1
Switch(config)# radius-server host 172.20.36.50 acct-port 1618 key rad2
```

This example shows how to configure *host1* as the RADIUS server and to use the default ports for both authentication and accounting:

```
Switch(config)# radius-server host host1
```

Examples: Configuring the Switch to Use Vendor-Specific RADIUS Attributes

For example, this AV pair activates Cisco's *multiple named ip address pools* feature during IP authorization (during PPP IPCP address assignment):

```
cisco-avpair= ”ip:addr-pool=first“
```

This example shows how to provide a user logging in from a switch with immediate access to privileged EXEC commands:

```
cisco-avpair= ”shell:priv-lvl=15“
```

This example shows how to specify an authorized VLAN in the RADIUS server database:

```
cisco-avpair= ”tunnel-type(#64)=VLAN(13)”
cisco-avpair= ”tunnel-medium-type(#65)=802 media(6)”
cisco-avpair= ”tunnel-private-group-id(#81)=vlanid”
```

This example shows how to apply an input ACL in ASCII format to an interface for the duration of this connection:

```
cisco-avpair= "ip:inacl#1=deny ip 10.10.10.10 0.0.255.255 20.20.20.20 255.255.0.0"
cisco-avpair= "ip:inacl#2=deny ip 10.10.10.10 0.0.255.255 any"
cisco-avpair= "mac:inacl#3=deny any any decnet-iv"
```

This example shows how to apply an output ACL in ASCII format to an interface for the duration of this connection:

```
cisco-avpair= "ip:outacl#2=deny ip 10.10.10.10 0.0.255.255 any"
```

Example: Configuring the Switch for Vendor-Proprietary RADIUS Server Communication

This example shows how to specify a vendor-proprietary RADIUS host and to use a secret key of *rad124* between the switch and the server:

```
Switch(config)# radius-server host 172.20.30.15 nonstandard
Switch(config)# radius-server key rad124
```

Additional References

Related Documents

Related Topic	Document Title
Configuring Identity Control policies and Identity Service templates for Session Aware networking.	Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/en/US/docs/ios-xml/ios/san/configuration/xe-3se/3850/san-xe-3se-3850-book.html
Configuring RADIUS, TACACS+, Secure Shell, 802.1X and AAA.	Securing User Services Configuration Guide Library, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/en/US/docs/ios-xml/ios/security/config_library/xe-3se/3850/secuser-xe-3se-3850-library.html

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

CHAPTER 47

Configuring Kerberos

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for Controlling Switch Access with Kerberos

The following are the prerequisites for controlling switch access with Kerberos.

- So that remote users can authenticate to network services, you must configure the hosts and the KDC in the Kerberos realm to communicate and mutually authenticate users and network services. To do this, you must identify them to each other. You add entries for the hosts to the Kerberos database on the KDC and add KEYTAB files generated by the KDC to all hosts in the Kerberos realm. You also create entries for the users in the KDC database.
- A Kerberos server can be a switch that is configured as a network security server and that can authenticate users by using the Kerberos protocol.

When you add or create entries for the hosts and users, follow these guidelines:

- The Kerberos principal name *must* be in all lowercase characters.
- The Kerberos instance name *must* be in all lowercase characters.
- The Kerberos realm name *must* be in all uppercase characters.

Information about Kerberos

This section provides Kerberos information.

Kerberos and Switch Access

This section describes how to enable and configure the Kerberos security system, which authenticates requests for network resources by using a trusted third party.

For Kerberos configuration examples, see the "Kerberos Configuration Examples" section in the "Security Server Protocols" chapter of the *Cisco IOS Security Configuration Guide, Release 12.4.*.

For complete syntax and usage information for the commands used in this section, see the "Kerberos Commands" section in the "Security Server Protocols" chapter of the *Cisco IOS Security Command Reference, Release 12.4*.

Note In the Kerberos configuration examples and in the *Cisco IOS Security Command Reference, Release 12.4*, the trusted third party can be a switch that supports Kerberos, that is configured as a network security server, and that can authenticate users by using the Kerberos protocol.

Kerberos Overview

Kerberos is a secret-key network authentication protocol, which was developed at the Massachusetts Institute of Technology (MIT). It uses the Data Encryption Standard (DES) cryptographic algorithm for encryption and authentication and authenticates requests for network resources. Kerberos uses the concept of a trusted third party to perform secure verification of users and services. This trusted third party is called the *key distribution center* (KDC).

Kerberos verifies that users are who they claim to be and the network services that they use are what the services claim to be. To do this, a KDC or trusted Kerberos server issues tickets to users. These tickets, which have a limited life span, are stored in user credential caches. The Kerberos server uses the tickets instead of user names and passwords to authenticate users and network services.

Note A Kerberos server can be a switch that is configured as a network security server and that can authenticate users by using the Kerberos protocol.

The Kerberos credential scheme uses a process called *single logon*. This process authenticates a user once and then allows secure authentication (without encrypting another password) wherever that user credential is accepted.

This software release supports Kerberos 5, which allows organizations that are already using Kerberos 5 to use the same Kerberos authentication database on the KDC that they are already using on their other network hosts (such as UNIX servers and PCs).

In this software release, Kerberos supports these network services:

- Telnet
- rlogin
- rsh

This table lists the common Kerberos-related terms and definitions.

Table 121: Kerberos Terms

Term	Definition
Authentication	A process by which a user or service identifies itself to another service. For example, a client can authenticate to a switch or a switch can authenticate to another switch.
Authorization	A means by which the switch identifies what privileges the user has in a network or on the switch and what actions the user can perform.
Credential	A general term that refers to authentication tickets, such as TGTs[13] and service credentials. Kerberos credentials verify the identity of a user or service. If a network service decides to trust the Kerberos server that issued a ticket, it can be used in place of re-entering a username and password. Credentials have a default life span of eight hours.
Instance	An authorization level label for Kerberos principals. Most Kerberos principals are of the form *user@REALM* (for example, smith@EXAMPLE.COM). A Kerberos principal with a Kerberos instance has the form *user/instance@REALM* (for example, smith/admin@EXAMPLE.COM). The Kerberos instance can be used to specify the authorization level for the user if authentication is successful. The server of each network service might implement and enforce the authorization mappings of Kerberos instances but is not required to do so. **Note** The Kerberos principal and instance names *must* be in all lowercase characters. **Note** The Kerberos realm name *must* be in all uppercase characters.

Term	Definition
KDC[14]	Key distribution center that consists of a Kerberos server and database program that is running on a network host.
Kerberized	A term that describes applications and services that have been modified to support the Kerberos credential infrastructure.
Kerberos realm	A domain consisting of users, hosts, and network services that are registered to a Kerberos server. The Kerberos server is trusted to verify the identity of a user or network service to another user or network service. **Note** The Kerberos realm name *must* be in all uppercase characters.
Kerberos server	A daemon that is running on a network host. Users and network services register their identity with the Kerberos server. Network services query the Kerberos server to authenticate to other network services.
KEYTAB[15]	A password that a network service shares with the KDC. In Kerberos 5 and later Kerberos versions, the network service authenticates an encrypted service credential by using the KEYTAB to decrypt it. In Kerberos versions earlier than Kerberos 5, KEYTAB is referred to as SRVTAB[16].
Principal	Also known as a Kerberos identity, this is who you are or what a service is according to the Kerberos server. **Note** The Kerberos principal name *must* be in all lowercase characters.
Service credential	A credential for a network service. When issued from the KDC, this credential is encrypted with the password shared by the network service and the KDC. The password is also shared with the user TGT.
SRVTAB	A password that a network service shares with the KDC. In Kerberos 5 or later Kerberos versions, SRVTAB is referred to as KEYTAB.
TGT	Ticket granting ticket that is a credential that the KDC issues to authenticated users. When users receive a TGT, they can authenticate to network services within the Kerberos realm represented by the KDC.

13 ticket granting ticket
14 key distribution center
15 key table
16 server table

Kerberos Operation

A Kerberos server can be a switch that is configured as a network security server and that can authenticate remote users by using the Kerberos protocol. Although you can customize Kerberos in a number of ways, remote users attempting to access network services must pass through three layers of security before they can access network services.

To authenticate to network services by using a switch as a Kerberos server, remote users must follow these steps:

1. Authenticating to a Boundary Switch, on page 1003
2. Obtaining a TGT from a KDC, on page 1003
3. Authenticating to Network Services, on page 1004

Authenticating to a Boundary Switch

This section describes the first layer of security through which a remote user must pass. The user must first authenticate to the boundary switch. This process then occurs:

1. The user opens an un-Kerberized Telnet connection to the boundary switch.
2. The switch prompts the user for a username and password.
3. The switch requests a TGT from the KDC for this user.
4. The KDC sends an encrypted TGT that includes the user identity to the switch.
5. The switch attempts to decrypt the TGT by using the password that the user entered.
    - If the decryption is successful, the user is authenticated to the switch.
    - If the decryption is not successful, the user repeats Step 2 either by re-entering the username and password (noting if Caps Lock or Num Lock is on or off) or by entering a different username and password.

A remote user who initiates a un-Kerberized Telnet session and authenticates to a boundary switch is inside the firewall, but the user must still authenticate directly to the KDC before getting access to the network services. The user must authenticate to the KDC because the TGT that the KDC issues is stored on the switch and cannot be used for additional authentication until the user logs on to the switch.

Obtaining a TGT from a KDC

This section describes the second layer of security through which a remote user must pass. The user must now authenticate to a KDC and obtain a TGT from the KDC to access network services.

For instructions about how to authenticate to a KDC, see the "Obtaining a TGT from a KDC" section in the "Security Server Protocols" chapter of the *Cisco IOS Security Configuration Guide, Release 12.4*.

Authenticating to Network Services

This section describes the third layer of security through which a remote user must pass. The user with a TGT must now authenticate to the network services in a Kerberos realm.

For instructions about how to authenticate to a network service, see the "Authenticating to Network Services" section in the "Security Server Protocols" chapter of the *Cisco IOS Security Configuration Guide, Release 12.4.*

How to Configure Kerberos

To set up a Kerberos-authenticated server-client system, follow these steps:

- Configure the KDC by using Kerberos commands.
- Configure the switch to use the Kerberos protocol.

For instructions, see the "Kerberos Configuration Task List" section in the "Security Server Protocols" chapter of the *Cisco IOS Security Configuration Guide, Release 12.4.*

Monitoring the Kerberos Configuration

To display the Kerberos configuration, use the show running-config privileged EXEC command.

Additional References

Related Documents

Related Topic	Document Title
Configuring Identity Control policies and Identity Service templates for Session Aware networking.	Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/en/US/docs/ios-xml/ios/san/configuration/xe-3se/3850/san-xe-3se-3850-book.html
Configuring RADIUS, TACACS+, Secure Shell, 802.1X and AAA.	Securing User Services Configuration Guide Library, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/en/US/docs/ios-xml/ios/security/config_library/xe-3se/3850/secuser-xe-3se-3850-library.html

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

CHAPTER 48

Configuring Local Authentication and Authorization

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

How to Configure Local Authentication and Authorization

Configuring the Switch for Local Authentication and Authorization

You can configure AAA to operate without a server by setting the switch to implement AAA in local mode. The switch then handles authentication and authorization. No accounting is available in this configuration.

Note To secure the switch for HTTP access by using AAA methods, you must configure the switch with the **ip http authentication aaa** global configuration command. Configuring AAA authentication does not secure the switch for HTTP access by using AAA methods.

Beginning in privileged EXEC mode, follow these steps to configure AAA to operate without a server by setting the switch to implement AAA in local mode:

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **aaa authentication login default local**
4. **aaa authorization exec local**
5. **aaa authorization network local**
6. **username** *name* [**privilege** *level*] {**password** *encryption-type password*}
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**aaa new-model** **Example:** `Switch(config)# aaa new-model`	Enables AAA.
Step 3	**aaa authentication login default local** **Example:** `Switch(config)# aaa authentication login default local`	Sets the login authentication to use the local username database. The **default** keyword applies the local user database authentication to all ports.
Step 4	**aaa authorization exec local** **Example:** `Switch(config)# aaa authorization exec local`	Configures user AAA authorization, check the local database, and allow the user to run an EXEC shell.
Step 5	**aaa authorization network local** **Example:** `Switch(config)# aaa authorization network local`	Configures user AAA authorization for all network-related service requests.

	Command or Action	Purpose
Step 6	**username** *name* [**privilege** *level*] {**password** *encryption-type password*} **Example:** `Switch(config)# username your_user_name privilege 1 password 7 secret567`	Enters the local database, and establishes a username-based authentication system. Repeat this command for each user. • For *name*, specify the user ID as one word. Spaces and quotation marks are not allowed. • (Optional) For *level*, specify the privilege level the user has after gaining access. The range is 0 to 15. Level 15 gives privileged EXEC mode access. Level 0 gives user EXEC mode access. • For *encryption-type*, enter 0 to specify that an unencrypted password follows. Enter 7 to specify that a hidden password follows. • For *password*, specify the password the user must enter to gain access to the switch. The password must be from 1 to 25 characters, can contain embedded spaces, and must be the last option specified in the **username** command.
Step 7	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Related Topics

Setting Up the Switch to Run SSH, on page 1015
SSH Configuration Guidelines, on page 1013

Monitoring Local Authentication and Authorization

To display Local Authentication and Authorization configuration, use the **show running-config** privileged EXEC command.

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

CHAPTER 49

Configuring Secure Shell (SSH)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for Configuring the Switch for Secure Shell (SSH) and Secure Copy Protocol (SCP)

The following are the prerequisites for configuring the switch for secure shell (SSH):

- For SSH to work, the switch needs an RSA public/private key pair. This is the same with Secure Copy Protocol (SCP), which relies on SSH for its secure transport.
- Before enabling SCP, you must correctly configure SSH, authentication, and authorization on the switch.

- Because SCP relies on SSH for its secure transport, the router must have an Rivest, Shamir, and Adelman (RSA) key pair.
- SCP relies on SSH for security.
- SCP requires that authentication, authorization, and accounting (AAA) authorization be configured so the router can determine whether the user has the correct privilege level.
- A user must have appropriate authorization to use SCP.
- A user who has appropriate authorization can use SCP to copy any file in the Cisco IOS File System (IFS) to and from a switch by using the **copy** command. An authorized administrator can also do this from a workstation.

Related Topics

Secure Copy Protocol Concepts, on page 1014

Restrictions for Configuring the Switch for SSH

The following are restrictions for configuring the switch for secure shell.

- The switch supports Rivest, Shamir, and Adelman (RSA) authentication.
- SSH supports only the execution-shell application.
- The SSH server and the SSH client are supported only on DES (56-bit) and 3DES (168-bit) data encryption software.
- The switch supports the Advanced Encryption Standard (AES) encryption algorithm with a 128-bit key, 192-bit key, or 256-bit key. However, symmetric cipher AES to encrypt the keys is not supported.
- This software release does not support IP Security (IPSec).
- When using SCP, you cannot enter the password into the copy command. You must enter the password when prompted.

Related Topics

Secure Copy Protocol Concepts, on page 1014

Information about SSH

Secure Shell (SSH) is a protocol that provides a secure, remote connection to a device. SSH provides more security for remote connections than Telnet does by providing strong encryption when a device is authenticated. This software release supports SSH Version 1 (SSHv1) and SSH Version 2 (SSHv2).

SSH and Switch Access

For SSH configuration examples, see the “SSH Configuration Examples” section in the “Configuring Secure Shell” section in the “Other Security Features” chapter of the *Cisco IOS Security Configuration Guide, Cisco IOS Release 12.4*.

SSH functions the same in IPv6 as in IPv4. For IPv6, SSH supports IPv6 addresses and enables secure, encrypted connections with remote IPv6 nodes over an IPv6 transport.

Note For complete syntax and usage information for the commands used in this section, see the command reference for this release and the "Secure Shell Commands" section of the "Other Security Features" chapter of the *Cisco IOS Security Command Reference, Release 12.4* and the *Cisco IOS IPv6 Command Reference*.

SSH Servers, Integrated Clients, and Supported Versions

The SSH feature has an SSH server and an SSH integrated client, which are applications that run on the switch. You can use an SSH client to connect to a switch running the SSH server. The SSH server works with the SSH client supported in this release and with non-Cisco SSH clients. The SSH client also works with the SSH server supported in this release and with non-Cisco SSH servers.

The switch supports an SSHv1 or an SSHv2 server.

The switch supports an SSHv1 client.

SSH supports the Data Encryption Standard (DES) encryption algorithm, the Triple DES (3DES) encryption algorithm, and password-based user authentication.

SSH also supports these user authentication methods:

- TACACS+
- RADIUS
- Local authentication and authorization

Related Topics

Configuring the Switch for Local Authentication and Authorization, on page 1007
TACACS+ and Switch Access, on page 953
RADIUS and Switch Access, on page 969

SSH Configuration Guidelines

Follow these guidelines when configuring the switch as an SSH server or SSH client:

- An RSA key pair generated by a SSHv1 server can be used by an SSHv2 server, and the reverse.
- If the SSH server is running on a stack master and the stack master fails, the new stack master uses the RSA key pair generated by the previous stack master.
- If you get CLI error messages after entering the **crypto key generate rsa** global configuration command, an RSA key pair has not been generated. Reconfigure the hostname and domain, and then enter the **crypto key generate rsa** command. For more information, see Related Topics below.
- When generating the RSA key pair, the message No host name specified might appear. If it does, you must configure a hostname by using the **hostname** global configuration command.
- When generating the RSA key pair, the message No domain specified might appear. If it does, you must configure an IP domain name by using the **ip domain-name** global configuration command.

- When configuring the local authentication and authorization authentication method, make sure that AAA is disabled on the console.

Related Topics

Setting Up the Switch to Run SSH, on page 1015
Configuring the Switch for Local Authentication and Authorization, on page 1007

Secure Copy Protocol Overview

The Secure Copy Protocol (SCP) feature provides a secure and authenticated method for copying switch configurations or switch image files. SCP relies on Secure Shell (SSH), an application and a protocol that provides a secure replacement for the Berkeley r-tools.

For SSH to work, the switch needs an RSA public/private key pair. This is the same with SCP, which relies on SSH for its secure transport.

Because SSH also relies on AAA authentication, and SCP relies further on AAA authorization, correct configuration is necessary.

- Before enabling SCP, you must correctly configure SSH, authentication, and authorization on the switch.
- Because SCP relies on SSH for its secure transport, the router must have an Rivest, Shamir, and Adelman (RSA) key pair.

Note When using SCP, you cannot enter the password into the copy command. You must enter the password when prompted.

Secure Copy Protocol Concepts

The Secure Copy Protocol (SCP) feature provides a secure and authenticated method for copying switch configurations or switch image files. SCP relies on Secure Shell (SSH), an application and a protocol that provides a secure replacement for the Berkeley r-tools.

To configure the Secure Copy feature, you should understand the SCP concepts.

The behavior of SCP is similar to that of remote copy (rcp), which comes from the Berkeley r-tools suite, except that SCP relies on SSH for security. SCP also requires that authentication, authorization, and accounting (AAA) authorization be configured so the router can determine whether the user has the correct privilege level.

For information about how to configure and verify SCP, see the “Secure Copy Protocol” section in the *Cisco IOS Security Configuration Guide: Securing User Services, Release 12.4*.

Related Topics

Prerequisites for Configuring the Switch for Secure Shell (SSH) and Secure Copy Protocol (SCP), on page 1011
Restrictions for Configuring the Switch for SSH, on page 1012

How to Configure SSH

Setting Up the Switch to Run SSH

Beginning in privileged EXEC mode, follow these steps to set up your switch to run SSH:

Before You Begin

Configure user authentication for local or remote access. This step is required. For more information, see Related Topics below.

SUMMARY STEPS

1. **configure terminal**
2. **hostname** *hostname*
3. **ip domain-name** *domain_name*
4. **crypto key generate rsa**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**hostname** *hostname* **Example:** `Switch(config)# hostname your_hostname`	Configures a hostname and IP domain name for your switch. **Note** Follow this procedure only if you are configuring the switch as an SSH server.
Step 3	**ip domain-name** *domain_name* **Example:** `Switch(config)# ip domain-name your_domain`	Configures a host domain for your switch.
Step 4	**crypto key generate rsa** **Example:** `Switch(config)# crypto key generate rsa`	Enables the SSH server for local and remote authentication on the switch and generates an RSA key pair. Generating an RSA key pair for the switch automatically enables SSH. We recommend that a minimum modulus size of 1024 bits.

	Command or Action	Purpose
		When you generate RSA keys, you are prompted to enter a modulus length. A longer modulus length might be more secure, but it takes longer to generate and to use. **Note** Follow this procedure only if you are configuring the switch as an SSH server.
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Related Topics

SSH Configuration Guidelines, on page 1013
Configuring the Switch for Local Authentication and Authorization, on page 1007

Configuring the SSH Server

Beginning in privileged EXEC mode, follow these steps to configure the SSH server:

Note This procedure is only required if you are configuring the switch as an SSH server.

SUMMARY STEPS

1. **configure terminal**
2. **ip ssh version** [**1** | **2**]
3. **ip ssh** {**timeout** *seconds* | **authentication-retries** *number*}
4. Use one or both of the following:
   - **line vty***line_number*[*ending_line_number*]
   - **transport input ssh**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip ssh version** [**1** | **2**] **Example:** `Switch(config)# ip ssh version 1`	(Optional) Configures the switch to run SSH Version 1 or SSH Version 2. • **1**—Configure the switch to run SSH Version 1. • **2**—Configure the switch to run SSH Version 2. If you do not enter this command or do not specify a keyword, the SSH server selects the latest SSH version supported by the SSH client. For example, if the SSH client supports SSHv1 and SSHv2, the SSH server selects SSHv2.
Step 3	**ip ssh** {**timeout** *seconds* | **authentication-retries** *number*} **Example:** `Switch(config)# ip ssh timeout 90 authentication-retries 2`	Configures the SSH control parameters: • Specify the time-out value in seconds; the default is 120 seconds. The range is 0 to 120 seconds. This parameter applies to the SSH negotiation phase. After the connection is established, the switch uses the default time-out values of the CLI-based sessions. By default, up to five simultaneous, encrypted SSH connections for multiple CLI-based sessions over the network are available (session 0 to session 4). After the execution shell starts, the CLI-based session time-out value returns to the default of 10 minutes. • Specify the number of times that a client can re-authenticate to the server. The default is 3; the range is 0 to 5. Repeat this step when configuring both parameters.

Step 4	Use one or both of the following:
	• **line vty***line_number*[*ending_line_number*]
	• **transport input ssh**
	
	Example:
	`Switch(config)# line vty 1 10`
	
	or
	
	`Switch(config-line)# transport input ssh`
		(Optional) Configures the virtual terminal line settings.
		• Enters line configuration mode to configure the virtual terminal line settings. For *line_number* and *ending_line_number*, specify a pair of lines. The range is 0 to 15.
		• Specifies that the switch prevent non-SSH Telnet connections. This limits the router to only SSH connections.

	Command or Action	Purpose
Step 5	**end** **Example:** `Switch(config-line)# end`	Returns to privileged EXEC mode.

Monitoring the SSH Configuration and Status

This table displays the SSH server configuration and status.

Table 122: Commands for Displaying the SSH Server Configuration and Status

Command	Purpose
show ip ssh	Shows the version and configuration information for the SSH server.
show ssh	Shows the status of the SSH server.

For more information about these commands, see the “Secure Shell Commands” section in the “Other Security Features” chapter of the *Cisco IOS Security Command Reference* .

Additional References

Related Documents

Related Topic	Document Title
Configuring Identity Control policies and Identity Service templates for Session Aware networking.	Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/en/US/docs/ios-xml/ios/san/configuration/xe-3se/3850/san-xe-3se-3850-book.html
Configuring RADIUS, TACACS+, Secure Shell, 802.1X and AAA.	Securing User Services Configuration Guide Library, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/en/US/docs/ios-xml/ios/security/config_library/xe-3se/3850/secuser-xe-3se-3850-library.html

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

CHAPTER 50

Configuring Secure Socket Layer HTTP

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information about Secure Sockets Layer (SSL) HTTP

This section describes how to configure Secure Sockets Layer (SSL) Version 3.0 support for the HTTP 1.1 server and client. SSL provides server authentication, encryption, and message integrity, as well as HTTP client authentication, to allow secure HTTP communications.

Note SSL evolved into Transport Layer Security (TLS) in 1999, but is still used in this particular context.

On a secure HTTP connection, data to and from an HTTP server is encrypted before being sent over the Internet. HTTP with SSL encryption provides a secure connection to allow such functions as configuring a switch from a Web browser. Cisco's implementation of the secure HTTP server and secure HTTP client uses an implementation of SSL Version 3.0 with application-layer encryption. HTTP over SSL is abbreviated as HTTPS; the URL of a secure connection begins with https:// instead of http://.

The primary role of the HTTP secure server (the switch) is to listen for HTTPS requests on a designated port (the default HTTPS port is 443) and pass the request to the HTTP 1.1 Web server. The HTTP 1.1 server processes requests and passes responses (pages) back to the HTTP secure server, which, in turn, responds to the original request.

The primary role of the HTTP secure client (the web browser) is to respond to Cisco IOS application requests for HTTPS User Agent services, perform HTTPS User Agent services for the application, and pass the response back to the application.

For configuration examples and complete syntax and usage information for the commands used in this section, see the "HTTPS - HTTP Server and Client with SSL 3.0" feature description for Cisco IOS Release 12.2(15)T.

Certificate Authority Trustpoints

Certificate authorities (CAs) manage certificate requests and issue certificates to participating network devices. These services provide centralized security key and certificate management for the participating devices. Specific CA servers are referred to as *trustpoints*.

When a connection attempt is made, the HTTPS server provides a secure connection by issuing a certified X.509v3 certificate, obtained from a specified CA trustpoint, to the client. The client (usually a Web browser), in turn, has a public key that allows it to authenticate the certificate.

For secure HTTP connections, we highly recommend that you configure a CA trustpoint. If a CA trustpoint is not configured for the device running the HTTPS server, the server certifies itself and generates the needed RSA key pair. Because a self-certified (self-signed) certificate does not provide adequate security, the connecting client generates a notification that the certificate is self-certified, and the user has the opportunity to accept or reject the connection. This option is useful for internal network topologies (such as testing).

If you do not configure a CA trustpoint, when you enable a secure HTTP connection, either a temporary or a persistent self-signed certificate for the secure HTTP server (or client) is automatically generated.

- If the switch is not configured with a hostname and a domain name, a temporary self-signed certificate is generated. If the switch reboots, any temporary self-signed certificate is lost, and a new temporary new self-signed certificate is assigned.
- If the switch has been configured with a host and domain name, a persistent self-signed certificate is generated. This certificate remains active if you reboot the switch or if you disable the secure HTTP server so that it will be there the next time you re-enable a secure HTTP connection.

Note The certificate authorities and trustpoints must be configured on each device individually. Copying them from other devices makes them invalid on the switch.

If a self-signed certificate has been generated, this information is included in the output of the **show running-config** privileged EXEC command. This is a partial sample output from that command displaying a self-signed certificate.

```
Switch# show running-config
Building configuration...
```

```
<output truncated>

crypto pki trustpoint TP-self-signed-3080755072
 enrollment selfsigned
 subject-name cn=IOS-Self-Signed-Certificate-3080755072
 revocation-check none
 rsakeypair TP-self-signed-3080755072
!
!
crypto ca certificate chain TP-self-signed-3080755072
 certificate self-signed 01
  3082029F 30820208 A0030201 02020101 300D0609 2A864886 F70D0101 04050030
  59312F30 2D060355 04031326 494F532D 53656C66 2D536967 6E65642D 43657274
  69666963 6174652D 33303830 37353530 37323126 30240609 2A864886 F70D0109
  02161743 45322D33 3535302D 31332E73 756D6D30 342D3335 3530301E 170D3933
  30333031 30303030 35395A17 0D323030 31303130 30303030 305A3059 312F302D

<output truncated>
```

You can remove this self-signed certificate by disabling the secure HTTP server and entering the **no crypto pki trustpoint TP-self-signed-30890755072** global configuration command. If you later re-enable a secure HTTP server, a new self-signed certificate is generated.

Note The values that follow *TP self-signed* depend on the serial number of the device.

You can use an optional command (**ip http secure-client-auth**) to allow the HTTPS server to request an X.509v3 certificate from the client. Authenticating the client provides more security than server authentication by itself.

For additional information on Certificate Authorities, see the “Configuring Certification Authority Interoperability” chapter in the *Cisco IOS Security Configuration Guide, Release 12.4*.

CipherSuites

A CipherSuite specifies the encryption algorithm and the digest algorithm to use on a SSL connection. When connecting to the HTTPS server, the client Web browser offers a list of supported CipherSuites, and the client and server negotiate the best encryption algorithm to use from those on the list that are supported by both. For example, Netscape Communicator 4.76 supports U.S. security with RSA Public Key Cryptography, MD2, MD5, RC2-CBC, RC4, DES-CBC, and DES-EDE3-CBC.

For the best possible encryption, you should use a client browser that supports 128-bit encryption, such as Microsoft Internet Explorer Version 5.5 (or later) or Netscape Communicator Version 4.76 (or later). The SSL_RSA_WITH_DES_CBC_SHA CipherSuite provides less security than the other CipherSuites, as it does not offer 128-bit encryption.

The more secure and more complex CipherSuites require slightly more processing time. This list defines the CipherSuites supported by the switch and ranks them from fastest to slowest in terms of router processing load (speed):

1. SSL_RSA_WITH_DES_CBC_SHA—RSA key exchange (RSA Public Key Cryptography) with DES-CBC for message encryption and SHA for message digest
2. SSL_RSA_WITH_RC4_128_MD5—RSA key exchange with RC4 128-bit encryption and MD5 for message digest
3. SSL_RSA_WITH_RC4_128_SHA—RSA key exchange with RC4 128-bit encryption and SHA for message digest

4 SSL_RSA_WITH_3DES_EDE_CBC_SHA—RSA key exchange with 3DES and DES-EDE3-CBC for message encryption and SHA for message digest

RSA (in conjunction with the specified encryption and digest algorithm combinations) is used for both key generation and authentication on SSL connections. This usage is independent of whether or not a CA trustpoint is configured.

Default SSL Configuration

The standard HTTP server is enabled.

SSL is enabled.

No CA trustpoints are configured.

No self-signed certificates are generated.

SSL Configuration Guidelines

When SSL is used in a switch cluster, the SSL session terminates at the cluster commander. Cluster member switches must run standard HTTP.

Before you configure a CA trustpoint, you should ensure that the system clock is set. If the clock is not set, the certificate is rejected due to an incorrect date.

In a switch stack, the SSL session terminates at the stack master.

Secure HTTP Servers and Clients Overview

On a secure HTTP connection, data to and from an HTTP server is encrypted before being sent over the Internet. HTTP with SSL encryption provides a secure connection to allow such functions as configuring a switch from a Web browser. Cisco's implementation of the secure HTTP server and secure HTTP client uses an implementation of SSL Version 3.0 with application-layer encryption. HTTP over SSL is abbreviated as HTTPS; the URL of a secure connection begins with https:// instead of http://.

The primary role of the HTTP secure server (the switch) is to listen for HTTPS requests on a designated port (the default HTTPS port is 443) and pass the request to the HTTP 1.1 Web server. The HTTP 1.1 server processes requests and passes responses (pages) back to the HTTP secure server, which, in turn, responds to the original request.

The primary role of the HTTP secure client (the web browser) is to respond to Cisco IOS application requests for HTTPS User Agent services, perform HTTPS User Agent services for the application, and pass the response back to the application.

How to Configure Secure HTTP Servers and Clients

Configuring a CA Trustpoint

For secure HTTP connections, we recommend that you configure an official CA trustpoint. A CA trustpoint is more secure than a self-signed certificate.

Beginning in privileged EXEC mode, follow these steps to configure a CA Trustpoint:

SUMMARY STEPS

1. **configure terminal**
2. **hostname** *hostname*
3. **ip domain-name** *domain-name*
4. **crypto key generate rsa**
5. **crypto ca trustpoint** *name*
6. **enrollment url** *url*
7. **enrollment http-proxy** *host-name port-number*
8. **crl query** *url*
9. **primary** *name*
10. **exit**
11. **crypto ca authentication** *name*
12. **crypto ca enroll** *name*
13. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**hostname** *hostname* **Example:** `Switch(config)# hostname your_hostname`	Specifies the hostname of the switch (required only if you have not previously configured a hostname). The hostname is required for security keys and certificates.
Step 3	**ip domain-name** *domain-name* **Example:** `Switch(config)# ip domain-name your_domain`	Specifies the IP domain name of the switch (required only if you have not previously configured an IP domain name). The domain name is required for security keys and certificates.
Step 4	**crypto key generate rsa** **Example:** `Switch(config)# crypto key generate rsa`	(Optional) Generates an RSA key pair. RSA key pairs are required before you can obtain a certificate for the switch. RSA key pairs are generated automatically. You can use this command to regenerate the keys, if needed.

	Command or Action	Purpose
Step 5	**crypto ca trustpoint** *name* **Example:** `Switch(config)# crypto ca trustpoint your_trustpoint`	Specifies a local configuration name for the CA trustpoint and enter CA trustpoint configuration mode.
Step 6	**enrollment url** *url* **Example:** `Switch(ca-trustpoint)# enrollment url http://your_server:80`	Specifies the URL to which the switch should send certificate requests.
Step 7	**enrollment http-proxy** *host-name port-number* **Example:** `Switch(ca-trustpoint)# enrollment http-proxy your_host 49`	(Optional) Configures the switch to obtain certificates from the CA through an HTTP proxy server. • For *host-name* , specify the proxy server used to get the CA. • For *port-number*, specify the port number used to access the CA.
Step 8	**crl query** *url* **Example:** `Switch(ca-trustpoint)# crl query ldap://your_host:49`	Configures the switch to request a certificate revocation list (CRL) to ensure that the certificate of the peer has not been revoked.
Step 9	**primary** *name* **Example:** `Switch(ca-trustpoint)# primary your_trustpoint`	(Optional) Specifies that the trustpoint should be used as the primary (default) trustpoint for CA requests. • For *name*, specify the trustpoint that you just configured.
Step 10	**exit** **Example:** `Switch(ca-trustpoint)# exit`	Exits CA trustpoint configuration mode and return to global configuration mode.
Step 11	**crypto ca authentication** *name* **Example:** `Switch(config)# crypto ca authentication your_trustpoint`	Authenticates the CA by getting the public key of the CA. Use the same name used in Step 5.

	Command or Action	Purpose
Step 12	**crypto ca enroll** *name* **Example:** `Switch(config)# crypto ca enroll your_trustpoint`	Obtains the certificate from the specified CA trustpoint. This command requests a signed certificate for each RSA key pair.
Step 13	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring the Secure HTTP Server

Beginning in privileged EXEC mode, follow these steps to configure a secure HTTP server:

Before You Begin

If you are using a certificate authority for certification, you should use the previous procedure to configure the CA trustpoint on the switch before enabling the HTTP server. If you have not configured a CA trustpoint, a self-signed certificate is generated the first time that you enable the secure HTTP server. After you have configured the server, you can configure options (path, access list to apply, maximum number of connections, or timeout policy) that apply to both standard and secure HTTP servers.

To verify the secure HTTP connection by using a Web browser, enter https://*URL*, where the URL is the IP address or hostname of the server switch. If you configure a port other than the default port, you must also specify the port number after the URL. For example:

```
https://209.165.129:1026
```

or

```
https://host.domain.com:1026
```

SUMMARY STEPS

1. **show ip http server status**
2. **configure terminal**
3. **ip http secure-server**
4. **ip http secure-port** *port-number*
5. **ip http secure-ciphersuite** {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]}
6. **ip http secure-client-auth**
7. **ip http secure-trustpoint** *name*
8. **ip http path** *path-name*
9. **ip http access-class** *access-list-number*
10. **ip http max-connections** *value*
11. **ip http timeout-policy idle** *seconds* **life** *seconds* **requests** *value*
12. **end**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**show ip http server status**
	
	Example:
	`Switch# show ip http server status`
		(Optional) Displays the status of the HTTP server to determine if the secure HTTP server feature is supported in the software. You should see one of these lines in the output:
		
		`HTTP secure server capability: Present`
		
		or
		
		`HTTP secure server capability: Not present`

Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 3	**ip http secure-server** **Example:** `Switch(config)# ip http secure-server`	Enables the HTTPS server if it has been disabled. The HTTPS server is enabled by default.
Step 4	**ip http secure-port** *port-number* **Example:** `Switch(config)# ip http secure-port 443`	(Optional) Specifies the port number to be used for the HTTPS server. The default port number is 443. Valid options are 443 or any number in the range 1025 to 65535.

	Command or Action	Purpose
Step 5	**ip http secure-ciphersuite** {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]} **Example:** `Switch(config)# ip http secure-ciphersuite rc4-128-md5`	(Optional) Specifies the CipherSuites (encryption algorithms) to be used for encryption over the HTTPS connection. If you do not have a reason to specify a particularly CipherSuite, you should allow the server and client to negotiate a CipherSuite that they both support. This is the default.
Step 6	**ip http secure-client-auth** **Example:** `Switch(config)# ip http secure-client-auth`	(Optional) Configures the HTTP server to request an X.509v3 certificate from the client for authentication during the connection process. The default is for the client to request a certificate from the server, but the server does not attempt to authenticate the client.
Step 7	**ip http secure-trustpoint** *name* **Example:** `Switch(config)# ip http secure-trustpoint your_trustpoint`	Specifies the CA trustpoint to use to get an X.509v3 security certificate and to authenticate the client certificate connection. **Note** Use of this command assumes you have already configured a CA trustpoint according to the previous procedure.
Step 8	**ip http path** *path-name* **Example:** `Switch(config)# ip http path /your_server:80`	(Optional) Sets a base HTTP path for HTML files. The path specifies the location of the HTTP server files on the local system (usually located in system flash memory).
Step 9	**ip http access-class** *access-list-number* **Example:** `Switch(config)# ip http access-class 2`	(Optional) Specifies an access list to use to allow access to the HTTP server.
Step 10	**ip http max-connections** *value* **Example:** `Switch(config)# ip http max-connections 4`	(Optional) Sets the maximum number of concurrent connections that are allowed to the HTTP server. The range is 1 to 16; the default value is 5.
Step 11	**ip http timeout-policy idle** *seconds* **life** *seconds* **requests** *value*	(Optional) Specifies how long a connection to the HTTP server can remain open under the defined circumstances:

	Command or Action	Purpose
	Example: `Switch(config)#` **`ip http timeout-policy idle 120 life 240 requests 1`**	• **idle—**the maximum time period when no data is received or response data cannot be sent. The range is 1 to 600 seconds. The default is 180 seconds (3 minutes). • **life—**the maximum time period from the time that the connection is established. The range is 1 to 86400 seconds (24 hours). The default is 180 seconds. • **requests—**the maximum number of requests processed on a persistent connection. The maximum value is 86400. The default is 1.
Step 12	**end** **Example:** `Switch(config)#` **`end`**	Returns to privileged EXEC mode.

Configuring the Secure HTTP Client

Beginning in privileged EXEC mode, follow these steps to configure a secure HTTP client:

Before You Begin

The standard HTTP client and secure HTTP client are always enabled. A certificate authority is required for secure HTTP client certification. This procedure assumes that you have previously configured a CA trustpoint on the switch. If a CA trustpoint is not configured and the remote HTTPS server requires client authentication, connections to the secure HTTP client fail.

SUMMARY STEPS

1. **configure terminal**
2. **ip http client secure-trustpoint** *name*
3. **ip http client secure-ciphersuite** {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]}
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch#` **`configure terminal`**	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	**ip http client secure-trustpoint** *name* **Example:** `Switch(config)# ip http client secure-trustpoint your_trustpoint`	(Optional) Specifies the CA trustpoint to be used if the remote HTTP server requests client authentication. Using this command assumes that you have already configured a CA trustpoint by using the previous procedure. The command is optional if client authentication is not needed or if a primary trustpoint has been configured.
Step 3	**ip http client secure-ciphersuite** {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]} **Example:** `Switch(config)# ip http client secure-ciphersuite rc4-128-md5`	(Optional) Specifies the CipherSuites (encryption algorithms) to be used for encryption over the HTTPS connection. If you do not have a reason to specify a particular CipherSuite, you should allow the server and client to negotiate a CipherSuite that they both support. This is the default.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

How to Configure Secure HTTP Servers and Clients

These sections contain this configuration information:

Monitoring Secure HTTP Server and Client Status

To monitor the SSL secure server and client status, use the privileged EXEC commands in the following table.

Table 123: Commands for Displaying the SSL Secure Server and Client Status

Command	Purpose
show ip http client secure status	Shows the HTTP secure client configuration.
show ip http server secure status	Shows the HTTP secure server configuration.
show running-config	Shows the generated self-signed certificate for secure HTTP connections.

Additional References

Related Documents

Related Topic	Document Title
Configuring Identity Control policies and Identity Service templates for Session Aware networking.	Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/en/US/docs/ios-xml/ios/san/configuration/xe-3se/3850/san-xe-3se-3850-book.html
Configuring RADIUS, TACACS+, Secure Shell, 802.1X and AAA.	Securing User Services Configuration Guide Library, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/en/US/docs/ios-xml/ios/security/config_library/xe-3se/3850/secuser-xe-3se-3850-library.html

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

CHAPTER 51

Configuring IPv4 ACLs

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for Configuring Network Security with ACLs

This section lists the prerequisites for configuring network security with Access Control Lists (ACLs).

- On switches running the LAN base feature set, VLAN maps are not supported.

Restrictions for Configuring Network Security with ACLs

General Network Security

The following are restrictions for configuring network security with ACLs:

- You cannot apply named MAC extended ACLs to Layer 3 interfaces.
- Though visible in the command-line help strings, **appletalk** is not supported as a matching condition for the **deny** and **permit** MAC access-list configuration mode commands.

ACL Filtering

The following are restrictions on ACL filtering:

- If IEEE 802.1Q tunneling is configured on an interface, any IEEE 802.1Q encapsulated IP packets received on the tunnel port can be filtered by MAC ACLs, but not by IP ACLs. This is because the switch does not recognize the protocol inside the IEEE 802.1Q header. This restriction applies to router ACLs, port ACLs, and VLAN maps.

IPv4 ACL Network Interfaces

The following restrictions apply to IPv4 ACLs to network interfaces:

- When controlling access to an interface, you can use a named or numbered ACL.
- If you apply an ACL to a Layer 2 interface that is a member of a VLAN, the Layer 2 (port) ACL takes precedence over an input Layer 3 ACL applied to the VLAN interface or a VLAN map applied to the VLAN.
- If you apply an ACL to a Layer 3 interface and routing is not enabled on the switch, the ACL only filters packets that are intended for the CPU, such as SNMP, Telnet, or web traffic.
- You do not have to enable routing to apply ACLs to Layer 2 interfaces.

Note By default, the router sends Internet Control Message Protocol (ICMP) unreachable messages when a packet is denied by an access group on a Layer 3 interface. These access-group denied packets are not dropped in hardware but are bridged to the switch CPU so that it can generate the ICMP-unreachable message. They do not generate ICMP unreachable messages. ICMP unreachable messages can be disabled on router ACLs with the **no ip unreachables** interface command.

MAC ACLs on a Layer 2 Interface

After you create a MAC ACL, you can apply it to a Layer 2 interface to filter non-IP traffic coming in that interface. When you apply the MAC ACL, consider these guidelines:

- If you apply an ACL to a Layer 2 interface that is a member of a VLAN, the Layer 2 (port) ACL takes precedence over an input Layer 3 ACL applied to the VLAN interface or a VLAN map applied to the VLAN. Incoming packets received on the Layer 2 port are always filtered by the port ACL.

- You can apply no more than one IP access list and one MAC access list to the same Layer 2 interface. The IP access list filters only IP packets, and the MAC access list filters non-IP packets.
- A Layer 2 interface can have only one MAC access list. If you apply a MAC access list to a Layer 2 interface that has a MAC ACL configured, the new ACL replaces the previously configured one.

Note The **mac access-group** interface configuration command is only valid when applied to a physical Layer 2 interface. You cannot use the command on EtherChannel port channels.

Related Topics

Applying an IPv4 ACL to an Interface, on page 1060
IPv4 ACL Interface Considerations, on page 1049
Creating Named MAC Extended ACLs, on page 1062
Applying a MAC ACL to a Layer 2 Interface, on page 1063

Information about Network Security with ACLs

This chapter describes how to configure network security on the switch by using access control lists (ACLs), which in commands and tables are also referred to as access lists.

ACL Overview

Packet filtering can help limit network traffic and restrict network use by certain users or devices. ACLs filter traffic as it passes through a router or switch and permit or deny packets crossing specified interfaces or VLANs. An ACL is a sequential collection of permit and deny conditions that apply to packets. When a packet is received on an interface, the switch compares the fields in the packet against any applied ACLs to verify that the packet has the required permissions to be forwarded, based on the criteria specified in the access lists. One by one, it tests packets against the conditions in an access list. The first match decides whether the switch accepts or rejects the packets. Because the switch stops testing after the first match, the order of conditions in the list is critical. If no conditions match, the switch rejects the packet. If there are no restrictions, the switch forwards the packet; otherwise, the switch drops the packet. The switch can use ACLs on all packets it forwards, including packets bridged within a VLAN.

You configure access lists on a router or Layer 3 switch to provide basic security for your network. If you do not configure ACLs, all packets passing through the switch could be allowed onto all parts of the network. You can use ACLs to control which hosts can access different parts of a network or to decide which types of traffic are forwarded or blocked at router interfaces. For example, you can allow e-mail traffic to be forwarded but not Telnet traffic. ACLs can be configured to block inbound traffic, outbound traffic, or both.

Access Control Entries

An ACL contains an ordered list of access control entries (ACEs). Each ACE specifies *permit* or *deny* and a set of conditions the packet must satisfy in order to match the ACE. The meaning of *permit* or *deny* depends on the context in which the ACL is used.

ACL Supported Types

The switch supports IP ACLs and Ethernet (MAC) ACLs:

- IP ACLs filter IPv4 traffic, including TCP, User Datagram Protocol (UDP), Internet Group Management Protocol (IGMP), and Internet Control Message Protocol (ICMP).
- Ethernet ACLs filter non-IP traffic.

This switch also supports quality of service (QoS) classification ACLs.

Supported ACLs

The switch supports three types of ACLs to filter traffic:

- Port ACLs access-control traffic entering a Layer 2 interface. You can apply only one IP access list and one MAC access list to a Layer 2 interface.
- Router ACLs access-control routed traffic between VLANs and are applied to Layer 3 interfaces in a specific direction (inbound or outbound).
- VLAN ACLs or VLAN maps access-control all packets (bridged and routed). You can use VLAN maps to filter traffic between devices in the same VLAN. VLAN maps are configured to provide access control based on Layer 3 addresses for IPv4. Unsupported protocols are access-controlled through MAC addresses using Ethernet ACEs. After a VLAN map is applied to a VLAN, all packets (routed or bridged) entering the VLAN are checked against the VLAN map. Packets can either enter the VLAN through a switch port or through a routed port after being routed.

ACL Precedence

When Port ACLs, router ACLs, and VLAN maps are configured on the same switch, the filtering precedence, from greatest to least, is port ACL, router ACL, then VLAN map. The following examples describe simple use cases:

- When both an input port ACL and a VLAN map are applied, incoming packets received on ports with a port ACL applied are filtered by the port ACL. Other packets are filtered by the VLAN map
- When an input router ACL and input port ACL exist in a switch virtual interface (SVI), incoming packets received on ports to which a port ACL is applied are filtered by the port ACL. Incoming routed IP packets received on other ports are filtered by the router ACL. Other packets are not filtered.
- When an output router ACL and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are filtered by the port ACL. Outgoing routed IP packets are filtered by the router ACL. Other packets are not filtered.
- When a VLAN map, input router ACL, and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are only filtered by the port ACL. Incoming routed IP packets received on other ports are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.
- When a VLAN map, output router ACL, and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are only filtered by the port ACL. Outgoing routed IP packets are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.

Related Topics

Restrictions for Configuring Network Security with ACLs, on page 1036

Port ACLs

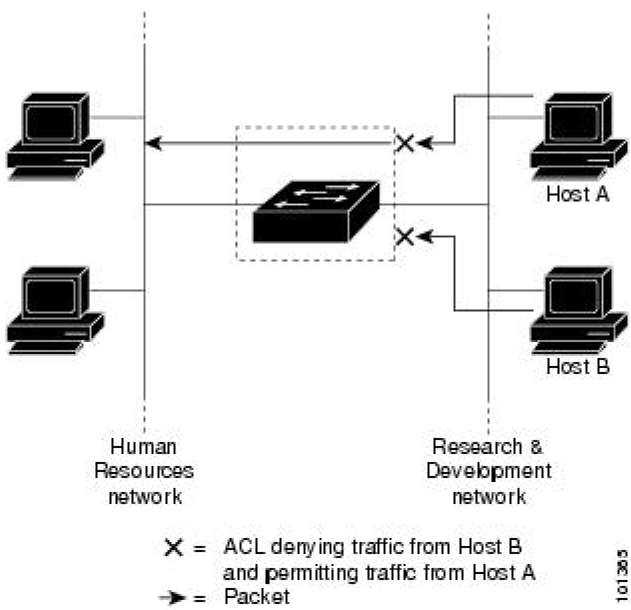
Port ACLs are ACLs that are applied to Layer 2 interfaces on a switch. Port ACLs are supported only on physical interfaces and not on EtherChannel interfaces. Port ACLs can be applied on outbound and inbound interfaces. The following access lists are supported:

- Standard IP access lists using source addresses
- Extended IP access lists using source and destination addresses and optional protocol type information
- MAC extended access lists using source and destination MAC addresses and optional protocol type information

The switch examines ACLs on an interface and permits or denies packet forwarding based on how the packet matches the entries in the ACL. In this way, ACLs control access to a network or to part of a network.

This is an example of using port ACLs to control access to a network when all workstations are in the same VLAN. ACLs applied at the Layer 2 input would allow Host A to access the Human Resources network, but prevent Host B from accessing the same network. Port ACLs can only be applied to Layer 2 interfaces in the inbound direction.

Figure 36: Using ACLs to Control Traffic in a Network



When you apply a port ACL to a trunk port, the ACL filters traffic on all VLANs present on the trunk port. When you apply a port ACL to a port with voice VLAN, the ACL filters traffic on both data and voice VLANs.

With port ACLs, you can filter IP traffic by using IP access lists and non-IP traffic by using MAC addresses. You can filter both IP and non-IP traffic on the same Layer 2 interface by applying both an IP access list and a MAC access list to the interface.

Note You cannot apply more than one IP access list and one MAC access list to a Layer 2 interface. If an IP access list or MAC access list is already configured on a Layer 2 interface and you apply a new IP access list or MAC access list to the interface, the new ACL replaces the previously configured one.

Router ACLs

You can apply router ACLs on switch virtual interfaces (SVIs), which are Layer 3 interfaces to VLANs; on physical Layer 3 interfaces; and on Layer 3 EtherChannel interfaces. You apply router ACLs on interfaces for specific directions (inbound or outbound). You can apply one router ACL in each direction on an interface.

The switch supports these access lists for IPv4 traffic:

- Standard IP access lists use source addresses for matching operations.
- Extended IP access lists use source and destination addresses and optional protocol type information for matching operations.

As with port ACLs, the switch examines ACLs associated with features configured on a given interface. As packets enter the switch on an interface, ACLs associated with all inbound features configured on that interface are examined. After packets are routed and before they are forwarded to the next hop, all ACLs associated with outbound features configured on the egress interface are examined.

ACLs permit or deny packet forwarding based on how the packet matches the entries in the ACL, and can be used to control access to a network or to part of a network.

VLAN Maps

Use VLAN ACLs or VLAN maps to access-control all traffic. You can apply VLAN maps to all packets that are routed into or out of a VLAN or are bridged within a VLAN in the switch or switch stack.

Use VLAN maps for security packet filtering. VLAN maps are not defined by direction (input or output).

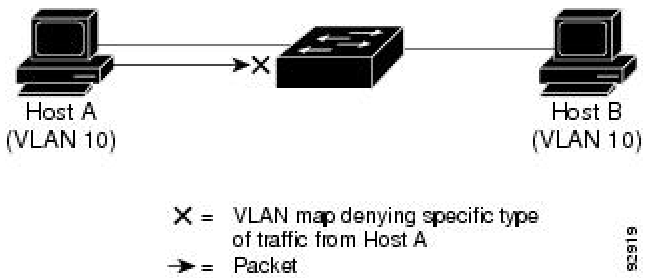
You can configure VLAN maps to match Layer 3 addresses for IPv4 traffic.

All non-IP protocols are access-controlled through MAC addresses and Ethertype using MAC VLAN maps. (IP traffic is not access controlled by MAC VLAN maps.) You can enforce VLAN maps only on packets going through the switch; you cannot enforce VLAN maps on traffic between hosts on a hub or on another switch connected to this switch.

With VLAN maps, forwarding of packets is permitted or denied, based on the action specified in the map.

This shows how a VLAN map is applied to prevent a specific type of traffic from Host A in VLAN 10 from being forwarded. You can apply only one VLAN map to a VLAN.

Figure 37: Using VLAN Maps to Control Traffic



ACEs and Fragmented and Unfragmented Traffic

IP packets can be fragmented as they cross the network. When this happens, only the fragment containing the beginning of the packet contains the Layer 4 information, such as TCP or UDP port numbers, ICMP type and code, and so on. All other fragments are missing this information.

Some access control entries (ACEs) do not check Layer 4 information and therefore can be applied to all packet fragments. ACEs that do test Layer 4 information cannot be applied in the standard manner to most of the fragments in a fragmented IP packet. When the fragment contains no Layer 4 information and the ACE tests some Layer 4 information, the matching rules are modified:

- Permit ACEs that check the Layer 3 information in the fragment (including protocol type, such as TCP, UDP, and so on) are considered to match the fragment regardless of what the missing Layer 4 information might have been.
- Deny ACEs that check Layer 4 information never match a fragment unless the fragment contains Layer 4 information.

Example: ACEs and Fragmented and Unfragmented Traffic

Consider access list 102, configured with these commands, applied to three fragmented packets:

```
Switch(config)# access-list 102 permit tcp any host 10.1.1.1 eq smtp
Switch(config)# access-list 102 deny tcp any host 10.1.1.2 eq telnet
Switch(config)# access-list 102 permit tcp any host 10.1.1.2
Switch(config)# access-list 102 deny tcp any any
```

Note In the first and second ACEs in the examples, the *eq* keyword after the destination address means to test for the TCP-destination-port well-known numbers equaling Simple Mail Transfer Protocol (SMTP) and Telnet, respectively.

- Packet A is a TCP packet from host 10.2.2.2., port 65000, going to host 10.1.1.1 on the SMTP port. If this packet is fragmented, the first fragment matches the first ACE (a permit) as if it were a complete packet because all Layer 4 information is present. The remaining fragments also match the first ACE, even though they do not contain the SMTP port information, because the first ACE only checks Layer

3 information when applied to fragments. The information in this example is that the packet is TCP and that the destination is 10.1.1.1.

- Packet B is from host 10.2.2.2, port 65001, going to host 10.1.1.2 on the Telnet port. If this packet is fragmented, the first fragment matches the second ACE (a deny) because all Layer 3 and Layer 4 information is present. The remaining fragments in the packet do not match the second ACE because they are missing Layer 4 information. Instead, they match the third ACE (a permit).

  Because the first fragment was denied, host 10.1.1.2 cannot reassemble a complete packet, so packet B is effectively denied. However, the later fragments that are permitted will consume bandwidth on the network and resources of host 10.1.1.2 as it tries to reassemble the packet.

- Fragmented packet C is from host 10.2.2.2, port 65001, going to host 10.1.1.3, port ftp. If this packet is fragmented, the first fragment matches the fourth ACE (a deny). All other fragments also match the fourth ACE because that ACE does not check any Layer 4 information and because Layer 3 information in all fragments shows that they are being sent to host 10.1.1.3, and the earlier permit ACEs were checking different hosts.

ACLs and Switch Stacks

ACL support is the same for a switch stack as for a standalone switch. ACL configuration information is propagated to all switches in the stack. All switches in the stack, including the active switch, process the information and program their hardware.

Active Switch and ACL Functions

The active switch performs these ACL functions:

- It processes the ACL configuration and propagates the information to all stack members.
- It distributes the ACL information to any switch that joins the stack.
- If packets must be forwarded by software for any reason (for example, not enough hardware resources), the active switch forwards the packets only after applying ACLs on the packets.
- It programs its hardware with the ACL information it processes.

Stack Member and ACL Functions

Stack members perform these ACL functions:

- They receive the ACL information from the active switch and program their hardware.
- A stack member configured as a standby switch, performs the functions of the active switch in the event the active switch fails.

Active Switch Failure and ACLs

Both the active and standby switches have the ACL information. When the active switch fails, the standby takes over. The new active switch distributes the ACL information to all stack members.

Standard and Extended IPv4 ACLs

This section describes IP ACLs.

An ACL is a sequential collection of permit and deny conditions. One by one, the switch tests packets against the conditions in an access list. The first match determines whether the switch accepts or rejects the packet. Because the switch stops testing after the first match, the order of the conditions is critical. If no conditions match, the switch denies the packet.

The software supports these types of ACLs or access lists for IPv4:

- Standard IP access lists use source addresses for matching operations.
- Extended IP access lists use source and destination addresses for matching operations and optional protocol-type information for finer granularity of control.

IPv4 ACL Switch Unsupported Features

Configuring IPv4 ACLs on the switch is the same as configuring IPv4 ACLs on other Cisco switches and routers.

The switch does not support these Cisco IOS router ACL-related features:

- Non-IP protocol ACLs
- IP accounting
- Reflexive ACLs and dynamic ACLs are not supported.
- ACL logging for port ACLs and VLAN maps

Access List Numbers

The number you use to denote your ACL shows the type of access list that you are creating.

This lists the access-list number and corresponding access list type and shows whether or not they are supported in the switch. The switch supports IPv4 standard and extended access lists, numbers 1 to 199 and 1300 to 2699.

Table 124: Access List Numbers

Access List Number	Type	Supported
1–99	IP standard access list	Yes
100–199	IP extended access list	Yes
200–299	Protocol type-code access list	No
300–399	DECnet access list	No
400–499	XNS standard access list	No
500–599	XNS extended access list	No

Access List Number	Type	Supported
600–699	AppleTalk access list	No
700–799	48-bit MAC address access list	No
800–899	IPX standard access list	No
900–999	IPX extended access list	No
1000–1099	IPX SAP access list	No
1100–1199	Extended 48-bit MAC address access list	No
1200–1299	IPX summary address access list	No
1300–1999	IP standard access list (expanded range)	Yes
2000–2699	IP extended access list (expanded range)	Yes

In addition to numbered standard and extended ACLs, you can also create standard and extended named IP ACLs by using the supported numbers. That is, the name of a standard IP ACL can be 1 to 99; the name of an extended IP ACL can be 100 to 199. The advantage of using named ACLs instead of numbered lists is that you can delete individual entries from a named list.

Numbered Standard IPv4 ACLs

When creating an ACL, remember that, by default, the end of the ACL contains an implicit deny statement for all packets that it did not find a match for before reaching the end. With standard access lists, if you omit the mask from an associated IP host address ACL specification, 0.0.0.0 is assumed to be the mask.

The switch always rewrites the order of standard access lists so that entries with **host** matches and entries with matches having a *don't care* mask of 0.0.0.0 are moved to the top of the list, above any entries with non-zero *don't care* masks. Therefore, in **show** command output and in the configuration file, the ACEs do not necessarily appear in the order in which they were entered.

After creating a numbered standard IPv4 ACL, you can apply it to terminal lines, to interfaces, or to VLANs.

Numbered Extended IPv4 ACLs

Although standard ACLs use only source addresses for matching, you can use extended ACL source and destination addresses for matching operations and optional protocol type information for finer granularity of control. When you are creating ACEs in numbered extended access lists, remember that after you create the ACL, any additions are placed at the end of the list. You cannot reorder the list or selectively add or remove ACEs from a numbered list.

The switch does not support dynamic or reflexive access lists. It also does not support filtering based on the type of service (ToS) minimize-monetary-cost bit.

Some protocols also have specific parameters and keywords that apply to that protocol.

You can define an extended TCP, UDP, ICMP, IGMP, or other IP ACL. The switch also supports these IP protocols:

Note ICMP echo-reply cannot be filtered. All other ICMP codes or types can be filtered.

These IP protocols are supported:

- Authentication Header Protocol (**ahp**)
- Encapsulation Security Payload (**esp**)
- Enhanced Interior Gateway Routing Protocol (**eigrp**)
- generic routing encapsulation (**gre**)
- Internet Control Message Protocol (**icmp**)
- Internet Group Management Protocol (**igmp**)
- any Interior Protocol (**ip**)
- IP in IP tunneling (**ipinip**)
- KA9Q NOS-compatible IP over IP tunneling (**nos**)
- Open Shortest Path First routing (**ospf**)
- Payload Compression Protocol (**pcp**)
- Protocol-Independent Multicast (**pim**)
- Transmission Control Protocol (**tcp**)
- User Datagram Protocol (**udp**)

Named IPv4 ACLs

You can identify IPv4 ACLs with an alphanumeric string (a name) rather than a number. You can use named ACLs to configure more IPv4 access lists in a router than if you were to use numbered access lists. If you identify your access list with a name rather than a number, the mode and command syntax are slightly different. However, not all commands that use IP access lists accept a named access list.

Note The name you give to a standard or extended ACL can also be a number in the supported range of access list numbers. That is, the name of a standard IP ACL can be 1 to 99. The advantage of using named ACLs instead of numbered lists is that you can delete individual entries from a named list.

Consider these guidelines and limitations before configuring named ACLs:

- Not all commands that accept a numbered ACL accept a named ACL. ACLs for packet filters and route filters on interfaces can use a name. VLAN maps also accept a name.
- A standard ACL and an extended ACL cannot have the same name.
- Numbered ACLs are also available.
- You can use standard or extended ACLs (named or numbered) in VLAN maps.

ACL Logging

The switch software can provide logging messages about packets permitted or denied by a standard IP access list. That is, any packet that matches the ACL causes an informational logging message about the packet to be sent to the console. The level of messages logged to the console is controlled by the logging console commands controlling the syslog messages.

Note Because routing is done in hardware and logging is done in software, if a large number of packets match a *permit* or *deny* ACE containing a **log** keyword, the software might not be able to match the hardware processing rate, and not all packets will be logged.

The first packet that triggers the ACL causes a logging message right away, and subsequent packets are collected over 5-minute intervals before they appear or logged. The logging message includes the access list number, whether the packet was permitted or denied, the source IP address of the packet, and the number of packets from that source permitted or denied in the prior 5-minute interval.

Hardware and Software Treatment of IP ACLs

ACL processing is performed in hardware. If the hardware reaches its capacity to store ACL configurations, all packets on that interface are dropped.

Note If an ACL configuration cannot be implemented in hardware due to an out-of-resource condition on a switch or stack member, then only the traffic in that VLAN arriving on that switch is affected.

For router ACLs, other factors can cause packets to be sent to the CPU:

- Using the **log** keyword
- Generating ICMP unreachable messages

When traffic flows are both logged and forwarded, forwarding is done by hardware, but logging must be done by software. Because of the difference in packet handling capacity between hardware and software, if the sum of all flows being logged (both permitted flows and denied flows) is of great enough bandwidth, not all of the packets that are forwarded can be logged.

When you enter the **show ip access-lists** privileged EXEC command, the match count displayed does not account for packets that are access controlled in hardware. Use the **show platform acl counters hardware** privileged EXEC command to obtain some basic hardware ACL statistics for switched and routed packets.

Router ACLs function as follows:

- The hardware controls permit and deny actions of standard and extended ACLs (input and output) for security access control.
- If **log** has not been specified, the flows that match a *deny* statement in a security ACL are dropped by the hardware if *ip unreachables* is disabled. The flows matching a *permit* statement are switched in hardware.
- Adding the **log** keyword to an ACE in a router ACL causes a copy of the packet to be sent to the CPU for logging only. If the ACE is a *permit* statement, the packet is still switched and routed in hardware.

VLAN Map Configuration Guidelines

VLAN maps are the only way to control filtering within a VLAN. VLAN maps have no direction. To filter traffic in a specific direction by using a VLAN map, you need to include an ACL with specific source or destination addresses. If there is a match clause for that type of packet (IP or MAC) in the VLAN map, the default action is to drop the packet if the packet does not match any of the entries within the map. If there is no match clause for that type of packet, the default is to forward the packet.

The following are the VLAN map configuration guidelines:

- If there is no ACL configured to deny traffic on an interface and no VLAN map is configured, all traffic is permitted.
- Each VLAN map consists of a series of entries. The order of entries in an VLAN map is important. A packet that comes into the switch is tested against the first entry in the VLAN map. If it matches, the action specified for that part of the VLAN map is taken. If there is no match, the packet is tested against the next entry in the map.
- If the VLAN map has at least one match clause for the type of packet (IP or MAC) and the packet does not match any of these match clauses, the default is to drop the packet. If there is no match clause for that type of packet in the VLAN map, the default is to forward the packet.
- Logging is not supported for VLAN maps.
- When a switch has an IP access list or MAC access list applied to a Layer 2 interface, and you apply a VLAN map to a VLAN that the port belongs to, the port ACL takes precedence over the VLAN map.
- If a VLAN map configuration cannot be applied in hardware, all packets in that VLAN are dropped.

VLAN Maps with Router ACLs

To access control both bridged and routed traffic, you can use VLAN maps only or a combination of router ACLs and VLAN maps. You can define router ACLs on both input and output routed VLAN interfaces, and you can define a VLAN map to access control the bridged traffic.

If a packet flow matches a VLAN-map deny clause in the ACL, regardless of the router ACL configuration, the packet flow is denied.

Note When you use router ACLs with VLAN maps, packets that require logging on the router ACLs are not logged if they are denied by a VLAN map.

If the VLAN map has a match clause for the type of packet (IP or MAC) and the packet does not match the type, the default is to drop the packet. If there is no match clause in the VLAN map, and no action specified, the packet is forwarded if it does not match any VLAN map entry.

VLAN Maps and Router ACL Configuration Guidelines

These guidelines are for configurations where you need to have an router ACL and a VLAN map on the same VLAN. These guidelines do not apply to configurations where you are mapping router ACLs and VLAN maps on different VLANs.

If you must configure a router ACL and a VLAN map on the same VLAN, use these guidelines for both router ACL and VLAN map configuration:

- You can configure only one VLAN map and one router ACL in each direction (input/output) on a VLAN interface.
- Whenever possible, try to write the ACL with all entries having a single action except for the final, default action of the other type. That is, write the ACL using one of these two forms:

  permit... permit... permit... deny ip any any

  or

  deny... deny... deny... permit ip any any
- To define multiple actions in an ACL (permit, deny), group each action type together to reduce the number of entries.
- Avoid including Layer 4 information in an ACL; adding this information complicates the merging process. The best merge results are obtained if the ACLs are filtered based on IP addresses (source and destination) and not on the full flow (source IP address, destination IP address, protocol, and protocol ports). It is also helpful to use *don't care* bits in the IP address, whenever possible.

  If you need to specify the full-flow mode and the ACL contains both IP ACEs and TCP/UDP/ICMP ACEs with Layer 4 information, put the Layer 4 ACEs at the end of the list. This gives priority to the filtering of traffic based on IP addresses.

VACL Logging

When you configure VACL logging, syslog messages are generated for denied IP packets under these circumstances:

- When the first matching packet is received.
- For any matching packets received within the last 5 minutes.
- If the threshold is reached before the 5-minute interval.

Log messages are generated on a per-flow basis. A flow is defined as packets with the same IP addresses and Layer 4 (UDP or TCP) port numbers. If a flow does not receive any packets in the 5-minute interval, that flow is removed from the cache. When a syslog message is generated, the timer and packet counter are reset.

VACL logging restrictions:

- Only denied IP packets are logged.
- Packets that require logging on the outbound port ACLs are not logged if they are denied by a VACL.

Time Ranges for ACLs

You can selectively apply extended ACLs based on the time of day and the week by using the **time-range** global configuration command. First, define a time-range name and set the times and the dates or the days of the week in the time range. Then enter the time-range name when applying an ACL to set restrictions to the access list. You can use the time range to define when the permit or deny statements in the ACL are in effect, for example, during a specified time period or on specified days of the week. The **time-range** keyword and argument are referenced in the named and numbered extended ACL task tables.

These are some benefits of using time ranges:

- You have more control over permitting or denying a user access to resources, such as an application (identified by an IP address/mask pair and a port number).
- You can control logging messages. ACL entries can be set to log traffic only at certain times of the day. Therefore, you can simply deny access without needing to analyze many logs generated during peak hours.

Time-based access lists trigger CPU activity because the new configuration of the access list must be merged with other features and the combined configuration loaded into the hardware memory. For this reason, you should be careful not to have several access lists configured to take affect in close succession (within a small number of minutes of each other.)

Note The time range relies on the switch system clock; therefore, you need a reliable clock source. We recommend that you use Network Time Protocol (NTP) to synchronize the switch clock.

Related Topics

Configuring Time Ranges for ACLs, on page 1058

IPv4 ACL Interface Considerations

When you apply the **ip access-group** interface configuration command to a Layer 3 interface (an SVI, a Layer 3 EtherChannel, or a routed port), the interface must have been configured with an IP address. Layer 3 access groups filter packets that are routed or are received by Layer 3 processes on the CPU. They do not affect packets bridged within a VLAN.

For inbound ACLs, after receiving a packet, the switch checks the packet against the ACL. If the ACL permits the packet, the switch continues to process the packet. If the ACL rejects the packet, the switch discards the packet.

For outbound ACLs, after receiving and routing a packet to a controlled interface, the switch checks the packet against the ACL. If the ACL permits the packet, the switch sends the packet. If the ACL rejects the packet, the switch discards the packet.

By default, the input interface sends ICMP Unreachable messages whenever a packet is discarded, regardless of whether the packet was discarded because of an ACL on the input interface or because of an ACL on the output interface. ICMP Unreachables are normally limited to no more than one every one-half second per input interface, but this can be changed by using the **ip icmp rate-limit unreachable** global configuration command.

When you apply an undefined ACL to an interface, the switch acts as if the ACL has not been applied to the interface and permits all packets. Remember this behavior if you use undefined ACLs for network security.

Related Topics

Applying an IPv4 ACL to an Interface, on page 1060
Restrictions for Configuring Network Security with ACLs, on page 1036

How to Configure ACLs

Configuring IPv4 ACLs

These are the steps to use IP ACLs on the switch:

SUMMARY STEPS

1. Create an ACL by specifying an access list number or name and the access conditions.
2. Apply the ACL to interfaces or terminal lines. You can also apply standard and extended IP ACLs to VLAN maps.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Create an ACL by specifying an access list number or name and the access conditions.	
Step 2	Apply the ACL to interfaces or terminal lines. You can also apply standard and extended IP ACLs to VLAN maps.	

Creating a Numbered Standard ACL

Beginning in privileged EXEC mode, follow these steps to create a numbered standard ACL:

SUMMARY STEPS

1. **configure terminal**
2. **access-list** *access-list-number* {**deny** \| **permit**} *source source-wildcard* [**log**]
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**access-list** *access-list-number* {**deny** | **permit**} *source source-wildcard* [**log**]	Defines a standard IPv4 access list by using a source address and wildcard. The *access-list-number* is a decimal number from 1 to 99 or 1300 to 1999.

	Command or Action	Purpose
	Example: `Switch(config)# access-list 2 deny your_host`	Enter **deny** or **permit** to specify whether to deny or permit access if conditions are matched. The *source* is the source address of the network or host from which the packet is being sent specified as: • The 32-bit quantity in dotted-decimal format. • The keyword **any** as an abbreviation for *source* and *source-wildcard* of 0.0.0.0 255.255.255.255. You do not need to enter a source-wildcard. • The keyword **host** as an abbreviation for source and *source-wildcard* of *source* 0.0.0.0. (Optional) The *source-wildcard* applies wildcard bits to the source. (Optional) Enter **log** to cause an informational logging message about the packet that matches the entry to be sent to the console. (Optional) Enter **smartlog** to send copies of denied or permitted packets to a NetFlow collector. **Note** Logging is supported only on ACLs attached to Layer 3 interfaces.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Related Topics

Configuring VLAN Maps, on page 1065

Creating a Numbered Extended ACL

Beginning in privileged EXEC mode, follow these steps to create a numbered extended ACL:

SUMMARY STEPS

1. **configure terminal**
2. **access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*] [**tos** tos] [**fragments**] [**log** [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]
3. **access-list** *access-list-number* {**deny** | **permit**} **tcp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [**established**] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log** [**log-input**]] [**time-range** *time-range-name*] [**dscp** *dscp*] [*flag*]
4. **access-list** *access-list-number* {**deny** | **permit**} **udp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log** [**log-input**]] [**time-range** *time-range-name*] [**dscp** *dscp*]
5. **access-list** *access-list-number* {**deny** | **permit**} **icmp** *source source-wildcard destination destination-wildcard* [*icmp-type* | [[*icmp-type icmp-code*] | [*icmp-message*]] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log** [**log-input**]] [**time-range** *time-range-name*] [**dscp** *dscp*]
6. **access-list** *access-list-number* {**deny** | **permit**} **igmp** *source source-wildcard destination destination-wildcard* [*igmp-type*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log** [**log-input**]] [**time-range** *time-range-name*] [**dscp** *dscp*]
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*] [**tos** tos] [**fragments**] [**log** [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*] **Example:** `Switch(config)# access-list 101 permit ip host 10.1.1.2 any precedence 0 tos 0 log`	Defines an extended IPv4 access list and the access conditions. The *access-list-number* is a decimal number from 100 to 199 or 2000 to 2699. Enter **deny** or **permit** to specify whether to deny or permit the packet if conditions are matched. For *protocol*, enter the name or number of an P protocol: **ahp**, **eigrp**, **esp**, **gre**, **icmp**, **igmp**, **igrp**, **ip**, **ipinip**, **nos**, **ospf**, **pcp**, **pim**, **tcp**, or **udp**, or an integer in the range 0 to 255 representing an IP protocol number. To match any Internet protocol (including ICMP, TCP, and UDP), use the keyword **ip**. The *source* is the number of the network or host from which the packet is sent. The *source-wildcard* applies wildcard bits to the source. The *destination* is the network or host number to which the packet is sent. The *destination-wildcard* applies wildcard bits to the destination. Source, source-wildcard, destination, and destination-wildcard can be specified as:

	Command or Action	Purpose
		• The 32-bit quantity in dotted-decimal format. • The keyword **any** for 0.0.0.0 255.255.255.255 (any host). • The keyword **host** for a single host 0.0.0.0. The other keywords are optional and have these meanings: • **precedence—**Enter to match packets with a precedence level specified as a number from 0 to 7 or by name: **routine** (0), **priority** (1), **immediate** (2), **flash** (3), **flash-override** (4), **critical** (5), **internet** (6), **network** (7). • **fragments—**Enter to check non-initial fragments. • **tos—**Enter to match by type of service level, specified by a number from 0 to 15 or a name: **normal** (0), **max-reliability** (2), **max-throughput** (4), **min-delay** (8). • **log—**Enter to create an informational logging message to be sent to the console about the packet that matches the entry or **log-input** to include the input interface in the log entry. • **time-range—**Specify the time-range name. • **dscp—**Enter to match packets with the DSCP value specified by a number from 0 to 63, or use the question mark (?) to see a list of available values. **Note** If you enter a **dscp** value, you cannot enter **tos** or **precedence**. You can enter both a **tos** and a **precedence** value with no **dscp**.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} **tcp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [**established**] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log** [**log-input**]] [**time-range** *time-range-name*] [**dscp** *dscp*] [*flag*] **Example:** `Switch(config)# access-list 101 permit tcp any any eq 500`	Defines an extended TCP access list and the access conditions. The parameters are the same as those described for an extended IPv4 ACL, with these exceptions: (Optional) Enter an *operator* and *port* to compare source (if positioned after *source source-wildcard*) or destination (if positioned after *destination destination-wildcard*) port. Possible operators include **eq** (equal), **gt** (greater than), **lt** (less than), **neq** (not equal), and **range** (inclusive range). Operators require a port number (range requires two port numbers separated by a space). Enter the *port* number as a decimal number (from 0 to 65535) or the name of a TCP port. The other optional keywords have these meanings: • **established—**Enter to match an established connection. This has the same function as matching on the **ack** or **rst** flag. • *flag***—**Enter one of these flags to match by the specified TCP header bits: **ack** (acknowledge), **fin** (finish), **psh** (push), **rst** (reset), **syn** (synchronize), or **urg** (urgent).

	Command or Action	Purpose
Step 4	**access-list** *access-list-number* {**deny** | **permit**} **udp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log** [**log-input**]] [**time-range** *time-range-name*] [**dscp** *dscp*] **Example:** `Switch(config)# access-list 101 permit udp any any eq 100`	(Optional) Defines an extended UDP access list and the access conditions. The UDP parameters are the same as those described for TCP except that the [operator [port]] port number or name must be a UDP port number or name, and the **flag** and **established** keywords are not valid for UDP.
Step 5	**access-list** *access-list-number* {**deny** | **permit**} **icmp** *source source-wildcard destination destination-wildcard* [*icmp-type* | [[*icmp-type icmp-code*] | [*icmp-message*]] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log** [**log-input**]] [**time-range** *time-range-name*] [**dscp** *dscp*] **Example:** `Switch(config)# access-list 101 permit icmp any any 200`	Defines an extended ICMP access list and the access conditions. The ICMP parameters are the same as those described for most IP protocols in an extended IPv4 ACL, with the addition of the ICMP message type and code parameters. These optional keywords have these meanings: • *icmp-type*—Enter to filter by ICMP message type, a number from 0 to 255. • *icmp-code*—Enter to filter ICMP packets that are filtered by the ICMP message code type, a number from 0 to 255. • *icmp-message*—Enter to filter ICMP packets by the ICMP message type name or the ICMP message type and code name.
Step 6	**access-list** *access-list-number* {**deny** | **permit**} **igmp** *source source-wildcard destination destination-wildcard* [*igmp-type*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log** [**log-input**]] [**time-range** *time-range-name*] [**dscp** *dscp*] **Example:** `Switch(config)# access-list 101 permit igmp any any 14`	(Optional) Defines an extended IGMP access list and the access conditions. The IGMP parameters are the same as those described for most IP protocols in an extended IPv4 ACL, with this optional parameter. *igmp-type*—To match IGMP message type, enter a number from 0 to 15, or enter the message name: **dvmrp**, **host-query**, **host-report**, **pim**, or **trace**.
Step 7	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Extended IP ACL with the any Keyword

To use an abbreviation for a source and source wildcard of 0.0.0.0 255.255.255.255 and an abbreviation for a destination and destination wildcard of 0.0.0.0 255.255.255.255 when defining an extended IP ACL, use the **any** keyword in place of source and destination address and wildcard:

```
Switch# configure terminal
Switch(config)# access-list 101 permit ip any any precedence 0 tos 0 fragments
log time-range workhours dscp 10
Switch(config)# end
```

Extended IP ACL with the host Keyword

To use an abbreviation for a source and a source wildcard of source 0.0.0.0 and an abbreviation for a destination and destination wildcard of destination 0.0.0.0 when defining an extended IP ACL, use the **host** keyword in place of the source and destination wildcard or mask.

```
Switch# configure terminal
Switch(config)# access-list 101 permit ip host 10.1.1.2 any
Switch(config)# end
```

Related Topics

Configuring VLAN Maps, on page 1065

Creating Named Standard ACLs

Beginning in privileged EXEC mode, follow these steps to create a standard ACL using names:

SUMMARY STEPS

1. **configure terminal**
2. **ip access-list standard** *name*
3. Use one of the following:
   - **deny** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]
   - **permit** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	**ip access-list standard** *name* **Example:** `Switch(config)# ip access-list standard 20`	Defines a standard IPv4 access list using a name, and enter access-list configuration mode. The name can be a number from 1 to 99.

Step 3	Use one of the following:
	
	• **deny** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]
	
	• **permit** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]]
	
	Example:
	
	`Switch(config-std-nacl)# deny 192.168.0.0 0.0.255.255 255.255.0.0 0.0.255.255`
	
	or
	
	`Switch(config-std-nacl)# permit 10.108.0.0 0.0.0.0 255.255.255.0 0.0.0.0`
		In access-list configuration mode, specify one or more conditions denied or permitted to decide if the packet is forwarded or dropped.
		
		• **host** *source*—A source and source wildcard of *source* 0.0.0.0.
		
		• **any**—A source and source wildcard of 0.0.0.0 255.255.255.255.

Step 4	**end** **Example:** `Switch(config-std-nacl)# end`	Returns to privileged EXEC mode.

Creating Extended Named ACLs

Beginning in privileged EXEC mode, follow these steps to create an extended ACL using names:

SUMMARY STEPS

1. **configure terminal**
2. **ip access-list extended** *name*
3. {**deny** | **permit**} *protocol* {*source* [*source-wildcard*] | **host** *source* | **any**} {*destination* [*destination-wildcard*] | host *destination* | **any**} [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log**] [**time-range** *time-range-name*]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip access-list extended** *name* **Example:** `Switch(config)# ip access-list extended 150`	Defines an extended IPv4 access list using a name, and enter access-list configuration mode. The name can be a number from 100 to 199.
Step 3	{**deny** | **permit**} *protocol* {*source* [*source-wildcard*] | **host** *source* | **any**} {*destination* [*destination-wildcard*] | host *destination* | **any**} [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log**] [**time-range** *time-range-name*] **Example:** `Switch(config-ext-nacl)# permit 0 any any`	In access-list configuration mode, specify the conditions allowed or denied. Use the **log** keyword to get access list logging messages, including violations. • **host** *source*—A source and source wildcard of *source* 0.0.0.0. • **host** *destintation*—A destination and destination wildcard of *destination* 0.0.0.0. • **any**—A source and source wildcard or destination and destination wildcard of 0.0.0.0 255.255.255.255.
Step 4	**end** **Example:** `Switch(config-ext-nacl)# end`	Returns to privileged EXEC mode.

When you are creating extended ACLs, remember that, by default, the end of the ACL contains an implicit deny statement for everything if it did not find a match before reaching the end. For standard ACLs, if you omit the mask from an associated IP host address access list specification, 0.0.0.0 is assumed to be the mask.

After you create an ACL, any additions are placed at the end of the list. You cannot selectively add ACL entries to a specific ACL. However, you can use **no permit** and **no deny** access-list configuration mode commands to remove entries from a named ACL. This example shows how you can delete individual ACEs from the named access list *border-list*:

```
Switch(config)# ip access-list extended border-list
Switch(config-ext-nacl)# no permit ip host 10.1.1.3 any
```

Being able to selectively remove lines from a named ACL is one reason you might use named ACLs instead of numbered ACLs.

What to Do Next

After creating a named ACL, you can apply it to interfaces or to VLANs .

Configuring Time Ranges for ACLs

Beginning in privileged EXEC mode, follow these steps to configure a time-range parameter for an ACL:

SUMMARY STEPS

1. **configure terminal**
2. **time-range** *time-range-name*
3. Use one of the following:
    - **absolute** [**start** *time date*] [**end** *time date*]
    - **periodic** *day-of-the-week hh:mm to* [*day-of-the-week*] *hh:mm*
    - **periodic** {**weekdays** | **weekend** | **daily**} *hh:mm to hh:mm*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**time-range** *time-range-name* **Example:** `Switch(config)# time-range workhours`	Assigns a meaningful name (for example, *workhours*) to the time range to be created, and enter time-range configuration mode. The name cannot contain a space or quotation mark and must begin with a letter.
Step 3	Use one of the following: • **absolute** [**start** *time date*] [**end** *time date*] • **periodic** *day-of-the-week hh:mm to* [*day-of-the-week*] *hh:mm* • **periodic** {**weekdays** | **weekend** | **daily**} *hh:mm to hh:mm* **Example:** `Switch(config-time-range)# absolute start`	Specifies when the function it will be applied to is operational. • You can use only one **absolute** statement in the time range. If you configure more than one absolute statement, only the one configured last is executed. • You can enter multiple **periodic** statements. For example, you could configure different hours for weekdays and weekends. See the example configurations.

	Command or Action	Purpose
	00:00 1 Jan 2006 end 23:59 1 Jan 2006 or `Switch(config-time-range)#` **periodic weekdays 8:00 to 12:00**	
Step 4	**end** **Example:** `Switch(config)#` **end**	Returns to privileged EXEC mode.

What to Do Next

Repeat the steps if you have multiple items that you want in effect at different times.

Related Topics

Time Ranges for ACLs, on page 1048

Applying an IPv4 ACL to a Terminal Line

You can use numbered ACLs to control access to one or more terminal lines. You cannot apply named ACLs to lines. You must set identical restrictions on all the virtual terminal lines because a user can attempt to connect to any of them.

Beginning in privileged EXEC mode, follow these steps to restrict incoming and outgoing connections between a virtual terminal line and the addresses in an ACL:

SUMMARY STEPS

1. **configure terminal**
2. **line** [**console** | **vty**] *line-number*
3. **access-class** *access-list-number* {**in** | **out**}
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**line** [**console** | **vty**] *line-number* **Example:** `Switch(config)# line console 0`	Identifies a specific line to configure, and enter in-line configuration mode. • **console—**Specifies the console terminal line. The console port is DCE. • **vty—**Specifies a virtual terminal for remote console access. The *line-number* is the first line number in a contiguous group that you want to configure when the line type is specified. The range is from 0 to 16.
Step 3	**access-class** *access-list-number* {**in** | **out**} **Example:** `Switch(config-line)# access-class 10 in`	Restricts incoming and outgoing connections between a particular virtual terminal line (into a device) and the addresses in an access list.
Step 4	**end** **Example:** `Switch(config-line)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** `Switch# show running-config`	Displays the access list configuration.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Applying an IPv4 ACL to an Interface

This section describes how to apply IPv4 ACLs to network interfaces.

Beginning in privileged EXEC mode, follow these steps to control access to an interface:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip access-group** {*access-list-number* | *name*} {**in** | **out**}
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Identifies a specific interface for configuration, and enter interface configuration mode. The interface can be a Layer 2 interface (port ACL), or a Layer 3 interface (router ACL).
Step 3	**ip access-group** {*access-list-number* | *name*} {**in** | **out**} **Example:** `Switch(config-if)# ip access-group 2 in`	Controls access to the specified interface. The **out** keyword is not supported for Layer 2 interfaces (port ACLs).
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** `Switch# show running-config`	Displays the access list configuration.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

IPv4 ACL Interface Considerations, on page 1049

Restrictions for Configuring Network Security with ACLs, on page 1036

Creating Named MAC Extended ACLs

You can filter non-IPv4 traffic on a VLAN or on a Layer 2 interface by using MAC addresses and named MAC extended ACLs. The procedure is similar to that of configuring other extended named ACLs.

Beginning in privileged EXEC mode, follow these steps to create a named MAC extended ACL:

SUMMARY STEPS

1. **configure terminal**
2. **mac access-list extended** *name*
3. {**deny** | **permit**} {**any** | **host** *source MAC address* | *source MAC address mask*} {**any** | **host** *destination MAC address* | *destination MAC address mask*} [*type mask* | **lsap** *lsap mask* | **aarp** | **amber** | **dec-spanning** | **decnet-iv** | **diagnostic** | **dsm** | **etype-6000** | **etype-8042** | **lat** | **lavc-sca** | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbios** | **vines-echo** | **vines-ip** | **xns-idp** | 0-*65535*] [**cos** *cos*]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**mac access-list extended** *name* **Example:** `Switch(config)# mac access-list extended mac1`	Defines an extended MAC access list using a name.
Step 3	{**deny** | **permit**} {**any** | **host** *source MAC address* | *source MAC address mask*} {**any** | **host** *destination MAC address* | *destination MAC address mask*} [*type mask* | **lsap** *lsap mask* | **aarp** | **amber** | **dec-spanning** | **decnet-iv** | **diagnostic** | **dsm** | **etype-6000** | **etype-8042** | **lat** | **lavc-sca** | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbios** | **vines-echo** | **vines-ip** | **xns-idp** | 0-*65535*] [**cos** *cos*]	In extended MAC access-list configuration mode, specifies to **permit** or **deny** any source MAC address, a source MAC address with a mask, or a specific **host** source MAC address and **any** destination MAC address, destination MAC address with a mask, or a specific destination MAC address. (Optional) You can also enter these options:

	Command or Action	Purpose

	Example:
	
	`Switch(config-ext-macl)# **deny any any decnet-iv**`
	
	or
	
	`Switch(config-ext-macl)# **permit any any**`
		• *type mask*—An arbitrary EtherType number of a packet with Ethernet II or SNAP encapsulation in decimal, hexadecimal, or octal with optional mask of *don't care* bits applied to the EtherType before testing for a match.
		• **lsap** *lsap mask*—An LSAP number of a packet with IEEE 802.2 encapsulation in decimal, hexadecimal, or octal with optional mask of *don't care* bits.
		• **aarp** | **amber** | **dec-spanning** | **decnet-iv** | **diagnostic** | **dsm** | **etype-6000** | **etype-8042** | **lat** | **lavc-sca** | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbios** | **vines-echo** | **vines-ip** | **xns-idp**—A non-IP protocol.
		• **cos** *cos*—An IEEE 802.1Q cost of service number from 0 to 7 used to set priority.

Step 4	**end** **Example:** `Switch(config-ext-macl)# **end**`	Returns to privileged EXEC mode.

Related Topics

Restrictions for Configuring Network Security with ACLs, on page 1036

Configuring VLAN Maps, on page 1065

Applying a MAC ACL to a Layer 2 Interface

Beginning in privileged EXEC mode, follow these steps to apply a MAC access list to control access to a Layer 2 interface:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **mac access-group** {*name*} {**in** | **out** }
4. **end**
5. **show mac access-group** [**interface** *interface-id*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/2`	Identifies a specific interface, and enter interface configuration mode. The interface must be a physical Layer 2 interface (port ACL).
Step 3	**mac access-group** {*name*} {**in** | **out** } **Example:** `Switch(config-if)# mac access-group mac1 in`	Controls access to the specified interface by using the MAC access list. Port ACLs are supported in the outbound and inbound directions.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show mac access-group** [**interface** *interface-id*] **Example:** `Switch# show mac access-group interface gigabitethernet1/0/2`	Displays the MAC access list applied to the interface or all Layer 2 interfaces.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

After receiving a packet, the switch checks it against the inbound ACL. If the ACL permits it, the switch continues to process the packet. If the ACL rejects the packet, the switch discards it. When you apply an undefined ACL to an interface, the switch acts as if the ACL has not been applied and permits all packets. Remember this behavior if you use undefined ACLs for network security.

Related Topics

Restrictions for Configuring Network Security with ACLs, on page 1036

Configuring VLAN Maps

To create a VLAN map and apply it to one or more VLANs, perform these steps:

Before You Begin

Create the standard or extended IPv4 ACLs or named MAC extended ACLs that you want to apply to the VLAN.

SUMMARY STEPS

1. **vlan access-map** *name* [**number**]
2. **match** {**ip** | **mac**} **address** {*name* | *number*} [*name* | *number*]
3. Enter one of the following commands to specify an IP packet or a non-IP packet (with only a known MAC address) and to match the packet against one or more ACLs (standard or extended):
   - **action** { **forward**}
     ```
     Switch(config-access-map)# action forward
     ```
   - **action** { **drop**}
     ```
     Switch(config-access-map)# action drop
     ```
4. **vlan filter** *mapname* **vlan-list** *list*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**vlan access-map** *name* [**number**] **Example:** `Switch(config)# vlan access-map map_1 20`	Creates a VLAN map, and give it a name and (optionally) a number. The number is the sequence number of the entry within the map. When you create VLAN maps with the same name, numbers are assigned sequentially in increments of 10. When modifying or deleting maps, you can enter the number of the map entry that you want to modify or delete. VLAN maps do not use the specific permit or deny keywords. To deny a packet by using VLAN maps, create an ACL that would match the packet, and set the action to drop. A permit in the ACL counts as a match. A deny in the ACL means no match. Entering this command changes to access-map configuration mode.
Step 2	**match** {**ip** | **mac**} **address** {*name* | *number*} [*name* | *number*] **Example:** `Switch(config-access-map)# match ip`	Match the packet (using either the IP or MAC address) against one or more standard or extended access lists. Note that packets are only matched against access lists of the correct protocol type. IP packets are matched against standard or extended IP access lists. Non-IP packets are only matched against named MAC extended access lists.

	Command or Action	Purpose
	`address ip2`	**Note** If the VLAN map is configured with a match clause for a type of packet (IP or MAC) and the map action is drop, all packets that match the type are dropped. If the VLAN map has no match clause, and the configured action is drop, all IP and Layer 2 packets are dropped.

Step 3	Enter one of the following commands to specify an IP packet or a non-IP packet (with only a known MAC address) and to match the packet against one or more ACLs (standard or extended):
	• **action** { **forward**}
	`Switch(config-access-map)# action forward`
	• **action** { **drop**}
	`Switch(config-access-map)# action drop`
		Sets the action for the map entry.

Step 4	**vlan filter** *mapname* **vlan-list** *list* **Example:** `Switch(config)# vlan filter map 1 vlan-list 20-22`	Applies the VLAN map to one or more VLAN IDs. The list can be a single VLAN ID (22), a consecutive list (10-22), or a string of VLAN IDs (12, 22, 30). Spaces around the comma and hyphen are optional.

Related Topics

Creating a Numbered Standard ACL, on page 1050
Creating a Numbered Extended ACL, on page 1051
Creating Named MAC Extended ACLs, on page 1062
Creating a VLAN Map, on page 1066
Applying a VLAN Map to a VLAN, on page 1068

Creating a VLAN Map

Each VLAN map consists of an ordered series of entries. Beginning in privileged EXEC mode, follow these steps to create, add to, or delete a VLAN map entry:

SUMMARY STEPS

1. **configure terminal**
2. **vlan access-map** *name* [**number**]
3. **match** {**ip** | **mac**} **address** {*name* | *number*} [*name* | *number*]
4. **action** {**drop** | **forward**}
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**vlan access-map** *name* [**number**] **Example:** `Switch(config)# vlan access-map map_1 20`	Creates a VLAN map, and give it a name and (optionally) a number. The number is the sequence number of the entry within the map. When you create VLAN maps with the same name, numbers are assigned sequentially in increments of 10. When modifying or deleting maps, you can enter the number of the map entry that you want to modify or delete. VLAN maps do not use the specific permit or deny keywords. To deny a packet by using VLAN maps, create an ACL that would match the packet, and set the action to drop. A permit in the ACL counts as a match. A deny in the ACL means no match. Entering this command changes to access-map configuration mode.
Step 3	**match** {**ip** | **mac**} **address** {*name* | *number*} [*name* | *number*] **Example:** `Switch(config-access-map)# match ip address ip2`	Match the packet (using either the IP or MAC address) against one or more standard or extended access lists. Note that packets are only matched against access lists of the correct protocol type. IP packets are matched against standard or extended IP access lists. Non-IP packets are only matched against named MAC extended access lists.
Step 4	**action** {**drop** | **forward**} **Example:** `Switch(config-access-map)# action forward`	(Optional) Sets the action for the map entry. The default is to forward.

	Command or Action	Purpose
Step 5	**end** **Example:** `Switch(config-access-map)# end`	Returns to global configuration mode.
Step 6	**show running-config** **Example:** `Switch# show running-config`	Displays the access list configuration.
Step 7	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Configuring VLAN Maps, on page 1065

Applying a VLAN Map to a VLAN

Beginning in privileged EXEC mode, follow these steps to apply a VLAN map to one or more VLANs:

SUMMARY STEPS

1. **configure terminal**
2. **vlan filter** *mapname* **vlan-list** *list*
3. **end**
4. **show running-config**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	**vlan filter** *mapname* **vlan-list** *list* **Example:** `Switch(config)# vlan filter map 1 vlan-list 20-22`	Applies the VLAN map to one or more VLAN IDs. The list can be a single VLAN ID (22), a consecutive list (10-22), or a string of VLAN IDs (12, 22, 30). Spaces around the comma and hyphen are optional.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show running-config** **Example:** `Switch# show running-config`	Displays the access list configuration.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Configuring VLAN Maps, on page 1065

Monitoring IPv4 ACLs

You can monitor IPv4 ACLs by displaying the ACLs that are configured on the switch, and displaying the ACLs that have been applied to interfaces and VLANs.

When you use the **ip access-group** interface configuration command to apply ACLs to a Layer 2 or 3 interface, you can display the access groups on the interface. You can also display the MAC ACLs applied to a Layer 2 interface. You can use the privileged EXEC commands as described in this table to display this information.

Table 125: Commands for Displaying Access Lists and Access Groups

Command	Purpose
show access-lists [*number* | *name*]	Displays the contents of one or all current IP and MAC address access lists or a specific access list (numbered or named).
show ip access-lists [*number* | *name*]	Displays the contents of all current IP access lists or a specific IP access list (numbered or named).

Command	Purpose
show ip interface *interface-id*	Displays detailed configuration and status of an interface. If IP is enabled on the interface and ACLs have been applied by using the **ip access-group** interface configuration command, the access groups are included in the display.
show running-config [**interface** *interface-id*]	Displays the contents of the configuration file for the switch or the specified interface, including all configured MAC and IP access lists and which access groups are applied to an interface.
show mac access-group [**interface** *interface-id*]	Displays MAC access lists applied to all Layer 2 interfaces or the specified Layer 2 interface.

You can also monitor VLAN maps by displaying information about VLAN access maps or VLAN filters. Use the privileged EXEC commands in this table to display VLAN map information.

Table 126: Commands for Displaying VLAN Map Information

Command	Purpose
show vlan access-map [*mapname*]	Displays information about all VLAN access maps or the specified access map.
show vlan filter [**access-map** *name* | **vlan** *vlan-id*]	Displays information about all VLAN filters or about a specified VLAN or VLAN access map.

Configuration Examples for ACLs

Examples: Using Time Ranges with ACLs

This example shows how to verify after you configure time ranges for *workhours* and to configure January 1, 2006, as a company holiday.

```
Switch# show time-range
time-range entry: new_year_day_2003 (inactive)
   absolute start 00:00 01 January 2006 end 23:59 01 January 2006
time-range entry: workhours (inactive)
   periodic weekdays 8:00 to 12:00
   periodic weekdays 13:00 to 17:00
```

To apply a time range, enter the time-range name in an extended ACL that can implement time ranges. This example shows how to create and verify extended access list 188 that denies TCP traffic from any source to any destination during the defined holiday times and permits all TCP traffic during work hours.

```
Switch(config)# access-list 188 deny tcp any any time-range new_year_day_2006
Switch(config)# access-list 188 permit tcp any any time-range workhours
```

```
Switch(config)# end
Switch# show access-lists
Extended IP access list 188
   10 deny tcp any any time-range new_year_day_2006 (inactive)
   20 permit tcp any any time-range workhours (inactive)
```

This example uses named ACLs to permit and deny the same traffic.

```
Switch(config)# ip access-list extended deny_access
Switch(config-ext-nacl)# deny tcp any any time-range new_year_day_2006
Switch(config-ext-nacl)# exit
Switch(config)# ip access-list extended may_access
Switch(config-ext-nacl)# permit tcp any any time-range workhours
Switch(config-ext-nacl)# end
Switch# show ip access-lists
Extended IP access list lpip_default
    10 permit ip any any
Extended IP access list deny_access
    10 deny tcp any any time-range new_year_day_2006 (inactive)
Extended IP access list may_access
    10 permit tcp any any time-range workhours (inactive)
```

Examples: Including Comments in ACLs

You can use the **remark** keyword to include comments (remarks) about entries in any IP standard or extended ACL. The remarks make the ACL easier for you to understand and scan. Each remark line is limited to 100 characters.

The remark can go before or after a permit or deny statement. You should be consistent about where you put the remark so that it is clear which remark describes which permit or deny statement. For example, it would be confusing to have some remarks before the associated permit or deny statements and some remarks after the associated statements.

To include a comment for IP numbered standard or extended ACLs, use the **access-list** *access-list number* **remark** *remark* global configuration command. To remove the remark, use the **no** form of this command.

In this example, the workstation that belongs to Jones is allowed access, and the workstation that belongs to Smith is not allowed access:

```
Switch(config)# access-list 1 remark Permit only Jones workstation through
Switch(config)# access-list 1 permit 171.69.2.88
Switch(config)# access-list 1 remark Do not allow Smith through
Switch(config)# access-list 1 deny 171.69.3.13
```

For an entry in a named IP ACL, use the **remark** access-list configuration command. To remove the remark, use the **no** form of this command.

In this example, the Jones subnet is not allowed to use outbound Telnet:

```
Switch(config)# ip access-list extended telnetting
Switch(config-ext-nacl)# remark Do not allow Jones subnet to telnet out
Switch(config-ext-nacl)# deny tcp host 171.69.2.88 any eq telnet
```

IPv4 ACL Configuration Examples

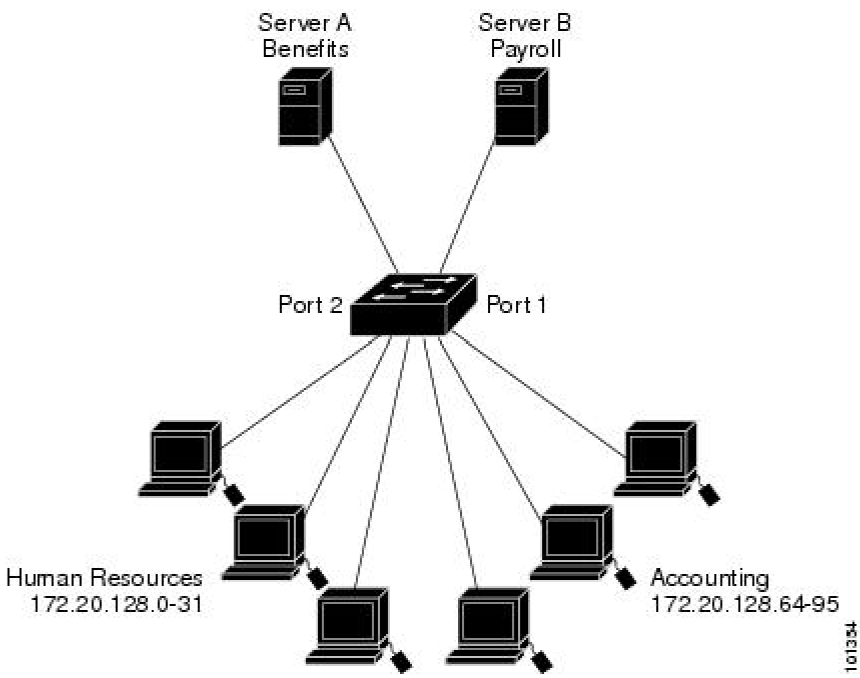
This section provides examples of configuring and applying IPv4 ACLs. For detailed information about compiling ACLs, see the *Cisco IOS Security Configuration Guide, Release 12.4* and to the Configuring IP

Services" section in the "IP Addressing and Services" chapter of the *Cisco IOS IP Configuration Guide, Release 12.4.*

ACLs in a Small Networked Office

This shows a small networked office environment with routed Port 2 connected to Server A, containing benefits and other information that all employees can access, and routed Port 1 connected to Server B, containing confidential payroll data. All users can access Server A, but Server B has restricted access.

Figure 38: Using Router ACLs to Control Traffic



Use router ACLs to do this in one of two ways:

- Create a standard ACL, and filter traffic coming to the server from Port 1.
- Create an extended ACL, and filter traffic coming from the server into Port 1.

Examples: ACLs in a Small Networked Office

This example uses a standard ACL to filter traffic coming into Server B from a port, permitting traffic only from Accounting's source addresses 172.20.128.64 to 172.20.128.95. The ACL is applied to traffic coming out of routed Port 1 from the specified source address.

```
Switch(config)# access-list 6 permit 172.20.128.64 0.0.0.31
Switch(config)# end
Switch# how access-lists
Standard IP access list 6
    10 permit 172.20.128.64, wildcard bits 0.0.0.31
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip access-group 6 out
```

This example uses an extended ACL to filter traffic coming from Server B into a port, permitting traffic from any source address (in this case Server B) to only the Accounting destination addresses 172.20.128.64 to 172.20.128.95. The ACL is applied to traffic going into routed Port 1, permitting it to go only to the specified destination addresses. Note that with extended ACLs, you must enter the protocol (IP) before the source and destination information.

```
Switch(config)# access-list 106 permit ip any 172.20.128.64 0.0.0.31
Switch(config)# end
Switch# show access-lists
Extended IP access list 106
    10 permit ip any 172.20.128.64 0.0.0.31
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip access-group 106 in
```

Example: Numbered ACLs

In this example, network 36.0.0.0 is a Class A network whose second octet specifies a subnet; that is, its subnet mask is 255.255.0.0. The third and fourth octets of a network 36.0.0.0 address specify a particular host. Using access list 2, the switch accepts one address on subnet 48 and reject all others on that subnet. The last line of the list shows that the switch accepts addresses on all other network 36.0.0.0 subnets. The ACL is applied to packets entering a port.

```
Switch(config)# access-list 2 permit 36.48.0.3
Switch(config)# access-list 2 deny 36.48.0.0 0.0.255.255
Switch(config)# access-list 2 permit 36.0.0.0 0.255.255.255
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# ip access-group 2 in
```

Examples: Extended ACLs

In this example, the first line permits any incoming TCP connections with destination ports greater than 1023. The second line permits incoming TCP connections to the Simple Mail Transfer Protocol (SMTP) port of host 128.88.1.2. The third line permits incoming ICMP messages for error feedback.

```
Switch(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 gt 1023
Switch(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Switch(config)# access-list 102 permit icmp any any
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# ip access-group 102 in
```

In this example, suppose that you have a network connected to the Internet, and you want any host on the network to be able to form TCP connections to any host on the Internet. However, you do not want IP hosts to be able to form TCP connections to hosts on your network, except to the mail (SMTP) port of a dedicated mail host.

SMTP uses TCP port 25 on one end of the connection and a random port number on the other end. The same port numbers are used throughout the life of the connection. Mail packets coming in from the Internet have a destination port of 25. Outbound packets have the port numbers reversed. Because the secure system of the network always accepts mail connections on port 25, the incoming and outgoing services are separately controlled. The ACL must be configured as an input ACL on the outbound interface and an output ACL on the inbound interface.

```
Switch(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 eq 23
Switch(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 eq 25
Switch(config)# interface gigabitethernet1/0/1
```

```
Switch(config-if)# ip access-group 102 in
```

In this example, the network is a Class B network with the address 128.88.0.0, and the mail host address is 128.88.1.2. The **established** keyword is used only for the TCP to show an established connection. A match occurs if the TCP datagram has the ACK or RST bits set, which show that the packet belongs to an existing connection. Gigabit Ethernet interface 1 on stack member 1 is the interface that connects the router to the Internet.

```
Switch(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 established
Switch(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip access-group 102 in
```

Examples: Named ACLs

This example creates a standard ACL named *internet_filter* and an extended ACL named *marketing_group*. The *internet_filter* ACL allows all traffic from the source address 1.2.3.4.

```
Switch(config)# ip access-list standard Internet_filter
Switch(config-ext-nacl)# permit 1.2.3.4
Switch(config-ext-nacl)# exit
```

The *marketing_group* ACL allows any TCP Telnet traffic to the destination address and wildcard 171.69.0.0 0.0.255.255 and denies any other TCP traffic. It permits ICMP traffic, denies UDP traffic from any source to the destination address range 171.69.0.0 through 179.69.255.255 with a destination port less than 1024, denies any other IP traffic, and provides a log of the result.

```
Switch(config)# ip access-list extended marketing_group
Switch(config-ext-nacl)# permit tcp any 171.69.0.0 0.0.255.255 eq telnet
Switch(config-ext-nacl)# deny tcp any any
Switch(config-ext-nacl)# permit icmp any any
Switch(config-ext-nacl)# deny udp any 171.69.0.0 0.0.255.255 lt 1024
Switch(config-ext-nacl)# deny ip any any log
Switch(config-ext-nacl)# exit
```

The *Internet_filter* ACL is applied to outgoing traffic and the *marketing_group* ACL is applied to incoming traffic on a Layer 3 port.

```
Switch(config)# interface gigabitethernet3/0/2
Switch(config-if)# no switchport
Switch(config-if)# ip address 2.0.5.1 255.255.255.0
Switch(config-if)# ip access-group Internet_filter out
Switch(config-if)# ip access-group marketing_group in
```

Examples: Time Range Applied to an IP ACL

This example denies HTTP traffic on IP on Monday through Friday between the hours of 8:00 a.m. and 6:00 p.m (18:00). The example allows UDP traffic only on Saturday and Sunday from noon to 8:00 p.m. (20:00).

```
Switch(config)# time-range no-http
Switch(config)# periodic weekdays 8:00 to 18:00
!
Switch(config)# time-range udp-yes
Switch(config)# periodic weekend 12:00 to 20:00
!
Switch(config)# ip access-list extended strict
Switch(config-ext-nacl)# deny tcp any any eq www time-range no-http
```

```
Switch(config-ext-nacl)# permit udp any any time-range udp-yes
!
Switch(config-ext-nacl)# exit
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# ip access-group strict in
```

Examples: Commented IP ACL Entries

In this example of a numbered ACL, the workstation that belongs to Jones is allowed access, and the workstation that belongs to Smith is not allowed access:

```
Switch(config)# access-list 1 remark Permit only Jones workstation through
Switch(config)# access-list 1 permit 171.69.2.88
Switch(config)# access-list 1 remark Do not allow Smith workstation through
Switch(config)# access-list 1 deny 171.69.3.13
```

In this example of a numbered ACL, the Winter and Smith workstations are not allowed to browse the web:

```
Switch(config)# access-list 100 remark Do not allow Winter to browse the web
Switch(config)# access-list 100 deny host 171.69.3.85 any eq www
Switch(config)# access-list 100 remark Do not allow Smith to browse the web
Switch(config)# access-list 100 deny host 171.69.3.13 any eq www
```

In this example of a named ACL, the Jones subnet is not allowed access:

```
Switch(config)# ip access-list standard prevention
Switch(config-std-nacl)# remark Do not allow Jones subnet through
Switch(config-std-nacl)# deny 171.69.0.0 0.0.255.255
```

In this example of a named ACL, the Jones subnet is not allowed to use outbound Telnet:

```
Switch(config)# ip access-list extended telnetting
Switch(config-ext-nacl)# remark Do not allow Jones subnet to telnet out
Switch(config-ext-nacl)# deny tcp 171.69.0.0 0.0.255.255 any eq telnet
```

Examples: ACL Logging

Two variations of logging are supported on router ACLs. The **log** keyword sends an informational logging message to the console about the packet that matches the entry; the **log-input** keyword includes the input interface in the log entry.

In this example, standard named access list *stan1* denies traffic from 10.1.1.0 0.0.0.255, allows traffic from all other sources, and includes the **log** keyword.

```
Switch(config)# ip access-list standard stan1
Switch(config-std-nacl)# deny 10.1.1.0 0.0.0.255 log
Switch(config-std-nacl)# permit any log
Switch(config-std-nacl)# exit
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip access-group stan1 in
Switch(config-if)# end
Switch# show logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
    Console logging: level debugging, 37 messages logged
    Monitor logging: level debugging, 0 messages logged
    Buffer logging: level debugging, 37 messages logged
    File logging: disabled
    Trap logging: level debugging, 39 message lines logged

Log Buffer (4096 bytes):
```

```
00:00:48: NTP: authentication delay calculation problems

<output truncated>

00:09:34:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
00:09:59:%SEC-6-IPACCESSLOGS:list stan1 denied 10.1.1.15 1 packet
00:10:11:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
```

This example is a named extended access list *ext1* that permits ICMP packets from any source to 10.1.1.0 0.0.0.255 and denies all UDP packets.

```
Switch(config)# ip access-list extended ext1
Switch(config-ext-nacl)# permit icmp any 10.1.1.0 0.0.0.255 log
Switch(config-ext-nacl)# deny udp any any log
Switch(config-std-nacl)# exit
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# ip access-group ext1 in
```

This is a an example of a log for an extended ACL:

```
01:24:23:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 1
packet
01:25:14:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 7
packets
01:26:12:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 1 packet
01:31:33:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 8 packets
```

Note that all logging entries for IP ACLs start with %SEC-6-IPACCESSLOG with minor variations in format depending on the kind of ACL and the access entry that has been matched.

This is an example of an output message when the **log-input** keyword is entered:

```
00:04:21:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 (Vlan1 0001.42ef.a400)
 ->
10.1.1.61 (0/0), 1 packet
```

A log message for the same sort of packet using the **log** keyword does not include the input interface information:

```
00:05:47:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 -> 10.1.1.61 (0/0), 1
 packet
```

Configuration Examples for ACLs and VLAN Maps

Example: Creating an ACL and a VLAN Map to Deny a Packet

This example shows how to create an ACL and a VLAN map to deny a packet. In the first map, any packets that match the *ip1* ACL (TCP packets) would be dropped. You first create the *ip1* ACL to permit any TCP packet and no other packets. Because there is a match clause for IP packets in the VLAN map, the default action is to drop any IP packet that does not match any of the match clauses.

```
Switch(config)# ip access-list extended ip1
Switch(config-ext-nacl)# permit tcp any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map map_1 10
Switch(config-access-map)# match ip address ip1
Switch(config-access-map)# action drop
```

Example: Creating an ACL and a VLAN Map to Permit a Packet

This example shows how to create a VLAN map to permit a packet. ACL *ip2* permits UDP packets and any packets that match the *ip2* ACL are forwarded. In this map, any IP packets that did not match any of the previous ACLs (that is, packets that are not TCP packets or UDP packets) would get dropped.

```
Switch(config)# ip access-list extended ip2
Switch(config-ext-nacl)# permit udp any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map map_1 20
Switch(config-access-map)# match ip address ip2
Switch(config-access-map)# action forward
```

Example: Default Action of Dropping IP Packets and Forwarding MAC Packets

In this example, the VLAN map has a default action of drop for IP packets and a default action of forward for MAC packets. Used with standard ACL 101 and extended named access lists **igmp-match** and **tcp-match**, the map will have the following results:

- Forward all UDP packets
- Drop all IGMP packets
- Forward all TCP packets
- Drop all other IP packets
- Forward all non-IP packets

```
Switch(config)# access-list 101 permit udp any any
Switch(config)# ip access-list extended igmp-match
Switch(config-ext-nacl)# permit igmp any any
Switch(config)# action forward
Switch(config-ext-nacl)# permit tcp any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map drop-ip-default 10
Switch(config-access-map)# match ip address 101
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-ip-default 20
Switch(config-access-map)# match ip address igmp-match
Switch(config-access-map)# action drop
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-ip-default 30
Switch(config-access-map)# match ip address tcp-match
Switch(config-access-map)# action forward
```

Example: Default Action of Dropping MAC Packets and Forwarding IP Packets

In this example, the VLAN map has a default action of drop for MAC packets and a default action of forward for IP packets. Used with MAC extended access lists **good-hosts** and **good-protocols**, the map will have the following results:

- Forward MAC packets from hosts 0000.0c00.0111 and 0000.0c00.0211
- Forward MAC packets with decnet-iv or vines-ip protocols
- Drop all other non-IP packets

• Forward all IP packets

```
Switch(config)# mac access-list extended good-hosts
Switch(config-ext-macl)# permit host 000.0c00.0111 any
Switch(config-ext-macl)# permit host 000.0c00.0211 any
Switch(config-ext-nacl)# exit
Switch(config)# action forward
Switch(config-ext-macl)# mac access-list extended good-protocols
Switch(config-ext-macl)# permit any any vines-ip
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map drop-mac-default 10
Switch(config-access-map)# match mac address good-hosts
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-mac-default 20
Switch(config-access-map)# match mac address good-protocols
Switch(config-access-map)# action forward
```

Example: Default Action of Dropping All Packets

In this example, the VLAN map has a default action of drop for all packets (IP and non-IP). Used with access lists **tcp-match** and **good-hosts** from Examples 2 and 3, the map will have the following results:

• Forward all TCP packets

• Forward MAC packets from hosts 0000.0c00.0111 and 0000.0c00.0211

• Drop all other IP packets

• Drop all other MAC packets

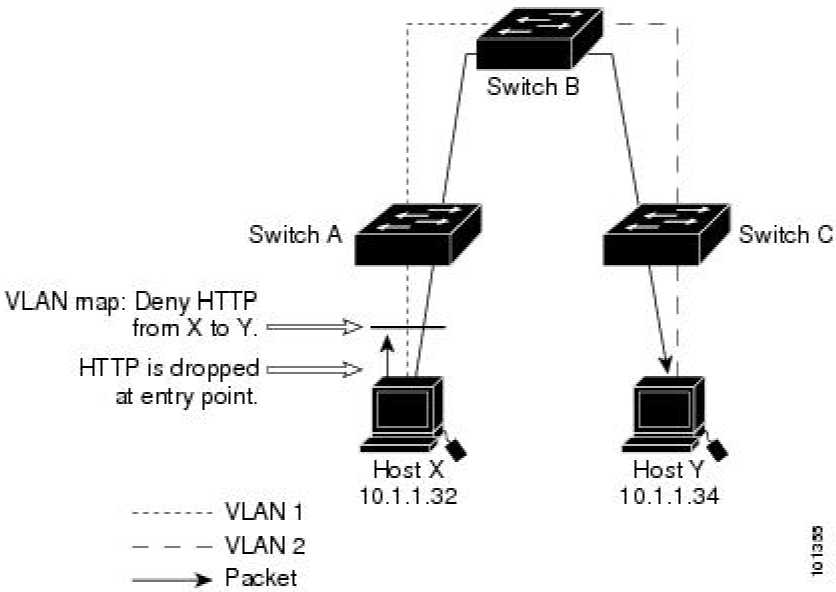
```
Switch(config)# vlan access-map drop-all-default 10
Switch(config-access-map)# match ip address tcp-match
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-all-default 20
Switch(config-access-map)# match mac address good-hosts
Switch(config-access-map)# action forward
```

Configuration Examples for Using VLAN Maps in Your Network

Example: Wiring Closet Configuration

In a wiring closet configuration, routing might not be enabled on the switch. In this configuration, the switch can still support a VLAN map and a QoS classification ACL. Assume that Host X and Host Y are in different VLANs and are connected to wiring closet switches A and C. Traffic from Host X to Host Y is eventually being routed by Switch B, a Layer 3 switch with routing enabled. Traffic from Host X to Host Y can be access-controlled at the traffic entry point, Switch A.

Figure 39: Wiring Closet Configuration



If you do not want HTTP traffic switched from Host X to Host Y, you can configure a VLAN map on Switch A to drop all HTTP traffic from Host X (IP address 10.1.1.32) to Host Y (IP address 10.1.1.34) at Switch A and not bridge it to Switch B.

First, define the IP access list *http* that permits (matches) any TCP traffic on the HTTP port.

```
Switch(config)# ip access-list extended http
Switch(config-ext-nacl)# permit tcp host 10.1.1.32 host 10.1.1.34 eq www
Switch(config-ext-nacl)# exit
```

Next, create VLAN access map *map2* so that traffic that matches the *http* access list is dropped and all other IP traffic is forwarded.

```
Switch(config)# vlan access-map map2 10
Switch(config-access-map)# match ip address http
Switch(config-access-map)# action drop
Switch(config-access-map)# exit
Switch(config)# ip access-list extended match_all
Switch(config-ext-nacl)# permit ip any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map map2 20
Switch(config-access-map)# match ip address match_all
Switch(config-access-map)# action forward
```

Then, apply VLAN access map *map2* to VLAN 1.

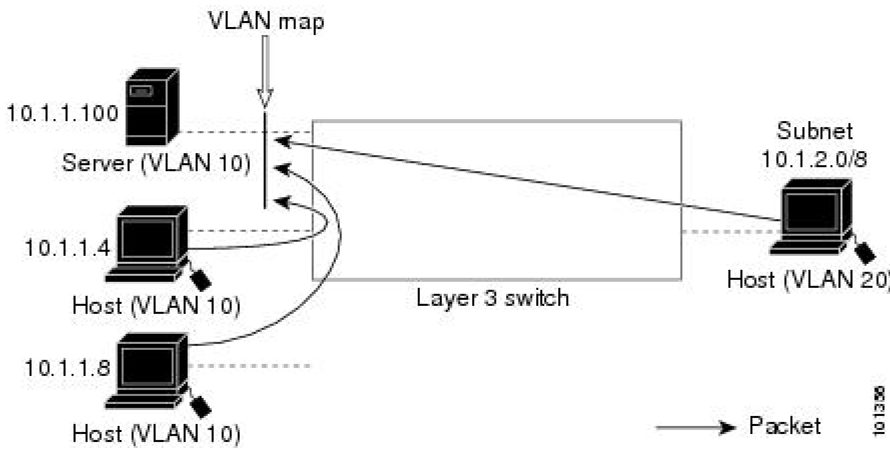
```
Switch(config)# vlan filter map2 vlan 1
```

Example: Restricting Access to a Server on Another VLAN

You can restrict access to a server on another VLAN. For example, server 10.1.1.100 in VLAN 10 needs to have access denied to these hosts:

- Hosts in subnet 10.1.2.0/8 in VLAN 20 should not have access.
- Hosts 10.1.1.4 and 10.1.1.8 in VLAN 10 should not have access.

Figure 40: Restricting Access to a Server on Another VLAN



Example: Denying Access to a Server on Another VLAN

This example shows how to deny access to a server on another VLAN by creating the VLAN map SERVER 1 that denies access to hosts in subnet 10.1.2.0.8, host 10.1.1.4, and host 10.1.1.8 and permits other IP traffic. The final step is to apply the map SERVER1 to VLAN 10.

Define the IP ACL that will match the correct packets.

```
Switch(config)# ip access-list extended SERVER1_ACL
Switch(config-ext-nacl))# permit ip 10.1.2.0 0.0.0.255 host 10.1.1.100
Switch(config-ext-nacl))# permit ip host 10.1.1.4 host 10.1.1.100
Switch(config-ext-nacl))# permit ip host 10.1.1.8 host 10.1.1.100
Switch(config-ext-nacl))# exit
```

Define a VLAN map using this ACL that will drop IP packets that match SERVER1_ACL and forward IP packets that do not match the ACL.

```
Switch(config)# vlan access-map SERVER1_MAP
Switch(config-access-map)# match ip address SERVER1_ACL
Switch(config-access-map)# action drop
Switch(config)# vlan access-map SERVER1_MAP 20
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
```

Apply the VLAN map to VLAN 10.

```
Switch(config)# vlan filter SERVER1_MAP vlan-list 10
```

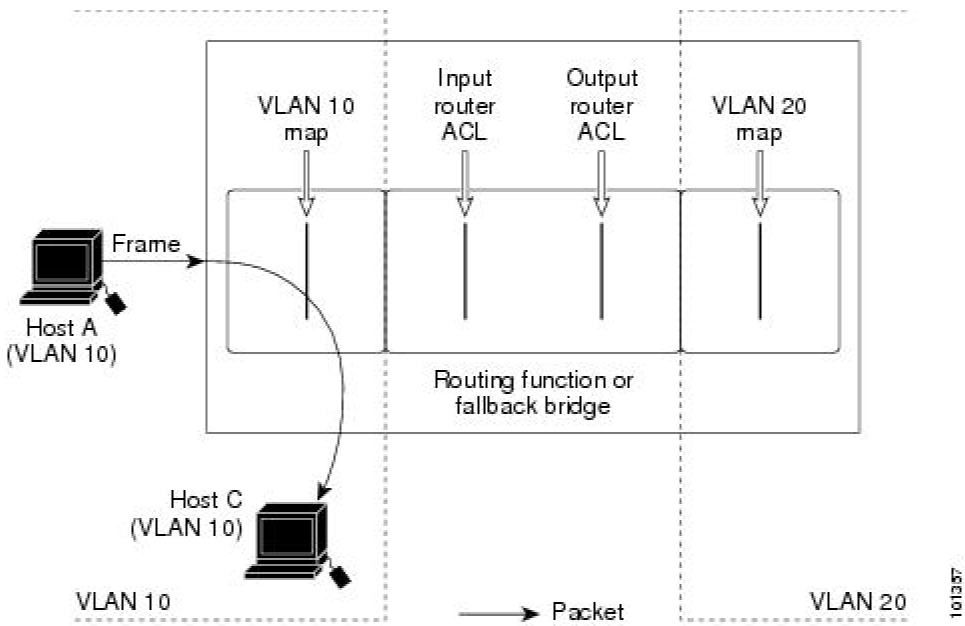
Configuration Examples of Router ACLs and VLAN Maps Applied to VLANs

This section gives examples of applying router ACLs and VLAN maps to a VLAN for switched, bridged, routed, and multicast packets. Although the following illustrations show packets being forwarded to their destination, each time the packet's path crosses a line indicating a VLAN map or an ACL, it is also possible that the packet might be dropped, rather than forwarded.

Example: ACLs and Switched Packets

This example shows how an ACL is applied on packets that are switched within a VLAN. Packets switched within the VLAN without being routed or forwarded by fallback bridging are only subject to the VLAN map of the input VLAN.

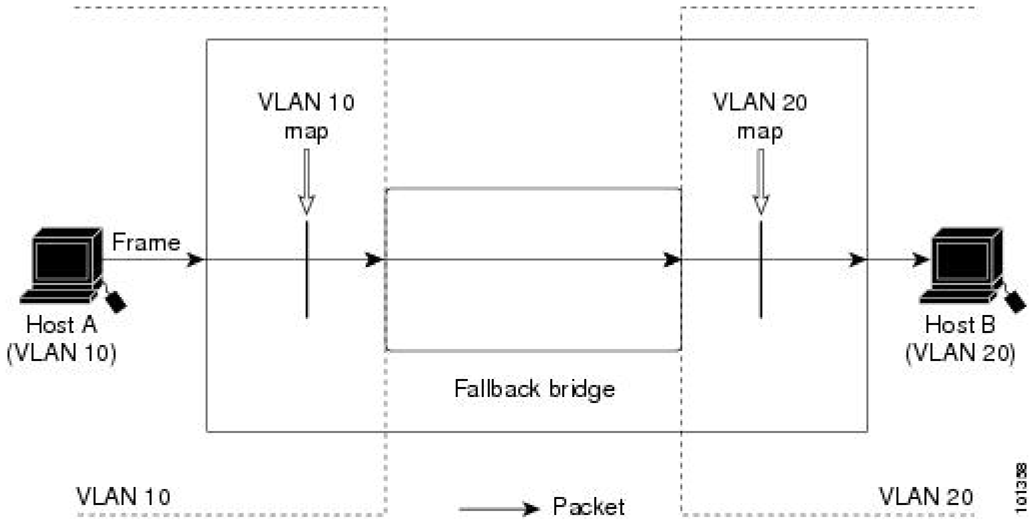
Figure 41: Applying ACLs on Switched Packets



Example: ACLs and Bridged Packets

This example shows how an ACL is applied on fallback-bridged packets. For bridged packets, only Layer 2 ACLs are applied to the input VLAN. Only non-IP, non-ARP packets can be fallback-bridged.

Figure 42: Applying ACLs on Bridged Packets

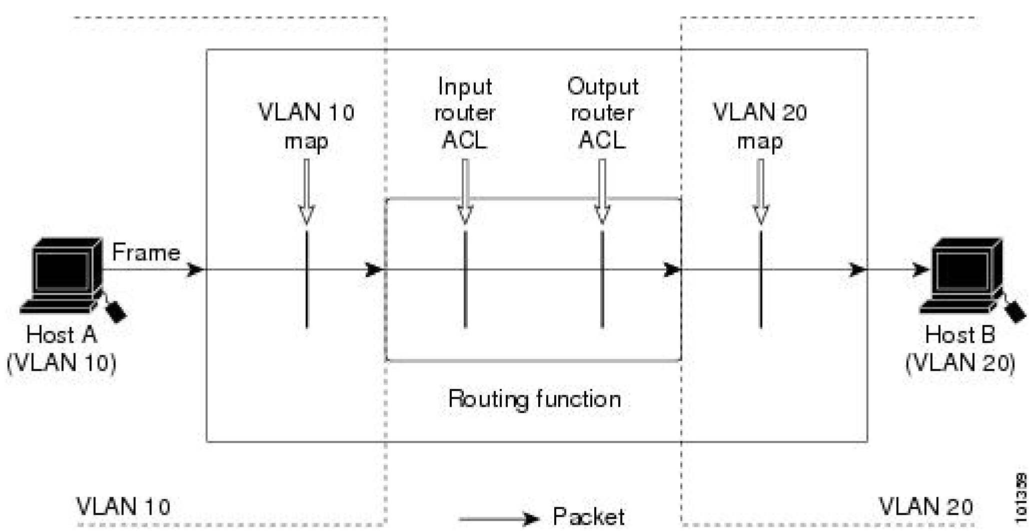


Example: ACLs and Routed Packets

This example shows how ACLs are applied on routed packets. The ACLs are applied in this order:

1. VLAN map for input VLAN
2. Input router ACL
3. Output router ACL
4. VLAN map for output VLAN

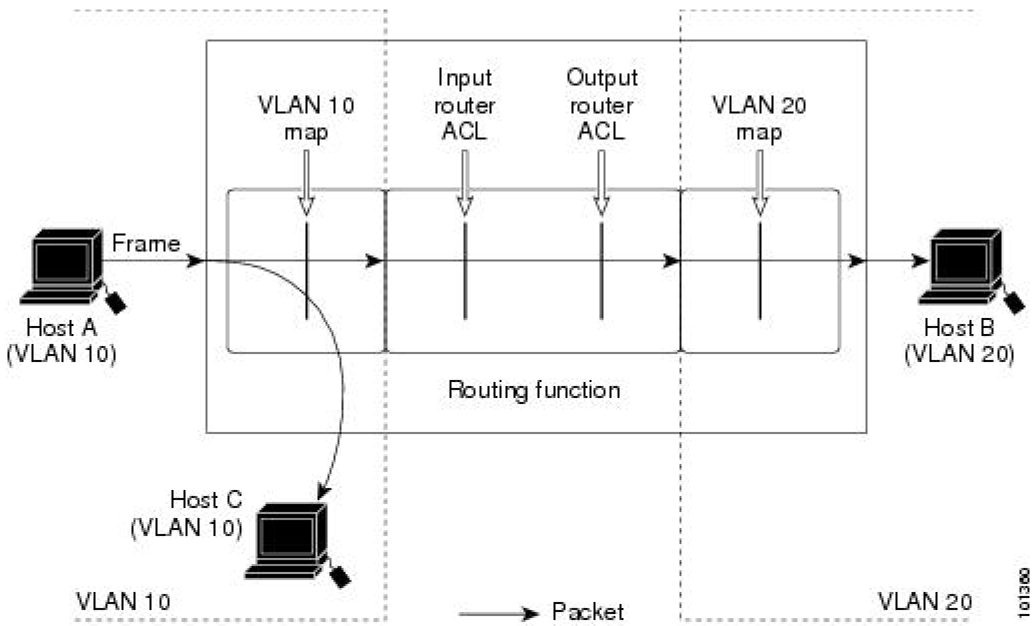
Figure 43: Applying ACLs on Routed Packets



Example: ACLs and Multicast Packets

This example shows how ACLs are applied on packets that are replicated for IP multicasting. A multicast packet being routed has two different kinds of filters applied: one for destinations that are other ports in the input VLAN and another for each of the destinations that are in other VLANs to which the packet has been routed. The packet might be routed to more than one output VLAN, in which case a different router output ACL and VLAN map would apply for each destination VLAN. The final result is that the packet might be permitted in some of the output VLANs and not in others. A copy of the packet is forwarded to those destinations where it is permitted. However, if the input VLAN map drops the packet, no destination receives a copy of the packet.

Figure 44: Applying ACLs on Multicast Packets



Additional References

Related Documents

Related Topic	Document Title
IPv4 Access Control List topics	Securing the Data Plane Configuration Guide Library, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/en/US/docs/ios-xml/ios/security/config_library/xe-3se/3850/secdata-xe-3se-3850-library.html

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

CHAPTER 52

Configuring DHCP

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information About DHCP

DHCP Server

The DHCP server assigns IP addresses from specified address pools on a switch or router to DHCP clients and manages them. If the DHCP server cannot give the DHCP client the requested configuration parameters from its database, it forwards the request to one or more secondary DHCP servers defined by the network administrator.

The switch can act as a DHCP server.

DHCP Relay Agent

A DHCP relay agent is a Layer 3 device that forwards DHCP packets between clients and servers. Relay agents forward requests and replies between clients and servers when they are not on the same physical subnet. Relay agent forwarding is different from the normal Layer 2 forwarding, in which IP datagrams are switched transparently between networks. Relay agents receive DHCP messages and generate new DHCP messages to send on output interfaces.

DHCP Snooping

DHCP snooping is a DHCP security feature that provides network security by filtering untrusted DHCP messages and by building and maintaining a DHCP snooping binding database, also referred to as a DHCP snooping binding table.

DHCP snooping acts like a firewall between untrusted hosts and DHCP servers. You use DHCP snooping to differentiate between untrusted interfaces connected to the end user and trusted interfaces connected to the DHCP server or another switch.

Note For DHCP snooping to function properly, all DHCP servers must be connected to the switch through trusted interfaces.

An untrusted DHCP message is a message that is received through an untrusted interface. By default, the switch considers all interfaces untrusted. So, the switch must be configured to trust some interfaces to use DHCP Snooping. When you use DHCP snooping in a service-provider environment, an untrusted message is sent from a device that is not in the service-provider network, such as a customer's switch. Messages from unknown devices are untrusted because they can be sources of traffic attacks.

The DHCP snooping binding database has the MAC address, the IP address, the lease time, the binding type, the VLAN number, and the interface information that corresponds to the local untrusted interfaces of a switch. It does not have information regarding hosts interconnected with a trusted interface.

In a service-provider network, an example of an interface you might configure as trusted is one connected to a port on a device in the same network. An example of an untrusted interface is one that is connected to an untrusted interface in the network or to an interface on a device that is not in the network.

When a switch receives a packet on an untrusted interface and the interface belongs to a VLAN in which DHCP snooping is enabled, the switch compares the source MAC address and the DHCP client hardware address. If the addresses match (the default), the switch forwards the packet. If the addresses do not match, the switch drops the packet.

The switch drops a DHCP packet when one of these situations occurs:

- A packet from a DHCP server, such as a DHCPOFFER, DHCPACK, DHCPNAK, or DHCPLEASEQUERY packet, is received from outside the network or firewall.
- A packet is received on an untrusted interface, and the source MAC address and the DHCP client hardware address do not match.
- The switch receives a DHCPRELEASE or DHCPDECLINE broadcast message that has a MAC address in the DHCP snooping binding database, but the interface information in the binding database does not match the interface on which the message was received.

- A DHCP relay agent forwards a DHCP packet that includes a relay-agent IP address that is not 0.0.0.0, or the relay agent forwards a packet that includes option-82 information to an untrusted port.

If the switch is an aggregation switch supporting DHCP snooping and is connected to an edge switch that is inserting DHCP option-82 information, the switch drops packets with option-82 information when packets are received on an untrusted interface. If DHCP snooping is enabled and packets are received on a trusted port, the aggregation switch does not learn the DHCP snooping bindings for connected devices and cannot build a complete DHCP snooping binding database.

When an aggregation switch can be connected to an edge switch through an untrusted interface and you enter the **ip dhcp snooping information option allow-untrusted** global configuration command, the aggregation switch accepts packets with option-82 information from the edge switch. The aggregation switch learns the bindings for hosts connected through an untrusted switch interface. The DHCP security features, such as dynamic ARP inspection or IP source guard, can still be enabled on the aggregation switch while the switch receives packets with option-82 information on untrusted input interfaces to which hosts are connected. The port on the edge switch that connects to the aggregation switch must be configured as a trusted interface.

Related Topics

Prerequisites for Configuring DHCP Snooping and Option 82, on page 1096

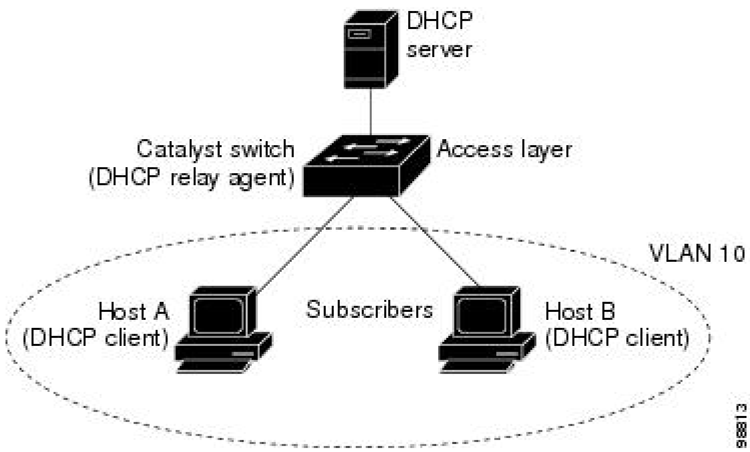
Option-82 Data Insertion

In residential, metropolitan Ethernet-access environments, DHCP can centrally manage the IP address assignments for a large number of subscribers. When the DHCP option-82 feature is enabled on the switch, a subscriber device is identified by the switch port through which it connects to the network (in addition to its MAC address). Multiple hosts on the subscriber LAN can be connected to the same port on the access switch and are uniquely identified.

Note The DHCP option-82 feature is supported only when DHCP snooping is globally enabled on the VLANs to which subscriber devices using option-82 are assigned.

The following illustration shows a metropolitan Ethernet network in which a centralized DHCP server assigns IP addresses to subscribers connected to the switch at the access layer. Because the DHCP clients and their associated DHCP server do not reside on the same IP network or subnet, a DHCP relay agent (the Catalyst switch) is configured with a helper address to enable broadcast forwarding and to transfer DHCP messages between the clients and the server.

Figure 45: DHCP Relay Agent in a Metropolitan Ethernet Network



When you enable the DHCP snooping information option 82 on the switch, the following sequence of events occurs:

- The host (DHCP client) generates a DHCP request and broadcasts it on the network.
- When the switch receives the DHCP request, it adds the option-82 information in the packet. By default, the remote-ID suboption is the switch MAC address, and the circuit-ID suboption is the port identifier, **vlan-mod-port**, from which the packet is received.You can configure the remote ID and circuit ID.
- If the IP address of the relay agent is configured, the switch adds this IP address in the DHCP packet.
- The switch forwards the DHCP request that includes the option-82 field to the DHCP server.
- The DHCP server receives the packet. If the server is option-82-capable, it can use the remote ID, the circuit ID, or both to assign IP addresses and implement policies, such as restricting the number of IP addresses that can be assigned to a single remote ID or circuit ID. Then the DHCP server echoes the option-82 field in the DHCP reply.
- The DHCP server unicasts the reply to the switch if the request was relayed to the server by the switch. The switch verifies that it originally inserted the option-82 data by inspecting the remote ID and possibly the circuit ID fields. The switch removes the option-82 field and forwards the packet to the switch port that connects to the DHCP client that sent the DHCP request.
-

In the default suboption configuration, when the described sequence of events occurs, the values in these fields do not change (see the illustration,*Suboption Packet Formats*):

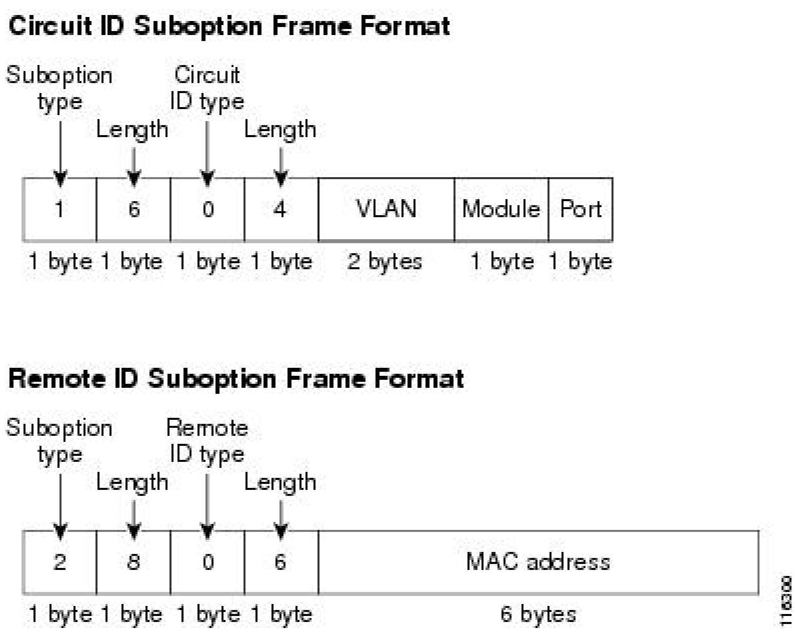
- Circuit-ID suboption fields
    - Suboption type
    - Length of the suboption type
    - Circuit-ID type
    - Length of the circuit-ID type

- Remote-ID suboption fields
    - Suboption type
    - Length of the suboption type
    - Remote-ID type
    - Length of the remote-ID type

In the port field of the circuit ID suboption, the port numbers start at 3. For example, on a switch with 24 10/100/1000 ports and four small form-factor pluggable (SFP) module slots, port 3 is the Gigabit Ethernet 1/0/1 port, port 4 is the Gigabit Ethernet 1/0/2 port, and so forth. Port 27 is the SFP module slot Gigabit Ethernet1/0/25, and so forth.

The illustration, *Suboption Packet Formats*. shows the packet formats for the remote-ID suboption and the circuit-ID suboption when the default suboption configuration is used. For the circuit-ID suboption, the module number corresponds to the switch number in the stack. The switch uses the packet formats when you globally enable DHCP snooping and enter the ip dhcp snooping information option global configuration command.

Figure 46: Suboption Packet Formats



The illustration, *User-Configured Suboption Packet Formats,* shows the packet formats for user-configured remote-ID and circuit-ID suboptions The switch uses these packet formats when DHCP snooping is globally enabled and when the **ip dhcp snooping information option format remote-id** global configuration command and the**ip dhcp snooping vlan information option format-type circuit-id string** interface configuration command are entered.

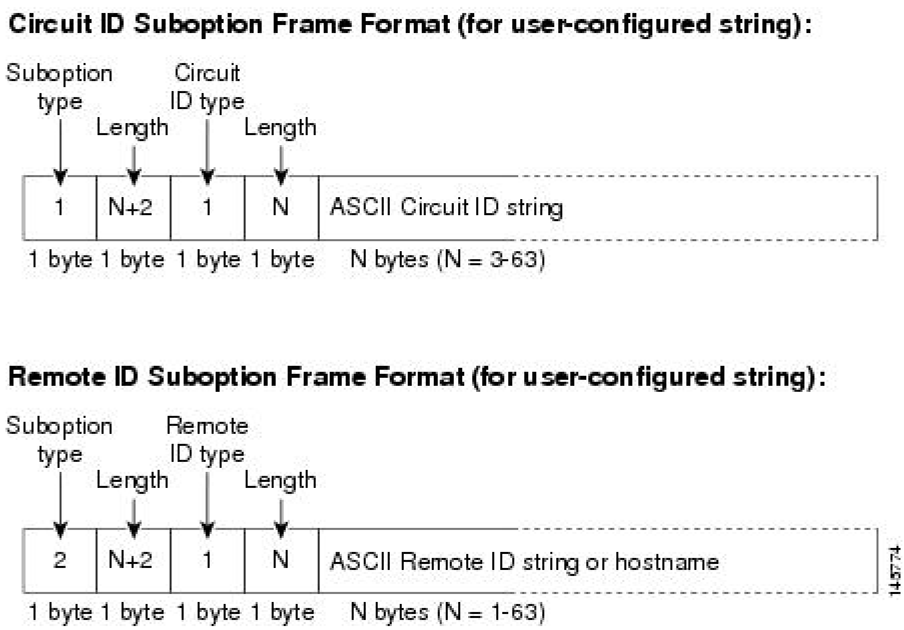
The values for these fields in the packets change from the default values when you configure the remote-ID and circuit-ID suboptions:

- Circuit-ID suboption fields
    - The circuit-ID type is 1.

- The length values are variable, depending on the length of the string that you configure.

- Remote-ID suboption fields
  - The remote-ID type is 1.
  - The length values are variable, depending on the length of the string that you configure.

Figure 47: User-Configured Suboption Packet Formats



Cisco IOS DHCP Server Database

During the DHCP-based autoconfiguration process, the designated DHCP server uses the Cisco IOS DHCP server database. It has IP addresses, address bindings, and configuration parameters, such as the boot file.

An address binding is a mapping between an IP address and a MAC address of a host in the Cisco IOS DHCP server database. You can manually assign the client IP address, or the DHCP server can allocate an IP address from a DHCP address pool. For more information about manual and automatic address bindings, see the "Configuring DHCP" chapter of the *Cisco IOS IP Configuration Guide, Release 12.4*.

For procedures to enable and configure the Cisco IOS DHCP server database, see the "DHCP Configuration Task List" section in the "Configuring DHCP" chapter of the *Cisco IOS IP Configuration Guide, Release 12.4*.

DHCP Snooping Binding Database

When DHCP snooping is enabled, the switch uses the DHCP snooping binding database to store information about untrusted interfaces. The database can have up to 64,000 bindings.

Each database entry (binding) has an IP address, an associated MAC address, the lease time (in hexadecimal format), the interface to which the binding applies, and the VLAN to which the interface belongs. The database agent stores the bindings in a file at a configured location. At the end of each entry is a checksum that accounts for all the bytes from the start of the file through all the bytes associated with the entry. Each entry is 72 bytes, followed by a space and then the checksum value.

To keep the bindings when the switch reloads, you must use the DHCP snooping database agent. If the agent is disabled, dynamic ARP inspection or IP source guard is enabled, and the DHCP snooping binding database has dynamic bindings, the switch loses its connectivity. If the agent is disabled and only DHCP snooping is enabled, the switch does not lose its connectivity, but DHCP snooping might not prevent DHCP spoofing attacks.

When reloading, the switch reads the binding file to build the DHCP snooping binding database. The switch updates the file when the database changes.

When a switch learns of new bindings or when it loses bindings, the switch immediately updates the entries in the database. The switch also updates the entries in the binding file. The frequency at which the file is updated is based on a configurable delay, and the updates are batched. If the file is not updated in a specified time (set by the write-delay and abort-timeout values), the update stops.

This is the format of the file with bindings:

```
<initial-checksum>
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
<entry-1> <checksum-1>
<entry-2> <checksum-1-2>
...
...
<entry-n> <checksum-1-2-..-n>
END
```

Each entry in the file is tagged with a checksum value that the switch uses to verify the entries when it reads the file. The initial-checksum entry on the first line distinguishes entries associated with the latest file update from entries associated with a previous file update.

This is an example of a binding file:

```
2bb4c2a1
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
192.1.168.1 3 0003.47d8.c91f 2BB6488E Gi1/0/4 21ae5fbb
192.1.168.3 3 0003.44d6.c52f 2BB648EB Gi1/0/4 1bdb223f
192.1.168.2 3 0003.47d9.c8f1 2BB648AB Gi1/0/4 584a38f0
END
```

When the switch starts and the calculated checksum value equals the stored checksum value, the switch reads entries from the binding file and adds the bindings to its DHCP snooping binding database. The switch ignores an entry when one of these situations occurs:

- The switch reads the entry and the calculated checksum value does not equal the stored checksum value. The entry and the ones following it are ignored.
- An entry has an expired lease time (the switch might not remove a binding entry when the lease time expires).
- The interface in the entry no longer exists on the system.
- The interface is a routed interface or a DHCP snooping-trusted interface.

DHCP Snooping and Switch Stacks

DHCP snooping is managed on the stack master. When a new switch joins the stack, the switch receives the DHCP snooping configuration from the stack master. When a member leaves the stack, all DHCP snooping address bindings associated with the switch age out.

All snooping statistics are generated on the stack master. If a new stack master is elected, the statistics counters reset.

When a stack merge occurs, all DHCP snooping bindings in the stack master are lost if it is no longer the stack master. With a stack partition, the existing stack master is unchanged, and the bindings belonging to the partitioned switches age out. The new master of the partitioned stack begins processing the new incoming DHCP packets.

How to Configure DHCP Features

Default DHCP Snooping Configuration

Table 127: Default DHCP Configuration

Feature	Default Setting
DHCP server	Enabled in Cisco IOS software, requires configuration[17]
DHCP relay agent	Enabled[18]
DHCP packet forwarding address	None configured
Checking the relay agent information	Enabled (invalid messages are dropped)
DHCP relay agent forwarding policy	Replace the existing relay agent information
DHCP snooping enabled globally	Disabled
DHCP snooping information option	Enabled
DHCP snooping option to accept packets on untrusted input interfaces[19]	Disabled
DHCP snooping limit rate	None configured
DHCP snooping trust	Untrusted
DHCP snooping VLAN	Disabled
DHCP snooping MAC address verification	Enabled

Feature	Default Setting
Cisco IOS DHCP server binding database	Enabled in Cisco IOS software, requires configuration. **Note** The switch gets network addresses and configuration parameters only from a device configured as a DHCP server.
DHCP snooping binding database agent	Enabled in Cisco IOS software, requires configuration. This feature is operational only when a destination is configured.

17 The switch responds to DHCP requests only if it is configured as a DHCP server.
18 The switch relays DHCP packets only if the IP address of the DHCP server is configured on the SVI of the DHCP client.
19 Use this feature when the switch is an aggregation switch that receives packets with option-82 information from an edge switch.

DHCP Snooping Configuration Guidelines

- If a switch port is connected to a DHCP server, configure a port as trusted by entering the **ip dhcp snooping trust interface** configuration command.
- If a switch port is connected to a DHCP client, configure a port as untrusted by entering the **no ip dhcp snooping trust** interface configuration command.
- You can display DHCP snooping statistics by entering the **show ip dhcp snooping statistics** user EXEC command, and you can clear the snooping statistics counters by entering the **clear ip dhcp snooping statistics** privileged EXEC command.

Configuring the DHCP Server

The switch can act as a DHCP server.

For procedures to configure the switch as a DHCP server, see the “Configuring DHCP” section of the “IP addressing and Services” section of the Cisco IOS IP Configuration Guide, Release 12.4.

DHCP Server and Switch Stacks

The DHCP binding database is managed on the stack master. When a new stack master is assigned, the new master downloads the saved binding database from the TFTP server. If the stack master fails, all unsaved bindings are lost. The IP addresses associated with the lost bindings are released. You should configure an automatic backup by using the **ip dhcp database** *url* [**timeout** *seconds* | **write-delay** *seconds*] global configuration command.

When a stack merge occurs, the stack master that becomes a stack member loses all of the DHCP lease bindings. With a stack partition, the new master in the partition acts as a new DHCP server without any of the existing DHCP lease bindings.

Configuring the DHCP Relay Agent

Beginning in privileged EXEC mode, follow these steps to enable the DHCP relay agent on the switch:

SUMMARY STEPS

1. **configure terminal**
2. **service dhcp**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**service dhcp** **Example:** `Switch(config)# service dhcp`	Enables the DHCP server and relay agent on your switch. By default, this feature is enabled.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

What to Do Next

See the "*Configuring DHCP*" section of the "IP Addressing and Services" section of the *Cisco IOS IP Configuration Guide, Release 12.4* for these procedures:

- Checking (validating) the relay agent information
- Configuring the relay agent forwarding policy

Specifying the Packet Forwarding Address

If the DHCP server and the DHCP clients are on different networks or subnets, you must configure the switch with the **ip helper-address** *address* interface configuration command. The general rule is to configure the command on the Layer 3 interface closest to the client. The address used in the **ip helper-address** command can be a specific DHCP server IP address, or it can be the network address if other DHCP servers are on the destination network segment. Using the network address enables any DHCP server to respond to requests.

Beginning in privileged EXEC mode, follow these steps to specify the packet forwarding address:

SUMMARY STEPS

1. **configure terminal**
2. **interface vlan** *vlan-id*
3. **ip address** *ip-address subnet-mask*
4. **ip helper-address** *address*
5. **end**
6. **interface range** *port-range* or **interface** *interface-id*
7. **switchport mode access**
8. **switchport access vlan** *vlan-id*
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface vlan** *vlan-id* **Example:** `Switch(config)# interface vlan 1`	Creates a switch virtual interface by entering a VLAN ID, and enter interface configuration mode.
Step 3	**ip address** *ip-address subnet-mask* **Example:** `Switch(config-if)# ip address 192.108.1.27 255.255.255.0`	Configures the interface with an IP address and an IP subnet.
Step 4	**ip helper-address** *address* **Example:** `Switch(config-if)# ip helper-address 172.16.1.2`	Specifies the DHCP packet forwarding address. The helper address can be a specific DHCP server address, or it can be the network address if other DHCP servers are on the destination network segment. Using the network address enables other servers to respond to DHCP requests. If you have multiple servers, you can configure one helper address for each server.
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to global configuration mode.

	Command or Action	Purpose
Step 6	**interface range** *port-range* or **interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/2`	Configures multiple physical ports that are connected to the DHCP clients, and enter interface range configuration mode. or Configures a single physical port that is connected to the DHCP client, and enter interface configuration mode.
Step 7	**switchport mode access** **Example:** `Switch(config-if)# switchport mode access`	Defines the VLAN membership mode for the port.
Step 8	**switchport access vlan** *vlan-id* **Example:** `Switch(config-if)# switchport access vlan 1`	Assigns the ports to the same VLAN as configured in Step 2.
Step 9	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Prerequisites for Configuring DHCP Snooping and Option 82

The prerequisites for DHCP Snooping and Option 82 are as follows:

- Before configuring the DHCP snooping information option on your switch, be sure to configure the device that is acting as the DHCP server. You must specify the IP addresses that the DHCP server can assign or exclude, or you must configure DHCP options for these devices.
- For DHCP snooping to function properly, all DHCP servers must be connected to the switch through trusted interfaces.
- Before configuring the DHCP relay agent on your switch, make sure to configure the device that is acting as the DHCP server. You must specify the IP addresses that the DHCP server can assign or exclude, configure DHCP options for devices, or set up the DHCP database agent.
- The following prerequisites apply to DHCP snooping binding database configuration:
    - Because both NVRAM and the flash memory have limited storage capacity, we recommend that you store the binding file on a TFTP server.
    - For network-based URLs (such as TFTP and FTP), you must create an empty file at the configured URL before the switch can write bindings to the binding file at that URL. See the documentation for your TFTP server to determine whether you must first create an empty file on the server; some TFTP servers cannot be configured this way.

    - To ensure that the lease time in the database is accurate, we recommend that you enable and configure Network Time Protocol (NTP).
    - If NTP is configured, the switch writes binding changes to the binding file only when the switch system clock is synchronized with NTP.

- If you want the switch to respond to DHCP requests, it must be configured as a DHCP server.
- If you want the switch to relay DHCP packets, the IP address of the DHCP server must be configured on the switch virtual interface (SVI) of the DHCP client.
- To use the DHCP snooping option of accepting packets on untrusted inputs, the switch must be an aggregation switch that receives packets with option-82 information from an edge switch.
- You must configure the switch to use the Cisco IOS DHCP server binding database to use it for DHCP snooping.
- You must configure a destination on the DHCP snooping binding database to use the switch for DHCP snooping.
- For DHCP snooping to function properly, all DHCP servers must be connected to the switch through trusted interfaces. In a service-provider network, a trusted interface is connected to a port on a device in the same network.
- You must globally enable DHCP snooping on the switch.
- Before globally enabling DHCP snooping on the switch, make sure that the devices acting as the DHCP server and the DHCP relay agent are configured and enabled.
- **Note** Do not enable Dynamic Host Configuration Protocol (DHCP) snooping on RSPAN VLANs. If DHCP snooping is enabled on RSPAN VLANs, DHCP packets might not reach the RSPAN destination port.

Related Topics

DHCP Snooping, on page 1086

Enabling DHCP Snooping and Option 82

Beginning in privileged EXEC mode, follow these steps to enable DHCP snooping on the switch:

SUMMARY STEPS

1. **configure terminal**
2. **ip dhcp snooping**
3. **ip dhcp snooping vlan** *vlan-range* []
4. **ip dhcp snooping information option**
5. **ip dhcp snooping information option format remote-id** [**string** *ASCII-string* | **hostname**]
6. **ip dhcp snooping information option allow-untrusted**
7. **interface** *interface-id*
8. **ip dhcp snooping vlan** *vlan* **information option format-type circuit-id** [**override**] **string** *ASCII-string*
9. **ip dhcp snooping trust**
10. **ip dhcp snooping limit rate** *rate*
11. **exit**
12. **ip dhcp snooping verify mac-address**
13. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip dhcp snooping** **Example:** `Switch(config)# ip dhcp snooping`	Enables DHCP snooping globally.
Step 3	**ip dhcp snooping vlan** *vlan-range* [] **Example:** `Switch(config)# ip dhcp snooping vlan 10`	Enables DHCP snooping on a VLAN or range of VLANs. The range is 1 to 4094. • You can enter a single VLAN ID identified by VLAN ID number, a series of VLAN IDs separated by commas, a range of VLAN IDs separated by hyphens, or a range of VLAN IDs separated by entering the starting and ending VLAN IDs separated by a space.
Step 4	**ip dhcp snooping information option** **Example:** `Switch(config)# ip dhcp snooping information option`	Enables the switch to insert and remove DHCP relay information (option-82 field) in forwarded DHCP request messages to the DHCP server. This is the default setting.
Step 5	**ip dhcp snooping information option format remote-id** [**string** *ASCII-string* | **hostname**]	(Optional) Configures the remote-ID suboption. You can configure the remote ID as:

	Command or Action	Purpose
	Example: `Switch(config)# ip dhcp snooping information option format remote-id string acsiistring2`	• String of up to 63 ASCII characters (no spaces) • Configured hostname for the switch **Note** If the hostname is longer than 63 characters, it is truncated to 63 characters in the remote-ID configuration. The default remote ID is the switch MAC address.
Step 6	**ip dhcp snooping information option allow-untrusted** **Example:** `Switch(config)# ip dhcp snooping information option allow-untrusted`	(Optional) If the switch is an aggregation switch connected to an edge switch, this command enables the switch to accept incoming DHCP snooping packets with option-82 information from the edge switch. The default setting is disabled. **Note** Enter this command only on aggregation switches that are connected to trusted devices.
Step 7	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/1`	Specifies the interface to be configured, and enter interface configuration mode.
Step 8	**ip dhcp snooping vlan** *vlan* **information option format-type circuit-id** [**override**] **string** *ASCII-string* **Example:** `Switch(config-if)# ip dhcp snooping vlan 1 information option format-type curcuit-id override string ovrride2`	(Optional) Configures the circuit-ID suboption for the specified interface. Specify the VLAN and port identifier, using a VLAN ID in the range of 1 to 4094. The default circuit ID is the port identifier, in the format **vlan-mod-port**. You can configure the circuit ID to be a string of 3 to 63 ASCII characters (no spaces). (Optional) Use the **override** keyword when you do not want the circuit-ID suboption inserted in TLV format to define subscriber information.
Step 9	**ip dhcp snooping trust** **Example:** `Switch(config-if)# ip dhcp snooping trust`	(Optional) Configures the interface as trusted or untrusted. Use the **no** keyword to configure an interface to receive messages from an untrusted client. The default setting is untrusted.
Step 10	**ip dhcp snooping limit rate** *rate* **Example:** `Switch(config-if)# ip dhcp snooping limit rate 100`	(Optional) Configures the number of DHCP packets per second that an interface can receive. The range is 1 to 2048. By default, no rate limit is configured. **Note** We recommend an untrusted rate limit of not more than 100 packets per second. If you configure rate limiting for trusted interfaces, you might need to increase the rate limit if the port is a trunk port assigned to more than one VLAN with DHCP snooping.

	Command or Action	Purpose
Step 11	**exit** **Example:** `Switch(config-if)# exit`	Returns to global configuration mode.
Step 12	**ip dhcp snooping verify mac-address** **Example:** `Switch(config)# ip dhcp snooping verify mac-address`	(Optional) Configures the switch to verify that the source MAC address in a DHCP packet received on untrusted ports matches the client hardware address in the packet. The default is to verify that the source MAC address matches the client hardware address in the packet.
Step 13	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Enabling the Cisco IOS DHCP Server Database

For procedures to enable and configure the Cisco IOS DHCP server database, see the "DHCP Configuration Task List" section in the "Configuring DHCP" chapter of the Cisco IOS IP Configuration Guide, Release 12.4

Monitoring DHCP Snooping Information

Table 128: Commands for Displaying DHCP Information

show ip dhcp snooping	Displays the DHCP snooping configuration for a switch
show ip dhcp snooping binding	Displays only the dynamically configured bindings in the DHCP snooping binding database, also referred to as a binding table.
show ip dhcp snooping database	Displays the DHCP snooping binding database status and statistics.
show ip dhcp snooping statistics	Displays the DHCP snooping statistics in summary or detail form.
show ip source binding	Display the dynamically and statically configured bindings.

Note If DHCP snooping is enabled and an interface changes to the down state, the switch does not delete the statically configured bindings.

Configuring DHCP Server Port-Based Address Allocation

Information About Configuring DHCP Server Port-Based Address Allocation

DHCP server port-based address allocation is a feature that enables DHCP to maintain the same IP address on an Ethernet switch port regardless of the attached device client identifier or client hardware address.

When Ethernet switches are deployed in the network, they offer connectivity to the directly connected devices. In some environments, such as on a factory floor, if a device fails, the replacement device must be working immediately in the existing network. With the current DHCP implementation, there is no guarantee that DHCP would offer the same IP address to the replacement device. Control, monitoring, and other software expect a stable IP address associated with each device. If a device is replaced, the address assignment should remain stable even though the DHCP client has changed.

When configured, the DHCP server port-based address allocation feature ensures that the same IP address is always offered to the same connected port even as the client identifier or client hardware address changes in the DHCP messages received on that port. The DHCP protocol recognizes DHCP clients by the client identifier option in the DHCP packet. Clients that do not include the client identifier option are identified by the client hardware address. When you configure this feature, the port name of the interface overrides the client identifier or hardware address and the actual point of connection, the switch port, becomes the client identifier.

In all cases, by connecting the Ethernet cable to the same port, the same IP address is allocated through DHCP to the attached device.

The DHCP server port-based address allocation feature is only supported on a Cisco IOS DHCP server and not a third-party server.

Default Port-Based Address Allocation Configuration

By default, DHCP server port-based address allocation is disabled.

Port-Based Address Allocation Configuration Guidelines

- By default, DHCP server port-based address allocation is disabled.
- To restrict assignments from the DHCP pool to preconfigured reservations (unreserved addresses are not offered to the client and other clients are not served by the pool), you can enter the **reserved-only** DHCP pool configuration command.

Enabling the DHCP Snooping Binding Database Agent

Beginning in privileged EXEC mode, follow these steps to enable and configure the DHCP snooping binding database agent on the switch:

SUMMARY STEPS

1. **configure terminal**
2. **ip dhcp snooping database** {**flash**[*number*]**:**/*filename* | **ftp:**//*user***:***password@host*/*filename* | **http:**//[[*username***:***password*]**@**]{*hostname* | *host-ip*}[/*directory*] /*image-name***.tar** | **rcp:**//*user@host*/*filename*}| **tftp:**//*host*/*filename*
3. **ip dhcp snooping database timeout** *seconds*
4. **ip dhcp snooping database write-delay** *seconds*
5. **end**
6. **ip dhcp snooping binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* **expiry** *seconds*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip dhcp snooping database** {**flash**[*number*]**:**/*filename* | **ftp:**//*user***:***password@host*/*filename* | **http:**//[[*username***:***password*]**@**]{*hostname* | *host-ip*}[/*directory*] /*image-name***.tar** | **rcp:**//*user@host*/*filename*}| **tftp:**//*host*/*filename* **Example:** `Switch(config)# ip dhcp snooping database tftp://10.90.90.90/snooping-rp2`	Specifies the URL for the database agent or the binding file by using one of these forms: • **flash**[*number*]**:**/*filename* (Optional) Use the *number* parameter to specify the stack member number of the stack master. The range for *number* is 1 to 9. • **ftp:**//*user***:***password@host*/*filename* • **http:**//[[*username***:***password*]**@**]{*hostname* | *host-ip*}[/*directory*] /*image-name***.tar** • **rcp:**//*user@host*/*filename* • **tftp:**//*host*/*filename*
Step 3	**ip dhcp snooping database timeout** *seconds* **Example:** `Switch(config)# ip dhcp snooping database timeout 300`	Specifies (in seconds) how long to wait for the database transfer process to finish before stopping the process. The default is 300 seconds. The range is 0 to 86400. Use 0 to define an infinite duration, which means to continue trying the transfer indefinitely.
Step 4	**ip dhcp snooping database write-delay** *seconds* **Example:** `Switch(config)# ip dhcp snooping database write-delay 15`	Specifies the duration for which the transfer should be delayed after the binding database changes. The range is from 15 to 86400 seconds. The default is 300 seconds (5 minutes).

	Command or Action	Purpose
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 6	**ip dhcp snooping binding** *mac-address* **vlan** *vlan-id* *ip-address* **interface** *interface-id* **expiry** *seconds* **Example:** `Switch# ip dhcp snooping binding 0001.1234.1234 vlan 1 172.20.50.5 interface gi1/1 expiry 1000`	(Optional) Adds binding entries to the DHCP snooping binding database. The *vlan-id* range is from 1 to 4904. The *seconds* range is from 1 to 4294967295. Enter this command for each entry that you add. Use this command when you are testing or debugging the switch.

Enabling DHCP Server Port-Based Address Allocation

Beginning in privileged EXEC mode, follow these steps to globally enable port-based address allocation and to automatically generate a subscriber identifier on an interface.

SUMMARY STEPS

1. **configure terminal**
2. **ip dhcp use subscriber-id client-id**
3. **ip dhcp subscriber-id interface-name**
4. **interface** *interface-id*
5. **ip dhcp server use subscriber-id client-id**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip dhcp use subscriber-id client-id** **Example:** `Switch(config)# ip dhcp use subscriber-id client-id`	Configures the DHCP server to globally use the subscriber identifier as the client identifier on all incoming DHCP messages.

	Command or Action	Purpose
Step 3	**ip dhcp subscriber-id interface-name** **Example:** `Switch(config)# ip dhcp subscriber-id interface-name`	Automatically generates a subscriber identifier based on the short name of the interface. A subscriber identifier configured on a specific interface takes precedence over this command.
Step 4	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Specifies the interface to be configured, and enter interface configuration mode.
Step 5	**ip dhcp server use subscriber-id client-id** **Example:** `Switch(config-if)# ip dhcp server use subscriber-id client-id`	Configures the DHCP server to use the subscriber identifier as the client identifier on all incoming DHCP messages on the interface.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Monitoring DHCP Server Port-Based Address Allocation

Table 129: Commands for Displaying DHCP Port-Based Address Allocation Information

Command	Purpose
show interface *interface id*	Displays the status and configuration of a specific interface.
show ip dhcp pool	Displays the DHCP address pools.
show ip dhcp binding	Displays address bindings on the Cisco IOS DHCP server.

Additional References

Related Documents

Related Topic	Document Title
DHCP Configuration Information and Procedures	IP Addressing: DHCP Configuration Guide, Cisco IOS XE Release 3S http://www.cisco.com/en/US/docs/ios-xml/ios/ipaddr_dhcp/configuration/xe-3s/dhcp-xe-3s-book.html

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

CHAPTER 53

Configuring IP Source Guard

IP Source Guard (IPSG) is a security feature that restricts IP traffic on nonrouted, Layer 2 interfaces by filtering traffic based on the DHCP snooping binding database and on manually configured IP source bindings.

This chapter contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information About IP Source Guard

IP Source Guard

You can use IP source guard to prevent traffic attacks if a host tries to use the IP address of its neighbor and you can enable IP source guard when DHCP snooping is enabled on an untrusted interface.

After IPSG is enabled on an interface, the switch blocks all IP traffic received on the interface except for DHCP packets allowed by DHCP snooping.

The switch uses a source IP lookup table in hardware to bind IP addresses to ports. For IP and MAC filtering, a combination of source IP and source MAC lookups are used. IP traffic with a source IP address is the binding table is allowed, all other traffic is denied.

The IP source binding table has bindings that are learned by DHCP snooping or are manually configured (static IP source bindings). An entry in this table has an IP address, its associated MAC address, and its associated VLAN number. The switch uses the IP source binding table only when IP source guard is enabled.

IPSG is supported only on Layer 2 ports, including access and trunk ports. You can configure IPSG with source IP address filtering or with source IP and MAC address filtering.

IP Source Guard for Static Hosts

Note Do not use IPSG (IP source guard) for static hosts on uplink ports or trunk ports.

IPSG for static hosts extends the IPSG capability to non-DHCP and static environments. The previous IPSG used the entries created by DHCP snooping to validate the hosts connected to a switch. Any traffic received from a host without a valid DHCP binding entry is dropped. This security feature restricts IP traffic on nonrouted Layer 2 interfaces. It filters traffic based on the DHCP snooping binding database and on manually configured IP source bindings. The previous version of IPSG required a DHCP environment for IPSG to work.

IPSG for static hosts allows IPSG to work without DHCP. IPSG for static hosts relies on IP device tracking-table entries to install port ACLs. The switch creates static entries based on ARP requests or other IP packets to maintain the list of valid hosts for a given port. You can also specify the number of hosts allowed to send traffic to a given port. This is equivalent to port security at Layer 3.

IPSG for static hosts also supports dynamic hosts. If a dynamic host receives a DHCP-assigned IP address that is available in the IP DHCP snooping table, the same entry is learned by the IP device tracking table. In a stacked environment, when the master failover occurs, the IP source guard entries for static hosts attached to member ports are retained. When you enter the **show ip device tracking all** EXEC command, the IP device tracking table displays the entries as ACTIVE.

Note Some IP hosts with multiple network interfaces can inject some invalid packets into a network interface. The invalid packets contain the IP or MAC address for another network interface of the host as the source address. The invalid packets can cause IPSG for static hosts to connect to the host, to learn the invalid IP or MAC address bindings, and to reject the valid bindings. Consult the vender of the corresponding operating system and the network interface to prevent the host from injecting invalid packets.

IPSG for static hosts initially learns IP or MAC bindings dynamically through an ACL-based snooping mechanism. IP or MAC bindings are learned from static hosts by ARP and IP packets. They are stored in the device tracking database. When the number of IP addresses that have been dynamically learned or statically configured on a given port reaches a maximum, the hardware drops any packet with a new IP address. To resolve hosts that have moved or gone away for any reason, IPSG for static hosts leverages IP device tracking to age out dynamically learned IP address bindings. This feature can be used with DHCP snooping. Multiple bindings are established on a port that is connected to both DHCP and static hosts. For example, bindings are stored in both the device tracking database as well as in the DHCP snooping binding database.

IP Source Guard Configuration Guidelines

- You can configure static IP bindings only on nonrouted ports. If you enter the **ip source binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* global configuration command on a routed interface, this error message appears:

```
Static IP source binding can only be configured on switch port.
```

- When IP source guard with source IP filtering is enabled on an interface, DHCP snooping must be enabled on the access VLAN for that interface.
- If you are enabling IP source guard on a trunk interface with multiple VLANs and DHCP snooping is enabled on all the VLANs, the source IP address filter is applied on all the VLANs.

Note If IP source guard is enabled and you enable or disable DHCP snooping on a VLAN on the trunk interface, the switch might not properly filter traffic.

- You can enable this feature when 802.1x port-based authentication is enabled.
- When you configure IP source guard smart logging, packets with a source address other than the specified address or an address learned by DHCP are denied, and the packet contents are sent to a NetFlow collector. If you configure this feature, make sure that smart logging is globally enabled.
- In a switch stack, if IP source guard is configured on a stack member interface and you remove the the configuration of that switch by entering the **no switch** *stack-member-number* **provision** global configuration command, the interface static bindings are removed from the binding table, but they are not removed from the running configuration. If you again provision the switch by entering the **switch** *stack-member-number* **provision** command, the binding is restored.

  To remove the binding from the running configuration, you must disable IP source guard before entering the **no switch provision** command. The configuration is also removed if the switch reloads while the interface is removed from the binding table.

How to Configure IP Source Guard

Enabling IP Source Guard

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip verify source** [**mac-check**]
4. **exit**
5. **ip source binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Specifies the interface to be configured, and enters interface configuration mode.
Step 3	**ip verify source** [**mac-check**] **Example:** `Switch(config-if)# ip verify source`	Enables IP source guard with source IP address filtering. (Optional) **mac-check—**Enables IP Source Guard with source IP address and MAC address filtering.
Step 4	**exit** **Example:** `Switch(config-if)# exit`	Returns to global configuration mode.
Step 5	**ip source binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* **Example:** `Switch(config)# ip source binding 0100.0230.0002 vlan 11 10.0.0.4 interface gigabitethernet1/0/1`	Adds a static IP source binding. Enter this command for each static binding.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Enabling IP source guard with source IP and MAC filtering on VLANs 10 and 11

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet 1/0/1
Switch(config-if)# ip verify source
Switch(config-if)# exit
Switch(config)# ip source binding 0100.0022.0010 vlan 10 10.0.0.2 interface gigabitethernet
 1/0/1
```

```
Switch(config)# ip source binding 0100.0230.0002 vlan 11 10.0.0.4 interface gigabitethernet
 1/0/1
Switch(config)# end
```

Configuring IP Source Guard for Static Hosts on a Layer 2 Access Port

You must configure the **ip device tracking maximum** *limit-number* interface configuration command globally for IPSG for static hosts to work. If you only configure this command on a port without enabling IP device tracking globally or by setting an IP device tracking maximum on that interface, IPSG with static hosts rejects all the IP traffic from that interface.

SUMMARY STEPS

1. **configure terminal**
2. **ip device tracking**
3. **interface** *interface-id*
4. **switchport mode access**
5. **switchport access vlan** *vlan-id*
6. **ip verify source**[**tracking**] [**mac-check**]
7. **ip device tracking maximum** *number*
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip device tracking** **Example:** `Switch(config)# ip device tracking`	Turns on the IP host table, and globally enables IP device tracking.
Step 3	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode.

	Command or Action	Purpose
Step 4	**switchport mode access** **Example:** `Switch(config-if)# switchport mode access`	Configures a port as access.
Step 5	**switchport access vlan** *vlan-id* **Example:** `Switch(config-if)# switchport access vlan 10`	Configures the VLAN for this port.
Step 6	**ip verify source**[**tracking**] [**mac-check**] **Example:** `Switch(config-if)# ip verify source tracking mac-check`	Enables IP source guard with source IP address filtering. (Optional) **tracking—**Enables IP source guard for static hosts. (Optional) **mac-check—**Enables MAC address filtering. The command **ip verify source tracking mac-check**enables IP source guard for static hosts with MAC address filtering.
Step 7	**ip device tracking maximum** *number* **Example:** `Switch(config-if)# ip device tracking maximum 8`	Establishes a maximum limit for the number of static IPs that the IP device tracking table allows on the port. The range is 1to 10. The maximum number is 10. **Note** You must configure the **ip device tracking maximum** *limit-number* interface configuration command.
Step 8	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Eight Examples

This example shows how to stop IPSG with static hosts on an interface.

```
Switch(config-if)# no ip verify source
Switch(config-if)# no ip device tracking max
```

This example shows how to enable IPSG with static hosts on a port.

```
Switch(config)# ip device tracking
Switch(config-if)# ip device tracking maximum 10
Switch(config-if)# ip verify source tracking
```

This example shows how to enable IPSG for static hosts with IP filters on a Layer 2 access port and to verify the valid IP bindings on the interface Gi1/0/3:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip device tracking
Switch(config)# interface gigabitethernet1/0/3
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 10
Switch(config-if)# ip device tracking maximum 5
Switch(config-if)# ip verify source tracking
Switch(config-if)# end

Switch# show ip verify source
Interface  Filter-type  Filter-mode  IP-address       Mac-address        Vlan
---------  -----------  -----------  ---------------  -----------------  ----
Gi1/0/3      ip trk       active       40.1.1.24                           10
Gi1/0/3      ip trk       active       40.1.1.20                           10
Gi1/0/3      ip trk       active       40.1.1.21                           10
```

This example shows how to enable IPSG for static hosts with IP-MAC filters on a Layer 2 access port, to verify the valid IP-MAC bindings on the interface Gi1/0/3, and to verify that the number of bindings on this interface has reached the maximum:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip device tracking
Switch(config)# interface gigabitethernet1/0/3
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 1
Switch(config-if)# ip device tracking maximum 5

Switch(config-if)# ip verify source tracking
Switch(config-if)# end

Switch# show ip verify source
Interface  Filter-type  Filter-mode  IP-address       Mac-address        Vlan
---------  -----------  -----------  ---------------  -----------------  ----
Gi1/0/3      ip trk       active       deny-all                           1
```

This example displays all IP or MAC binding entries for all interfaces. The CLI displays all active as well as inactive entries. When a host is learned on a interface, the new entry is marked as active. When the same host is disconnected from that interface and connected to a different interface, a new IP or MAC binding entry displays as active as soon as the host is detected. The old entry for this host on the previous interface is marked as INACTIVE.

```
Switch# show ip device tracking all
IP Device Tracking for wireless clients = Enabled
Global IP Device Tracking for wired clients = Enabled
Global IP Device Tracking Probe Count = 3
Global IP Device Tracking Probe Interval = 30
-----------------------------------------------------------------------------------------

  IP Address     MAC Address   Vlan  Interface           Probe-Timeout      STATE
-----------------------------------------------------------------------------------------
200.1.1.8       0001.0600.0000  8   GigabitEthernet1/0/1                   INACTIVE
200.1.1.9       0001.0600.0000  8   GigabitEthernet1/0/1                   INACTIVE
200.1.1.10      0001.0600.0000  8   GigabitEthernet1/0/1                   INACTIVE
200.1.1.1       0001.0600.0000  9   GigabitEthernet1/0/2                   ACTIVE
200.1.1.1       0001.0600.0000  8   GigabitEthernet1/0/1                   INACTIVE
200.1.1.2       0001.0600.0000  9   GigabitEthernet1/0/2                   ACTIVE
200.1.1.2       0001.0600.0000  8   GigabitEthernet1/0/1                   INACTIVE
200.1.1.3       0001.0600.0000  9   GigabitEthernet1/0/2                   ACTIVE
200.1.1.3       0001.0600.0000  8   GigabitEthernet1/0/1                   INACTIVE
200.1.1.4       0001.0600.0000  9   GigabitEthernet1/0/2                   ACTIVE
200.1.1.4       0001.0600.0000  8   GigabitEthernet1/0/1                   INACTIVE
```

```
200.1.1.5       0001.0600.0000  9   GigabitEthernet1/0/2                  ACTIVE
200.1.1.5       0001.0600.0000  8   GigabitEthernet1/0/1                  INACTIVE
200.1.1.6       0001.0600.0000  8   GigabitEthernet1/0/1                  INACTIVE
200.1.1.7       0001.0600.0000  8   GigabitEthernet1/0/1                  INACTIVE
```

This example displays all active IP or MAC binding entries for all interfaces:

```
Switch# show ip device tracking all active
IP Device Tracking for wireless clients = Enabled
Global IP Device Tracking for wired clients = Enabled
Global IP Device Tracking Probe Count = 3
Global IP Device Tracking Probe Interval = 30
-----------------------------------------------------------------------------------------

  IP Address     MAC Address   Vlan  Interface           Probe-Timeout      STATE
-----------------------------------------------------------------------------------------
200.1.1.1       0001.0600.0000  9    GigabitEthernet1/0/1                 ACTIVE
200.1.1.2       0001.0600.0000  9    GigabitEthernet1/0/1                 ACTIVE
200.1.1.3       0001.0600.0000  9    GigabitEthernet1/0/1                 ACTIVE
200.1.1.4       0001.0600.0000  9    GigabitEthernet1/0/1                 ACTIVE
200.1.1.5       0001.0600.0000  9    GigabitEthernet1/0/1                 ACTIVE
```

This example displays all inactive IP or MAC binding entries for all interfaces. The host was first learned on GigabitEthernet 1/0/1 and then moved to GigabitEthernet 0/2. the IP or MAC binding entries learned on GigabitEthernet1/ 0/1 are marked as inactive.

```
Switch# show ip device tracking all inactive
IP Device Tracking for wireless clients = Enabled
Global IP Device Tracking for wired clients= Enabled
Global IP Device Tracking Probe Count = 3
Global IP Device Tracking Probe Interval = 30
-----------------------------------------------------------------------------------------

  IP Address     MAC Address   Vlan  Interface           Probe-Timeout      STATE
-----------------------------------------------------------------------------------------
200.1.1.8       0001.0600.0000  8   GigabitEthernet1/0/1                  INACTIVE
200.1.1.9       0001.0600.0000  8   GigabitEthernet1/0/1                  INACTIVE
200.1.1.10      0001.0600.0000  8   GigabitEthernet1/0/1                  INACTIVE
200.1.1.1       0001.0600.0000  8   GigabitEthernet1/0/1                  INACTIVE
200.1.1.2       0001.0600.0000  8   GigabitEthernet1/0/1                  INACTIVE
200.1.1.3       0001.0600.0000  8   GigabitEthernet1/0/1                  INACTIVE
200.1.1.4       0001.0600.0000  8   GigabitEthernet1/0/1                  INACTIVE
200.1.1.5       0001.0600.0000  8   GigabitEthernet1/0/1                  INACTIVE
200.1.1.6       0001.0600.0000  8   GigabitEthernet1/0/1                  INACTIVE
200.1.1.7       0001.0600.0000  8   GigabitEthernet1/0/1                  INACTIVE
```

This example displays the count of all IP device tracking host entries for all interfaces:

```
Switch# show ip device tracking all count
Total IP Device Tracking Host entries: 5
---------------------------------------------------------------------
  Interface           Maximum Limit          Number of Entries
---------------------------------------------------------------------
Gi1/0/3                 5
```

Monitoring IP Source Guard

Table 130: Privileged EXEC show Commands

Command	Purpose
show ip verify source [**interface** *interface-id*]	Displays the IP source guard configuration on the switch or on a specific interface.
show ip device tracking { **all** | **interface** *interface-id* | **ip** *ip-address* | **mac** *imac-address*}	Displays information about the entries in the IP device tracking table.

Table 131: Interface Configuration Commands

Command	Purpose
ip verify source tracking	Verifies the data source.

For detailed information about the fields in these displays, see the command reference for this release.

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

CHAPTER 54

Configuring Dynamic ARP Inspection

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Restrictions for Dynamic ARP Inspection

This section lists the restrictions and guidelines for configuring Dynamic ARP Inspection on the switch.

- Dynamic ARP inspection is an ingress security feature; it does not perform any egress checking.
- Dynamic ARP inspection is not effective for hosts connected to switches that do not support dynamic ARP inspection or that do not have this feature enabled. Because man-in-the-middle attacks are limited to a single Layer 2 broadcast domain, separate the domain with dynamic ARP inspection checks from the one with no checking. This action secures the ARP caches of hosts in the domain enabled for dynamic ARP inspection.
- Dynamic ARP inspection depends on the entries in the DHCP snooping binding database to verify IP-to-MAC address bindings in incoming ARP requests and ARP responses. Make sure to enable DHCP snooping to permit ARP packets that have dynamically assigned IP addresses.

  When DHCP snooping is disabled or in non-DHCP environments, use ARP ACLs to permit or to deny packets.
- Dynamic ARP inspection is supported on access ports, trunk ports, and EtherChannel ports.

  Note Do not enable Dynamic ARP inspection on RSPAN VLANs. If Dynamic ARP inspection is enabled on RSPAN VLANs, Dynamic ARP inspection packets might not reach the RSPAN destination port.
- A physical port can join an EtherChannel port channel only when the trust state of the physical port and the channel port match. Otherwise, the physical port remains suspended in the port channel. A port channel inherits its trust state from the first physical port that joins the channel. Consequently, the trust state of the first physical port need not match the trust state of the channel.

  Conversely, when you change the trust state on the port channel, the switch configures a new trust state on all the physical ports that comprise the channel.
- The rate limit is calculated separately on each switch in a switch stack. For a cross-stack EtherChannel, this means that the actual rate limit might be higher than the configured value. For example, if you set the rate limit to 30 pps on an EtherChannel that has one port on switch 1 and one port on switch 2, each port can receive packets at 29 pps without causing the EtherChannel to become error-disabled.
- The operating rate for the port channel is cumulative across all the physical ports within the channel. For example, if you configure the port channel with an ARP rate-limit of 400 pps, all the interfaces combined on the channel receive an aggregate 400 pps. The rate of incoming ARP packets on EtherChannel ports is equal to the sum of the incoming rate of packets from all the channel members. Configure the rate limit for EtherChannel ports only after examining the rate of incoming ARP packets on the channel-port members.

  The rate of incoming packets on a physical port is checked against the port-channel configuration rather than the physical-ports configuration. The rate-limit configuration on a port channel is independent of the configuration on its physical ports.

  If the EtherChannel receives more ARP packets than the configured rate, the channel (including all physical ports) is placed in the error-disabled state.
- Make sure to limit the rate of ARP packets on incoming trunk ports. Configure trunk ports with higher rates to reflect their aggregation and to handle packets across multiple dynamic ARP inspection-enabled VLANs. You also can use the ip arp inspection limit none interface configuration command to make the rate unlimited. A high rate-limit on one VLAN can cause a denial-of-service attack to other VLANs when the software places the port in the error-disabled state.

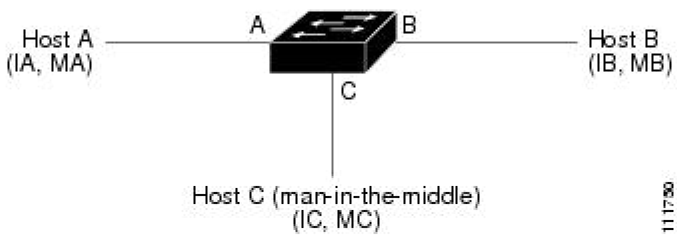
- When you enable dynamic ARP inspection on the switch, policers that were configured to police ARP traffic are no longer effective. The result is that all ARP traffic is sent to the CPU.

Understanding Dynamic ARP Inspection

ARP provides IP communication within a Layer 2 broadcast domain by mapping an IP address to a MAC address. For example, Host B wants to send information to Host A but does not have the MAC address of Host A in its ARP cache. Host B generates a broadcast message for all hosts within the broadcast domain to obtain the MAC address associated with the IP address of Host A. All hosts within the broadcast domain receive the ARP request, and Host A responds with its MAC address. However,because ARP allows a gratuitous reply from a host even if an ARP request was not received, an ARP spoofing attack and the poisoning of ARP caches can occur. After the attack, all traffic from the device under attack flows through the attacker's computer and then to the router, switch, or host.

A malicious user can attack hosts, switches, and routers connected to your Layer 2 network by poisoning the ARP caches of systems connected to the subnet and by intercepting traffic intended for other hosts on the subnet. Figure 26-1 shows an example of ARP cache poisoning.

Figure 48: ARP Cache Poisoning



Hosts A, B, and C are connected to the switch on interfaces A, B and C, all of which are on the same subnet. Their IP and MAC addresses are shown in parentheses; for example, Host A uses IP address IA and MAC address MA. When Host A needs to communicate to Host B at the IP layer, it broadcasts an ARP request for the MAC address associated with IP address IB. When the switch and Host B receive the ARP request, they populate their ARP caches with an ARP binding for a host with the IP address IA and a MAC address MA; for example, IP address IA is bound to MAC address MA. When Host B responds, the switch and Host A populate their ARP caches with a binding for a host with the IP address IB and the MAC address MB.

Host C can poison the ARP caches of the switch, Host A, and Host B by broadcasting forged ARP responses with bindings for a host with an IP address of IA (or IB) and a MAC address of MC. Hosts with poisoned ARP caches use the MAC address MC as the destination MAC address for traffic intended for IA or IB. This means that Host C intercepts that traffic. Because Host C knows the true MAC addresses associated with IA and IB, it can forward the intercepted traffic to those hosts by using the correct MAC address as the destination. Host C has inserted itself into the traffic stream from Host A to Host B, the classic *man-in-the middle*attack.

Dynamic ARP inspection is a security feature that validates ARP packets in a network. It intercepts, logs,and discards ARP packets with invalid IP-to-MAC address bindings. This capability protects the network from certain man-in-the-middle attacks.

Dynamic ARP inspection ensures that only valid ARP requests and responses are relayed. The switch performs these activities:

- Intercepts all ARP requests and responses on untrusted ports

- Verifies that each of these intercepted packets has a valid IP-to-MAC address binding before updating the local ARP cache or before forwarding the packet to the appropriate destination
- Drops invalid ARP packets

Dynamic ARP inspection determines the validity of an ARP packet based on valid IP-to-MAC address bindings stored in a trusted database, the DHCP snooping binding database. This database is built by DHCP snooping if DHCP snooping is enabled on the VLANs and on the switch. If the ARP packet is received on a trusted interface, the switch forwards the packet without any checks. On untrusted interfaces, the switch forwards the packet only if it is valid.

You enable dynamic ARP inspection on a per-VLAN basis by using the **ip arp inspection vlan** *vlan-range* global configuration command.

In non-DHCP environments, dynamic ARP inspection can validate ARP packets against user-configured ARP access control lists (ACLs) for hosts with statically configured IP addresses. You define an ARP ACL by using the **arp access-list** *acl-name* global configuration command.

You can configure dynamic ARP inspection to drop ARP packets when the IP addresses in the packets are invalid or when the MAC addresses in the body of the ARP packets do not match the addresses specified in the Ethernet header. Use the **ip arp inspection validate** {[**src-mac**] [**dst-mac**] [**ip**]} global configuration command.

Interface Trust States and Network Security

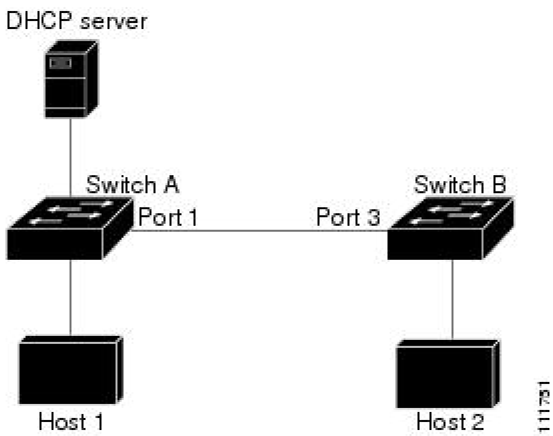
Dynamic ARP inspection associates a trust state with each interface on the switch. Packets arriving on trusted interfaces bypass all dynamic ARP inspection validation checks, and those arriving on untrusted interfaces undergo the dynamic ARP inspection validation process.

In a typical network configuration, you configure all switch ports connected to host ports as untrusted and configure all switch ports connected to switches as trusted. With this configuration, all ARP packets entering the network from a given switch bypass the security check. No other validation is needed at any other place in the VLAN or in the network. You configure the trust setting by using theip arp inspection trust interface configuration command.

Caution Use the trust state configuration carefully. Configuring interfaces as untrusted when they should betrusted can result in a loss of connectivity.

In the following figure, assume that both Switch A and Switch B are running dynamic ARP inspection on the VLAN that includes Host 1 and Host 2. If Host 1 and Host 2 acquire their IP addresses from the DHCP server connected to Switch A, only Switch A binds the IP-to-MAC address of Host 1. Therefore, if the interface between Switch A and Switch B is untrusted, the ARP packets from Host 1 are dropped by Switch B. Connectivity between Host 1 and Host 2 is lost.

Figure 49: ARP Packet Validation on a VLAN Enabled for Dynamic ARP Inspection



Configuring interfaces to be trusted when they are actually untrusted leaves a security hole in the network. If Switch A is not running dynamic ARP inspection, Host 1 can easily poison the ARP cache of Switch B (and Host 2, if the link between the switches is configured as trusted). This condition can occur even though Switch B is running dynamic ARP inspection.

Dynamic ARP inspection ensures that hosts (on untrusted interfaces) connected to a switch running dynamic ARP inspection do not poison the ARP caches of other hosts in the network. However, dynamic ARP inspection does not prevent hosts in other portions of the network from poisoning the caches of the hosts that are connected to a switch running dynamic ARP inspection.

In cases in which some switches in a VLAN run dynamic ARP inspection and other switches do not, configure the interfaces connecting such switches as untrusted. However, to validate the bindings of packets from nondynamic ARP inspection switches, configure the switch running dynamic ARP inspection with ARP ACLs. When you cannot determine such bindings, at Layer 3, isolate switches running dynamic ARP inspection from switches not running dynamic ARP inspection switches.

Note Depending on the setup of the DHCP server and the network, it might not be possible to validate a given ARP packet on all switches in the VLAN.

Rate Limiting of ARP Packets

The switch CPU performs dynamic ARP inspection validation checks; therefore, the number of incoming ARP packets is rate-limited to prevent a denial-of-service attack. By default, the rate for untrusted interfaces is 15 packets per second (pps). Trusted interfaces are not rate-limited. You can change this setting by using theip arp inspection limitinterface configuration command.

When the rate of incoming ARP packets exceeds the configured limit, the switch places the port in the error-disabled state. The port remains in that state until you intervene. You can use the errdisable recovery global configuration command to enable error disable recovery so that ports automatically emerge from this state after a specified timeout period.

Note The rate limit for an EtherChannel is applied separately to each switch in a stack. For example, if a limit of 20 pps is configured on the EtherChannel, each switch with ports in the EtherChannel can carry up to 20 pps. If any switch exceeds the limit, the entire EtherChannel is placed into the error-disabled state.

Relative Priority of ARP ACLs and DHCP Snooping Entries

Dynamic ARP inspection uses the DHCP snooping binding database for the list of valid IP-to-MAC address bindings.

ARP ACLs take precedence over entries in the DHCP snooping binding database. The switch uses ACLs only if you configure them by using the ip arp inspection filter vlan global configuration command. The switch first compares ARP packets to user-configured ARP ACLs. If the ARP ACL denies the ARP packet, the switch also denies the packet even if a valid binding exists in the database populated by DHCP snooping.

Logging of Dropped Packets

When the switch drops a packet, it places an entry in the log buffer and then generates system messages on a rate-controlled basis. After the message is generated, the switch clears the entry from the log buffer. Each log entry contains flow information, such as the receiving VLAN, the port number, the source and destination IP addresses, and the source and destination MAC addresses.

You use the **ip arp inspection log-buffer** global configuration command to configure the number of entries in the buffer and the number of entries needed in the specified interval to generate system messages. You specify the type of packets that are logged by using the **ip arp inspection vlan logging** global configuration command.

Default Dynamic ARP Inspection Configuration

Feature	Default Settings
Dynamic ARP inspection	Disabled on all VLANs.
Interface trust state	All interfaces are untrusted.
Feature	The rate is 15 pps on untrusted interfaces, assuming that the network is a switched network with a host connecting to as many as 15 new hosts per second. The rate is unlimited on all trusted interfaces. The burst interval is 1 second.
Dynamic ARP inspection	No ARP ACLs are defined.
Interface trust state	No checks are performed.

Feature	Default Settings
Rate limit of incoming ARP packets	When dynamic ARP inspection is enabled, all denied or dropped ARP packets are logged. The number of entries in the log is 32. The number of system messages is limited to 5 per second. The logging-rate interval is 1 second.
ARP ACLs for non-DHCP environments	All denied or dropped ARP packets are logged.

Relative Priority of ARP ACLs and DHCP Snooping Entries

Dynamic ARP inspection uses the DHCP snooping binding database for the list of valid IP-to-MAC address bindings.

ARP ACLs take precedence over entries in the DHCP snooping binding database. The switch uses ACLs only if you configure them by using the ip arp inspection filter vlan global configuration command. The switch first compares ARP packets to user-configured ARP ACLs. If the ARP ACL denies the ARP packet, the switch also denies the packet even if a valid binding exists in the database populated by DHCP snooping.

Configuring ARP ACLs for Non-DHCP Environments

This procedure shows how to configure dynamic ARP inspection when Switch B shown in Figure 2 does not support dynamic ARP inspection or DHCP snooping.

If you configure port 1 on Switch A as trusted, a security hole is created because both Switch A and Host 1 could be attacked by either Switch B or Host 2. To prevent this possibility, you must configure port 1 on Switch A as untrusted. To permit ARP packets from Host 2, you must set up an ARP ACL and apply it to VLAN 1. If the IP address of Host 2 is not static (it is impossible to apply the ACL configuration on Switch A) you must separate Switch A from Switch B at Layer 3 and use a router to route packets between them.

Beginning in privileged EXEC mode, follow these steps to configure an ARP ACL on Switch A. This procedure is required in non-DHCP environments.

SUMMARY STEPS

1. **Configureterminal**
2. **arp access-list** *acl-name*
3. **permit ip host** *sender-ip* **mac host** *sender-mac*
4. exit
5. **ip arp inspection filter** *arp-acl-name* **vlan** *vlan-range* [**static**]
6. **interface** *interface-id*
7. **no ip arp inspection trust**
8. **end**
9. **show arp access-list** acl-name **show ip arp inspection vlan** *vlan-range* **show ip arp inspection interfaces**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**Configureterminal**	Enter global configuration mode.
Step 2	**arp access-list** *acl-name*	Define an ARP ACL, and enter ARP access-list configuration mode. By default, no ARP access lists are defined. **Note** At the end of the ARP access list, there is an implicit**deny ip any mac any** command.
Step 3	**permit ip host** *sender-ip* **mac host** *sender-mac*	Permit ARP packets from the specified host (Host 2). • For*sender-ip*, enter the IP address of Host 2. • For *sender-mac*, enter the MAC address of Host 2.
Step 4	exit	Return to global configuration mode.
Step 5	**ip arp inspection filter** *arp-acl-name* **vlan** *vlan-range* [**static**]	Apply the ARP ACL to the VLAN. By default, no defined ARP ACLs are applied to any VLAN. • For *arp-acl-name*, specify the name of the ACL created in Step 2. • For *vlan-range*, specify the VLAN that the switches and hosts are in. You can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • (Optional) Specify **static** to treat implicit denies in the ARP ACL as explicit denies and to drop packets that do not match any previous clauses in the ACL. DHCP bindings are not used. If you do not specify this keyword, it means that there is no explicit deny in the ACL that denies the packet, and DHCP bindings determine whether a packet is permitted or denied if the packet does not match any clauses in the ACL.

	Command or Action	Purpose
		ARP packets containing only IP-to-MAC address bindings are compared against the ACL. Packets are permitted only if the access list permits them.
Step 6	**interface** *interface-id*	Specify the Switch A interface that is connected to Switch B, and enter interface configuration mode.
Step 7	**no ip arp inspection trust**	Configure the Switch A interface that is connected to Switch B as untrusted. By default, all interfaces are untrusted. For untrusted interfaces, the switch intercepts all ARP requests and responses. It verifies that the intercepted packets have valid IP-to-MAC address bindings before updating the local cache and before forwarding the packet to the appropriate destination. The switch drops invalid packets and logs them in the log buffer according to the logging configuration specified with the ip arp inspection vlan logging global configuration command.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	**show arp access-list** acl-name **show ip arp inspection vlan** *vlan-range* **show ip arp inspection interfaces**	Verify your entries.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the ARP ACL, use the **no arp access-list** global configuration command. To remove the ARP ACL attached to a VLAN, use the **no ip arp inspection filter** *arp-acl-name* **vlan** *vlan-range* global configuration command.

This example shows how to configure an ARP ACL called host2 on Switch A, to permit ARP packets from Host 2 (IP address 1.1.1.1 and MAC address 0001.0001.0001), to apply the ACL to VLAN 1, and to configure port 1 on Switch A as untrusted:

```
Switch(config)#arp access-list host2
 Switch(config-arp-acl)#permit ip host 1.1.1.1 mac host 1.1.1
 Switch(config-arp-acl)# exit
Switch(config)# ip arp inspection filter host2 vlan 1

Switch(config)# interface gigabitethernet1/0/1

 Switch(config-if)# no ip arp inspection trust
```

Configuring Dynamic ARP Inspection in DHCP Environments

Before You Begin

This procedure shows how to configure dynamic ARP inspection when two switches support this feature. Host 1 is connected to Switch A, and Host 2 is connected to Switch B. Both switches are running dynamic ARP inspection on VLAN 1 where the hosts are located. A DHCP server is connected to Switch A. Both hosts acquire their IP addresses from the same DHCP server. Therefore, Switch A has the bindings for Host 1 and Host 2, and Switch B has the binding for Host 2.

Note Dynamic ARP inspection depends on the entries in the DHCP snooping binding database to verify IP-to-MAC address bindings in incoming ARP requests and ARP responses. Make sure to enable DHCP snooping to permit ARP packets that have dynamically assigned IP addresses.

Beginning in privileged EXEC mode, follow these steps to configure dynamic ARP inspection. You must perform this procedure on both switches. This procedure is required.

SUMMARY STEPS

1. **show cdp neighbors**
2. **configure terminal**
3. **ip arp inspection vlan** *vlan-range*
4. **Interface***interface-id*
5. **ip arp inspection trust**
6. **end**
7. **show ip arp inspection interfaces**
8. **show ip arp inspection vlan** *vlan-range*
9. **show ip dhcp snooping binding**
10. **show ip arp inspection statistics vlan** *vlan-range*
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show cdp neighbors**	Verify the connection between the switches.
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 3	**ip arp inspection vlan** *vlan-range* **Example:**	Enable dynamic ARP inspection on a per-VLAN basis. By default, dynamic ARP inspection is disabled on all VLANs. For vlan-range, specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. Specify the same VLAN ID for both switches.
Step 4	**Interface***interface-id* **Example:**	Specify the interface connected to the other switch, and enter interface configuration mode.
Step 5	**ip arp inspection trust**	Configure the connection between the switches as trusted.

	Command or Action	Purpose
	Example:	By default, all interfaces are untrusted. The switch does not check ARP packets that it receives from the other switch on the trusted interface. It simply forwards the packets. For untrusted interfaces, the switch intercepts all ARP requests and responses. It verifies that the intercepted packets have valid IP-to-MAC address bindings before updating the local cache and before forwarding the packet to the appropriate destination. The switch drops invalid packets and logs them in the log buffer according to the logging configuration specified with the ip arp inspection vlan logging global configuration command.
Step 6	**end** **Example:**	Return to privileged EXEC mode.
Step 7	**show ip arp inspection interfaces** **Example:**	Verify the dynamic ARP inspection configuration on interfaces.
Step 8	**show ip arp inspection vlan** *vlan-range*	Verify the dynamic ARP inspection configuration on VLAN.
Step 9	**show ip dhcp snooping binding** **Example:**	Verify the DHCP bindings.
Step 10	**show ip arp inspection statistics vlan** *vlan-range* **Example:**	Check the dynamic ARP inspection statistics on VLAN.
Step 11	**copy running-config startup-config** **Example:**	(Optional) Save your entries in the configuration file.

To disable dynamic ARP inspection, use the **no ip arp inspection vlan** *vlan-range* global configuration command. To return the interfaces to an untrusted state, use the **no ip arp inspection trust** interface configuration command.

This example shows how to configure dynamic ARP inspection on Switch A in VLAN 1. You would perform a similar procedure on Switch B:

```
Switch(config)# ip arp inspection vlan 1

Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)#ip arp inspection trust
```

How to Limit the Rate of Incoming ARP Packets

The switch CPU performs dynamic ARP inspection validation checks; therefore, the number of incoming ARP packets is rate-limited to prevent a denial- of-service attack.

When the rate of incoming ARP packets exceeds the configured limit, the switch places the port in the error-disabled state. The port remains in that state until you enable error-disabled recovery so that ports automatically emerge from this state after a specified timeout period.

Note Unless you configure a rate limit on an interface, changing the trust state of the interface also changes its rate limit to the default value for that trust state. After you configure the rate limit, the interface retains the rate limit even when its trust state is changed. If you enter the no ip arp inspection limit interface configuration command, the interface reverts to its default rate limit.

For configuration guidelines for rate limiting trunk ports and EtherChannel ports, see the section, “Dynamic ARP Inspection Configuration Guidelines."

To return to the default rate-limit configuration, use the no ip arp inspection limit interface configuration command. To disable error recovery for dynamic ARP inspection, use the **no errdisable recovery cause arp-inspection** global configuration command.

Beginning in privileged EXEC mode, follow these steps to limit the rate of incoming ARP packets. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip arp inspection limit {rate pps [burst interval seconds] | none}**
4. **exit**
5. **errdisable detect cause arp-inspection** and **errdisable recovery causearp-inspection errdisable recovery interval** *interval*
6. **exit**
7. **show ip arp inspection interfaces show errdisable recovery**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be rate-limited, and enter interface configuration mode.
Step 3	**ip arp inspection limit {rate pps [burst interval seconds] | none}**	Limit the rate of incoming ARP requests and responses on the interface. Limit the rate of incoming ARP requests and responses on the interface. The default rate is 15 pps on untrusted interfaces and unlimited on trusted interfaces. The burst interval is 1 second. The keywords have these meanings: • For rate pps, specify an upper limit for the number of incoming packets processed per second. The range is 0 to 2048 pps.

	Command or Action	Purpose
		• (Optional) For burst interval seconds, specify the consecutive interval in seconds, over which the interface is monitored for a high rate of ARP packets. The range is 1 to 15. • For rate none, specify no upper limit for the rate of incoming ARP packets that can be processed.
Step 4	**exit**	Return to global configuration mode.
Step 5	**errdisable detect cause arp-inspection** and **errdisable recovery causearp-inspection errdisable recovery interval** *interval*	(Optional) Enable error recovery from the dynamic ARP inspection error-disabled state, and configure the dynamic ARP inspection recover mechanism variables. By default, recovery is disabled, and the recovery interval is 300 seconds. For interval interval, specify the time in seconds to recover from the error-disabled state. The range is 30 to 86400.
Step 6	**exit**	Return to privileged EXEC mode.
Step 7	**show ip arp inspection interfaces** **show errdisable recovery**	Verify your settings.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

How to Perform Validation Checks

Dynamic ARP inspection intercepts, logs, and discards ARP packets with invalid IP-to-MAC address bindings. You can configure the switch to perform additional checks on the destination MAC address, the sender and target IP addresses, and the source MAC address. Beginning in privileged EXEC mode, follow these steps to perform specific checks on incoming ARP packets.

This procedure is optional.

To disable checking, use the**no ip arp inspection validate [src-mac] [dst-mac] [ip]** global configuration command. To display statistics for forwarded, dropped, and MAC and IP validation failure packets, use the **show ip arp inspection statistics** privileged EXEC command.

SUMMARY STEPS

1. **configure terminal**
2. **ip arp inspection validate {[src-mac] [dst-mac] [ip]}**
3. **exit**
4. **show ip arp inspection vlan** *vlan-range*
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip arp inspection validate {[src-mac] [dst-mac] [ip]}**	Perform a specific check on incoming ARP packets. By default, no checks are performed. The keywords have these meanings: • For **src-mac**, check the source MAC address in the Ethernet header against the sender MAC address in the ARP body. This check is performed on both ARP requests and responses. When enabled, packets with different MAC addresses are classified as invalid and are dropped. • For **dst-mac**, check the destination MAC address in the Ethernet header against the target MAC address in ARP body. This check is performed for ARP responses. When enabled, packets with different MAC addresses are classified as invalid and are dropped. • For **ip**, check the ARP body for invalid and unexpected IP addresses. Addresses include 0.0.0.0, 255.255.255.255, and all IP multicast addresses. Sender IP addresses are checked in all ARP requests and responses, and target IP addresses are checked only in ARP responses. You must specify at least one of the keywords. Each command overrides the configuration of the previous command; that is, if a command enables src and dst mac validations, and a second command enables IP validation only, the src and dst mac validations are disabled as a result of the second command.
Step 3	**exit**	Return to privileged EXEC mode.
Step 4	**show ip arp inspection vlan** *vlan-range*	Verify your settings.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Monitoring DAI

To monitor DAI, use the following commands:

Command	Description
clear ip arp inspection statistics	Clears dynamic ARP inspection statistics.

Command	Description
show ip arp inspection statistics [**vlan** *vlan-range*]	Displays statistics for forwarded, dropped, MAC validation failure, IP validation failure, ACL permitted and denied, and DHCP permitted and denied packets for the specified VLAN. If no VLANs are specified or if a range is specified, displays information only for VLANs with dynamic ARP inspection enabled (active).
clear ip arp inspection log	Clears the dynamic ARP inspection log buffer.
show ip arp inspection log	Displays the configuration and contents of the dynamic ARP inspection log buffer.

For the **show ip arp inspection statistics** command, the switch increments the number of forwarded packets for each ARP request and response packet on a trusted dynamic ARP inspection port. The switch increments the number of ACL or DHCP permitted packets for each packet that is denied by source MAC, destination MAC, or IP validation checks, and the switch increments the appropriate.

Verifying the DAI Configuration

To display and verify the DAI configuration, use the following commands:

Command	Description
show arp access-list [*acl-name*]	Displays detailed information about ARP ACLs.
show ip arp inspection interfaces [interface-id]	Displays the trust state and the rate limit of ARP packets for the specified interface or all interfaces.
show ip arp inspection vlan *vlan-range*	Displays the configuration and the operating state of dynamic ARP inspection for the specified VLAN. If no VLANs are specified or if a range is specified, displays information only for VLANs with dynamic ARP inspection enabled (active).

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

CHAPTER 55

Configuring IEEE 802.1x Port-Based Authentication

This chapter describes how to configure IEEE 802.1x port-based authentication. IEEE 802.1x authentication prevents unauthorized devices (clients) from gaining access to the network. Unless otherwise noted, the term *switch* refers to a standalone switch or a switch stack.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information About 802.1x Port-Based Authentication

The 802.1x standard defines a client-server-based access control and authentication protocol that prevents unauthorized clients from connecting to a LAN through publicly accessible ports unless they are properly authenticated. The authentication server authenticates each client connected to a switch port before making available any services offered by the switch or the LAN.

Until the client is authenticated, 802.1x access control allows only Extensible Authentication Protocol over LAN (EAPOL), Cisco Discovery Protocol (CDP), and Spanning Tree Protocol (STP) traffic through the port to which the client is connected. After authentication is successful, normal traffic can pass through the port.

Note For complete syntax and usage information for the commands used in this chapter, see the "RADIUS Commands" section in the *Cisco IOS Security Command Reference, Release 12.4* and the command reference for this release.

Port-Based Authentication Process

When 802.1x port-based authentication is enabled and the client supports 802.1x-compliant client software, these events occur:

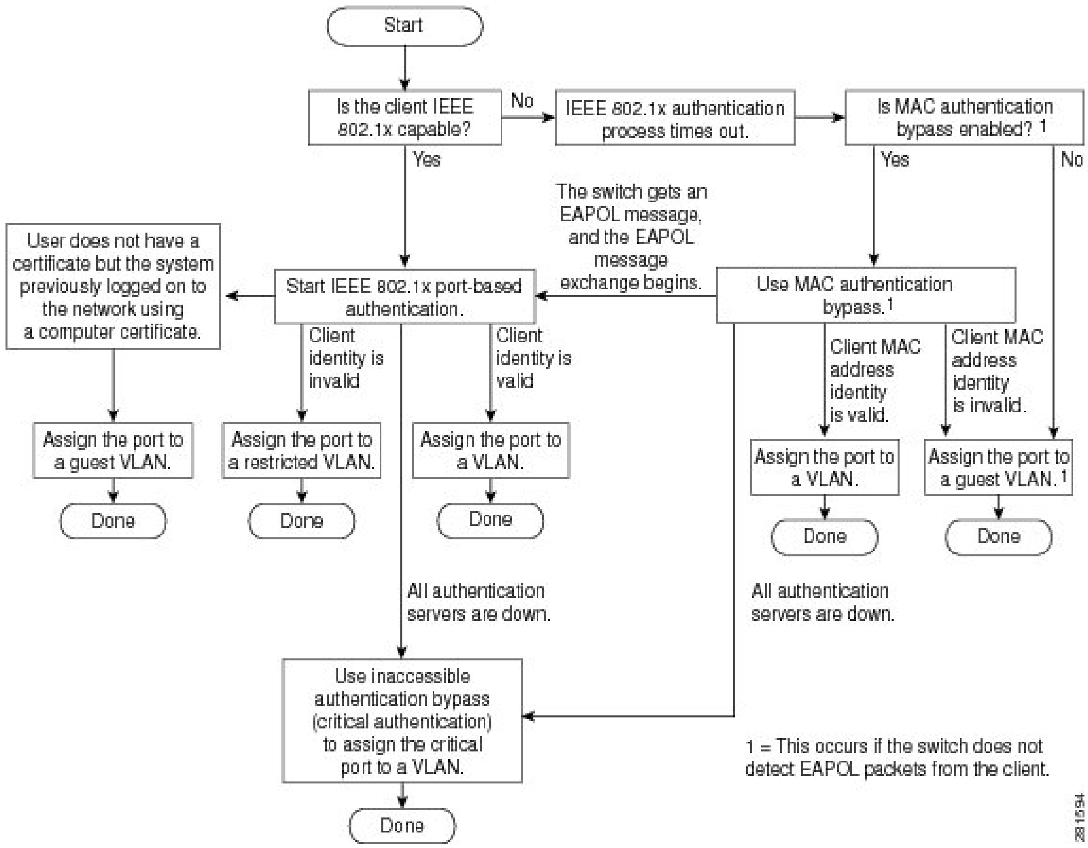
- If the client identity is valid and the 802.1x authentication succeeds, the switch grants the client access to the network.
- If 802.1x authentication times out while waiting for an EAPOL message exchange and MAC authentication bypass is enabled, the switch can use the client MAC address for authorization. If the client MAC address is valid and the authorization succeeds, the switch grants the client access to the network. If the client MAC address is invalid and the authorization fails, the switch assigns the client to a guest VLAN that provides limited services if a guest VLAN is configured.
- If the switch gets an invalid identity from an 802.1x-capable client and a restricted VLAN is specified, the switch can assign the client to a restricted VLAN that provides limited services.
- If the RADIUS authentication server is unavailable (down) and inaccessible authentication bypass is enabled, the switch grants the client access to the network by putting the port in the critical-authentication state in the RADIUS-configured or the user-specified access VLAN.

  Note Inaccessible authentication bypass is also referred to as critical authentication or the AAA fail policy.

If Multi Domain Authentication (MDA) is enabled on a port, this flow can be used with some exceptions that are applicable to voice authorization.

This figure shows the authentication process.

Figure 50: Authentication Flowchart



The switch re-authenticates a client when one of these situations occurs:

- Periodic re-authentication is enabled, and the re-authentication timer expires.

  You can configure the re-authentication timer to use a switch-specific value or to be based on values from the RADIUS server.

  After 802.1x authentication using a RADIUS server is configured, the switch uses timers based on the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute [29]).

  The Session-Timeout RADIUS attribute (Attribute[27]) specifies the time after which re-authentication occurs.

  The Termination-Action RADIUS attribute (Attribute [29]) specifies the action to take during re-authentication. The actions are *Initialize* and *ReAuthenticate*. When the *Initialize* action is set (the attribute value is *DEFAULT*), the 802.1x session ends, and connectivity is lost during re-authentication. When the *ReAuthenticate* action is set (the attribute value is RADIUS-Request), the session is not affected during re-authentication.

- You manually re-authenticate the client by entering the **dot1x re-authenticate interface** *interface-id* privileged EXEC command.

Port-Based Authentication Initiation and Message Exchange

During 802.1x authentication, the switch or the client can initiate authentication. If you enable authentication on a port by using the **authentication port-control auto** interface configuration command, the switch initiates authentication when the link state changes from down to up or periodically as long as the port remains up and unauthenticated. The switch sends an EAP-request/identity frame to the client to request its identity. Upon receipt of the frame, the client responds with an EAP-response/identity frame.

However, if during bootup, the client does not receive an EAP-request/identity frame from the switch, the client can initiate authentication by sending an EAPOL-start frame, which prompts the switch to request the client's identity.

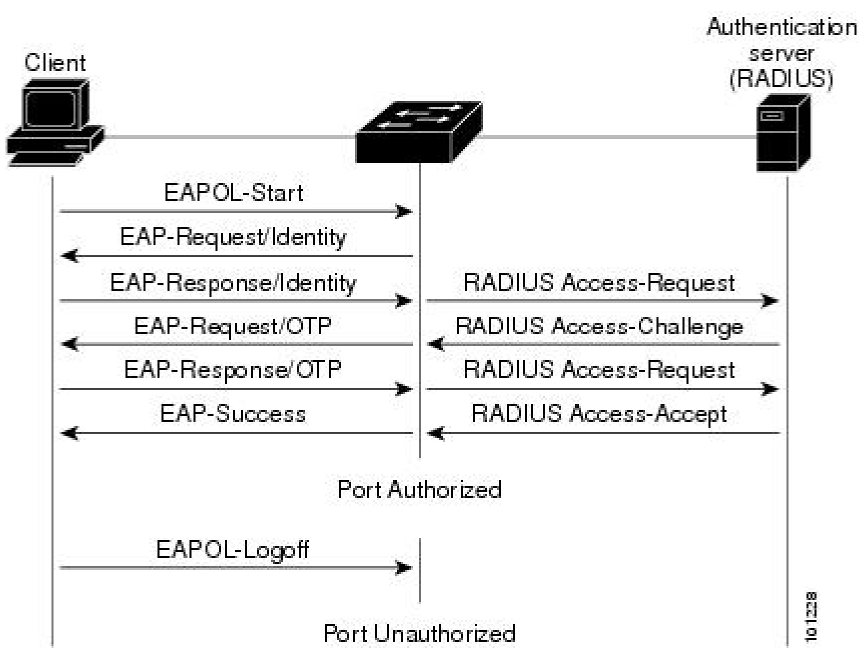
Note If 802.1x authentication is not enabled or supported on the network access device, any EAPOL frames from the client are dropped. If the client does not receive an EAP-request/identity frame after three attempts to start authentication, the client sends frames as if the port is in the authorized state. A port in the authorized state effectively means that the client has been successfully authenticated.

When the client supplies its identity, the switch begins its role as the intermediary, passing EAP frames between the client and the authentication server until authentication succeeds or fails. If the authentication succeeds, the switch port becomes authorized. If the authentication fails, authentication can be retried, the port might be assigned to a VLAN that provides limited services, or network access is not granted.

The specific exchange of EAP frames depends on the authentication method being used.

This figure shows a message exchange initiated by the client when the client uses the One-Time-Password (OTP) authentication method with a RADIUS server.

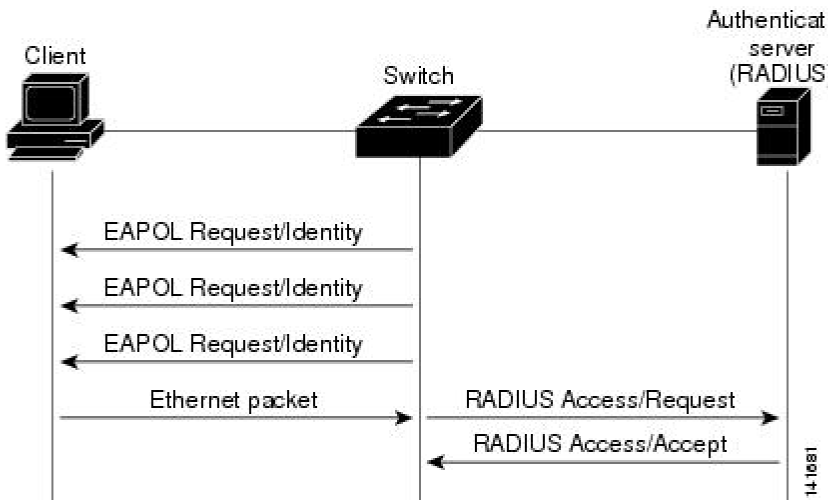
Figure 51: Message Exchange



If 802.1x authentication times out while waiting for an EAPOL message exchange and MAC authentication bypass is enabled, the switch can authorize the client when the switch detects an Ethernet packet from the client. The switch uses the MAC address of the client as its identity and includes this information in the RADIUS-access/request frame that is sent to the RADIUS server. After the server sends the switch the RADIUS-access/accept frame (authorization is successful), the port becomes authorized. If authorization fails and a guest VLAN is specified, the switch assigns the port to the guest VLAN. If the switch detects an EAPOL packet while waiting for an Ethernet packet, the switch stops the MAC authentication bypass process and starts 802.1x authentication.

This figure shows the message exchange during MAC authentication bypass.

Figure 52: Message Exchange During MAC Authentication Bypass



Authentication Manager for Port-Based Authentication

In Cisco IOS Release 12.2(46)SE and earlier, you could not use the same authorization methods, including CLI commands and messages, on this switch and also on other network devices, such as a Catalyst 6000. You had to use separate authentication configurations. Cisco IOS Release 12.2(50)SE and later supports the same authorization methods on all Catalyst switches in a network.

Cisco IOS Release 12.2(55)SE supports filtering verbose system messages from the authentication manager.

Port-Based Authentication Methods

Table 132: 802.1x Features

Authentication method	Mode			
	Single host	**Multiple host**	**MDA**	**Multiple Authentication**
802.1x	VLAN assignment Per-user ACL Filter-ID attribute Downloadable ACL[20] Redirect URL	VLAN assignment	VLAN assignment Per-user ACL Filter-Id attribute Downloadable ACL Redirect URL	VLAN assignment Per-user ACL Filter-Id attribute Downloadable ACL Redirect URL
MAC authentication bypass	VLAN assignment Per-user ACL Filter-ID attribute Downloadable ACL Redirect URL	VLAN assignment	VLAN assignment Per-user ACL Filter-Id attribute Downloadable ACL Redirect URL	VLAN assignment Per-user ACL Filter-Id attribute Downloadable ACL Redirect URL
Standalone web authentication	Proxy ACL, Filter-Id attribute, downloadable ACL			
NAC Layer 2 IP validation	Filter-Id attribute Downloadable ACL Redirect URL	Filter-Id attribute Downloadable ACL Redirect URL	Filter-Id attribute Downloadable ACL Redirect URL	Filter-Id attribute Downloadable ACL Redirect URL
Web authentication as fallback method[21]	Proxy ACL Filter-Id attribute Downloadable ACL	Proxy ACL Filter-Id attribute Downloadable ACL	Proxy ACL Filter-Id attribute Downloadable ACL	Proxy ACL Filter-Id attribute Downloadable ACL

20 Supported in Cisco IOS Release 12.2(50)SE and later.

21 For clients that do not support 802.1x authentication.

Per-User ACLs and Filter-Ids

ACLs configured on the switch are compatible with other devices running Cisco IOS releases.

You can only set **any** as the source in the ACL.

Note: For any ACL configured for multiple-host mode, the source portion of statement must be *any*. (For example, **permit icmp** *any* **host 10.10.1.1**.)

Port-Based Authentication Manager CLI Commands

The authentication-manager interface-configuration commands control all the authentication methods, such as 802.1x, MAC authentication bypass, and web authentication. The authentication manager commands determine the priority and order of authentication methods applied to a connected host.

The authentication manager commands control generic authentication features, such as host-mode, violation mode, and the authentication timer. Generic authentication commands include the **authentication host-mode**, **authentication violation**, and **authentication timer** interface configuration commands.

802.1x-specific commands begin with the **dot1x** keyword. For example, the **authentication port-control auto** interface configuration command enables authentication on an interface. However, the **dot1x system-authentication control** global configuration command only globally enables or disables 802.1x authentication.

Note: If 802.1x authentication is globally disabled, other authentication methods are still enabled on that port, such as web authentication.

The **authentication manager** commands provide the same functionality as earlier 802.1x commands.

Beginning with Cisco IOS Release 12.2(55)SE, you can filter out verbose system messages generated by the authentication manager. The filtered content typically relates to authentication success. You can also filter verbose messages for 802.1x authentication and MAB authentication. There is a separate command for each authentication method:

- The **no authentication logging verbose** global configuration command filters verbose messages from the authentication manager.
- The **no dot1x logging verbose** global configuration command filters 802.1x authentication verbose messages.
- The **no mab logging verbose** global configuration command filters MAC authentication bypass (MAB) verbose messages

Table 133: Authentication Manager Commands and Earlier 802.1x Commands

The authentication manager commands in Cisco IOS Release 12.2(50)SE or later	The equivalent 802.1x commands in Cisco IOS Release 12.2(46)SE and earlier	Description
authentication control-direction {**both** | **in**}	**dot1x control-direction** {**both** | **in**}	Enable 802.1x authentication with the wake-on-LAN (WoL) feature, and configure the port control as unidirectional or bidirectional.

The authentication manager commands in Cisco IOS Release 12.2(50)SE or later	The equivalent 802.1x commands in Cisco IOS Release 12.2(46)SE and earlier	Description
authentication event	**dot1x auth-fail vlan** **dot1x critical (interface configuration)** **dot1x guest-vlan6**	Enable the restricted VLAN on a port. Enable the inaccessible-authentication-bypass feature. Specify an active VLAN as an 802.1x guest VLAN.
authentication fallback *fallback-profile*	**dot1x fallback** *fallback-profile*	Configure a port to use web authentication as a fallback method for clients that do not support 802.1x authentication.
authentication host-mode [**multi-auth** | **multi-domain** | **multi-host** | **single-host**]	**dot1x host-mode** {**single-host** | **multi-host** | **multi-domain**}	Allow a single host (client) or multiple hosts on an 802.1x-authorized port.
authentication order	**mab**	Provides the flexibility to define the order of authentication methods to be used.
authentication periodic	**dot1x reauthentication**	Enable periodic re-authentication of the client.
authentication port-control {**auto** | **force-authorized** | **force-un authorized**}	**dot1x port-control** {**auto** | **force-authorized** | **force-unauthorized**}	Enable manual control of the authorization state of the port.
authentication timer	**dot1x timeout**	Set the 802.1x timers.
authentication violation {**protect** | **restrict** | **shutdown**}	**dot1x violation-mode** {**shutdown** | **restrict** | **protect**}	Configure the violation modes that occur when a new device connects to a port or when a new device connects to a port after the maximum number of devices are connected to that port.
show authentication	**show dot1x**	Display 802.1x statistics, administrative status, and operational status for the switch or for the specified port. authentication manager: compatibility with earlier 802.1x CLI commands

Ports in Authorized and Unauthorized States

During 802.1x authentication, depending on the switch port state, the switch can grant a client access to the network. The port starts in the *unauthorized* state. While in this state, the port that is not configured as a voice VLAN port disallows all ingress and egress traffic except for 802.1x authentication, CDP, and STP packets. When a client is successfully authenticated, the port changes to the *authorized* state, allowing all traffic for the client to flow normally. If the port is configured as a voice VLAN port, the port allows VoIP traffic and 802.1x protocol packets before the client is successfully authenticated.

If a client that does not support 802.1x authentication connects to an unauthorized 802.1x port, the switch requests the client's identity. In this situation, the client does not respond to the request, the port remains in the unauthorized state, and the client is not granted access to the network.

In contrast, when an 802.1x-enabled client connects to a port that is not running the 802.1x standard, the client initiates the authentication process by sending the EAPOL-start frame. When no response is received, the client sends the request for a fixed number of times. Because no response is received, the client begins sending frames as if the port is in the authorized state.

You control the port authorization state by using the **authentication port-control** interface configuration command and these keywords:

- **force-authorized—**disables 802.1x authentication and causes the port to change to the authorized state without any authentication exchange required. The port sends and receives normal traffic without 802.1x-based authentication of the client. This is the default setting.
- **force-unauthorized—**causes the port to remain in the unauthorized state, ignoring all attempts by the client to authenticate. The switch cannot provide authentication services to the client through the port.
- **auto—**enables 802.1x authentication and causes the port to begin in the unauthorized state, allowing only EAPOL frames to be sent and received through the port. The authentication process begins when the link state of the port changes from down to up or when an EAPOL-start frame is received. The switch requests the identity of the client and begins relaying authentication messages between the client and the authentication server. Each client attempting to access the network is uniquely identified by the switch by using the client MAC address.

Note In Session Aware Networking mode, the **authentication port-control** command is **access-session port-control**.

If the client is successfully authenticated (receives an Accept frame from the authentication server), the port state changes to authorized, and all frames from the authenticated client are allowed through the port. If the authentication fails, the port remains in the unauthorized state, but authentication can be retried. If the authentication server cannot be reached, the switch can resend the request. If no response is received from the server after the specified number of attempts, authentication fails, and network access is not granted.

When a client logs off, it sends an EAPOL-logoff message, causing the switch port to change to the unauthorized state.

If the link state of a port changes from up to down, or if an EAPOL-logoff frame is received, the port returns to the unauthorized state.

Port-Based Authentication and Switch Stacks

If a switch is added to or removed from a switch stack, 802.1x authentication is not affected as long as the IP connectivity between the RADIUS server and the stack remains intact. This statement also applies if the stack master is removed from the switch stack. Note that if the stack master fails, a stack member becomes the new stack master by using the election process, and the 802.1x authentication process continues as usual.

If IP connectivity to the RADIUS server is interrupted because the switch that was connected to the server is removed or fails, these events occur:

- Ports that are already authenticated and that do not have periodic re-authentication enabled remain in the authenticated state. Communication with the RADIUS server is not required.
- Ports that are already authenticated and that have periodic re-authentication enabled (with the **authentication periodic** global configuration command) fail the authentication process when the re-authentication occurs. Ports return to the unauthenticated state during the re-authentication process. Communication with the RADIUS server is required.

  For an ongoing authentication, the authentication fails immediately because there is no server connectivity.

If the switch that failed comes up and rejoins the switch stack, the authentications might or might not fail depending on the boot-up time and whether the connectivity to the RADIUS server is re-established by the time the authentication is attempted.

To avoid loss of connectivity to the RADIUS server, you should ensure that there is a redundant connection to it. For example, you can have a redundant connection to the stack master and another to a stack member, and if the stack master fails, the switch stack still has connectivity to the RADIUS server.

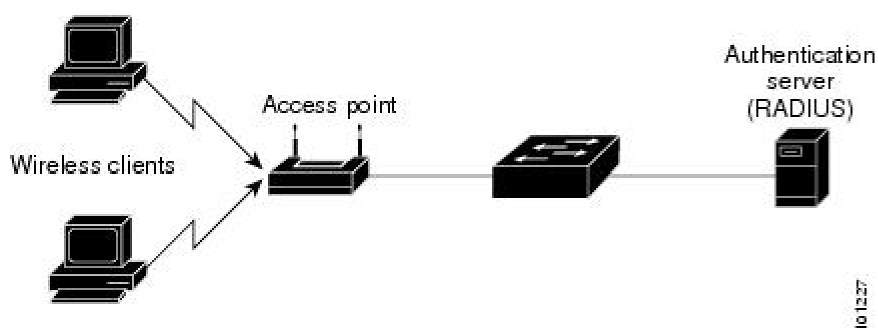
802.1x Host Mode

You can configure an 802.1x port for single-host or for multiple-hosts mode. In single-host mode, only one client can be connected to the 802.1x-enabled switch port. The switch detects the client by sending an EAPOL frame when the port link state changes to the up state. If a client leaves or is replaced with another client, the switch changes the port link state to down, and the port returns to the unauthorized state.

In multiple-hosts mode, you can attach multiple hosts to a single 802.1x-enabled port. In this mode, only one of the attached clients must be authorized for all clients to be granted network access. If the port becomes unauthorized (re-authentication fails or an EAPOL-logoff message is received), the switch denies network access to all of the attached clients. In this topology, the wireless access point is responsible for authenticating the clients attached to it, and it also acts as a client to the switch.

This figure shows 802.1x port-based authentication in a wireless LAN.

Figure 53: Multiple Host Mode Example



802.1x Multiple Authentication Mode

Multiple-authentication (multiauth) mode allows one client on the voice VLAN and multiple authenticated clients on the data VLAN. When a hub or access point is connected to an 802.1x-enabled port, multiple-authentication mode provides enhanced security over multiple-hosts mode by requiring authentication of each connected client. For non-802.1x devices, you can use MAC authentication bypass or web authentication as the fallback method for individual host authentications to authenticate different hosts through by different methods on a single port.

Multiple-authentication mode also supports MDA functionality on the voice VLAN by assigning authenticated devices to either a data or voice VLAN, depending on the VSAs received from the authentication server.

Note Guest VLAN and authentication-failed VLAN features are supported for ports configured in multiple-authentication mode.

Beginning with Cisco IOS Release 12.2(55)SE, you can assign a RADIUS-server-supplied VLAN in multi-auth mode, under these conditions:

- Only one voice VLAN assignment is supported on a multi-auth port.
- The behavior of the critical-auth VLAN is not changed for multi-auth mode. When a host tries to authenticate and the server is not reachable, all authorized hosts are reinitialized in the configured VLAN.

MAC Move

When a MAC address is authenticated on one switch port, that address is not allowed on another authentication manager-enabled port of the switch. If the switch detects that same MAC address on another authentication manager-enabled port, the address is not allowed.

There are situations where a MAC address might need to move from one port to another on the same switch. For example, when there is another device (for example a hub or an IP phone) between an authenticated host and a switch port, you might want to disconnect the host from the device and connect it directly to another port on the same switch.

You can globally enable MAC move so the device is reauthenticated on the new port. When a host moves to a second port, the session on the first port is deleted, and the host is reauthenticated on the new port.

MAC move is supported on all host modes. (The authenticated host can move to any port on the switch, no matter which host mode is enabled on the that port.)

When a MAC address moves from one port to another, the switch terminates the authenticated session on the original port and initiates a new authentication sequence on the new port.

The MAC move feature applies to both voice and data hosts.

Note In open authentication mode, a MAC address is immediately moved from the original port to the new port, with no requirement for authorization on the new port.

MAC Replace

Beginning with Cisco IOS Release 12.2(55)SE, the MAC replace feature can be configured to address the violation that occurs when a host attempts to connect to a port where another host was previously authenticated.

Note This feature does not apply to ports in multi-auth mode, because violations are not triggered in that mode. It does not apply to ports in multiple host mode, because in that mode, only the first host requires authentication.

If you configure the **authentication violation** interface configuration command with the **replace** keyword, the authentication process on a port in multi-domain mode is:

- A new MAC address is received on a port with an existing authenticated MAC address.
- The authentication manager replaces the MAC address of the current data host on the port with the new MAC address.
- The authentication manager initiates the authentication process for the new MAC address.
- If the authentication manager determines that the new host is a voice host, the original voice host is removed.

If a port is in open authentication mode, any new MAC address is immediately added to the MAC address table.

802.1x Accounting

The 802.1x standard defines how users are authorized and authenticated for network access but does not keep track of network usage. 802.1x accounting is disabled by default. You can enable 802.1x accounting to monitor this activity on 802.1x-enabled ports:

- User successfully authenticates.
- User logs off.
- Link-down occurs.
- Re-authentication successfully occurs.
- Re-authentication fails.

The switch does not log 802.1x accounting information. Instead, it sends this information to the RADIUS server, which must be configured to log accounting messages.

802.1x Accounting Attribute-Value Pairs

The information sent to the RADIUS server is represented in the form of Attribute-Value (AV) pairs. These AV pairs provide data for different applications. (For example, a billing application might require information that is in the Acct-Input-Octets or the Acct-Output-Octets attributes of a RADIUS packet.)

AV pairs are automatically sent by a switch that is configured for 802.1x accounting. Three types of RADIUS accounting packets are sent by a switch:

- START–sent when a new user session starts
- INTERIM–sent during an existing session for updates
- STOP–sent when a session terminates

You can view the AV pairs that are being sent by the switch by entering the **debug radius accounting** privileged EXEC command. For more information about this command, see the *Cisco IOS Debug Command Reference, Release 12.4*.

This table lists the AV pairs and when they are sent are sent by the switch.

Table 134: Accounting AV Pairs

Attribute Number	AV Pair Name	START	INTERIM	STOP
Attribute[1]	User-Name	Always	Always	Always
Attribute[4]	NAS-IP-Address	Always	Always	Always
Attribute[5]	NAS-Port	Always	Always	Always
Attribute[8]	Framed-IP-Address	Never	Sometimes[22]	Sometimes
Attribute[25]	Class	Always	Always	Always
Attribute[30]	Called-Station-ID	Always	Always	Always
Attribute[31]	Calling-Station-ID	Always	Always	Always
Attribute[40]	Acct-Status-Type	Always	Always	Always
Attribute[41]	Acct-Delay-Time	Always	Always	Always
Attribute[42]	Acct-Input-Octets	Never	Always	Always
Attribute[43]	Acct-Output-Octets	Never	Always	Always
Attribute[44]	Acct-Session-ID	Always	Always	Always
Attribute[45]	Acct-Authentic	Always	Always	Always
Attribute[46]	Acct-Session-Time	Never	Always	Always
Attribute[49]	Acct-Terminate-Cause	Never	Never	Always
Attribute[61]	NAS-Port-Type	Always	Always	Always

22 The Framed-IP-Address AV pair is sent only if a valid Dynamic Host Control Protocol (DHCP) binding exists for the host in the DHCP snooping bindings table.

802.1x Readiness Check

The 802.1x readiness check monitors 802.1x activity on all the switch ports and displays information about the devices connected to the ports that support 802.1x. You can use this feature to determine if the devices connected to the switch ports are 802.1x-capable. You use an alternate authentication such as MAC authentication bypass or web authentication for the devices that do not support 802.1x functionality.

This feature only works if the supplicant on the client supports a query with the NOTIFY EAP notification packet. The client must respond within the 802.1x timeout value.

The 802.1x readiness check is allowed on all ports that can be configured for 802.1x. The readiness check is not available on a port that is configured as **dot1x force-unauthorized**.

Follow these guidelines to enable the readiness check on the switch:

- The readiness check is typically used before 802.1x is enabled on the switch.
- If you use the **dot1x test eapol-capable** privileged EXEC command without specifying an interface, all the ports on the switch stack are tested.
- When you configure the **dot1x test eapol-capable** command on an 802.1x-enabled port, and the link comes up, the port queries the connected client about its 802.1x capability. When the client responds with a notification packet, it is 802.1x-capable. A syslog message is generated if the client responds within the timeout period. If the client does not respond to the query, the client is not 802.1x-capable. No syslog message is generated.
- The readiness check can be sent on a port that handles multiple hosts (for example, a PC that is connected to an IP phone). A syslog message is generated for each of the clients that respond to the readiness check within the timer period.

Related Topics

Configuring 802.1x Readiness Check, on page 1167

Switch-to-RADIUS-Server Communication

RADIUS security servers are identified by their hostname or IP address, hostname and specific UDP port numbers, or IP address and specific UDP port numbers. The combination of the IP address and UDP port number creates a unique identifier, which enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address. If two different host entries on the same RADIUS server are configured for the same service—for example, authentication—the second host entry configured acts as the fail-over backup to the first one. The RADIUS host entries are tried in the order that they were configured.

Related Topics

Configuring the Switch-to-RADIUS-Server Communication, on page 1175

802.1x Authentication with VLAN Assignment

The switch supports 802.1x authentication with VLAN assignment. After successful 802.1x authentication of a port, the RADIUS server sends the VLAN assignment to configure the switch port. The RADIUS server database maintains the username-to-VLAN mappings, assigning the VLAN based on the username of the client connected to the switch port. You can use this feature to limit network access for certain users.

Voice device authentication is supported with multidomain host mode. When a voice device is authorized and the RADIUS server returned an authorized VLAN, the voice VLAN on the port is configured to send and receive packets on the assigned voice VLAN. Voice VLAN assignment behaves the same as data VLAN assignment on multidomain authentication (MDA)-enabled ports.

When configured on the switch and the RADIUS server, 802.1x authentication with VLAN assignment has these characteristics:

- If no VLAN is supplied by the RADIUS server or if 802.1x authentication is disabled, the port is configured in its access VLAN after successful authentication. Recall that an access VLAN is a VLAN assigned to an access port. All packets sent from or received on this port belong to this VLAN.
- If 802.1x authentication is enabled but the VLAN information from the RADIUS server is not valid, authorization fails and configured VLAN remains in use. This prevents ports from appearing unexpectedly in an inappropriate VLAN because of a configuration error.

  Configuration errors could include specifying a VLAN for a routed port, a malformed VLAN ID, a nonexistent or internal (routed port) VLAN ID, an RSPAN VLAN, a shut down or suspended VLAN. In the case of a multidomain host port, configuration errors can also be due to an attempted assignment of a data VLAN that matches the configured or assigned voice VLAN ID (or the reverse).
- If 802.1x authentication is enabled and all information from the RADIUS server is valid, the authorized device is placed in the specified VLAN after authentication.
- If the multiple-hosts mode is enabled on an 802.1x port, all hosts are placed in the same VLAN (specified by the RADIUS server) as the first authenticated host.
- Enabling port security does not impact the RADIUS server-assigned VLAN behavior.
- If 802.1x authentication is disabled on the port, it is returned to the configured access VLAN and configured voice VLAN.

When the port is in the force authorized, force unauthorized, unauthorized, or shutdown state, it is put into the configured access VLAN.

If an 802.1x port is authenticated and put in the RADIUS server-assigned VLAN, any change to the port access VLAN configuration does not take effect. In the case of a multidomain host, the same applies to voice devices when the port is fully authorized with these exceptions:

- If the VLAN configuration change of one device results in matching the other device configured or assigned VLAN, authorization of all devices on the port is terminated and multidomain host mode is disabled until a valid configuration is restored where data and voice device configured VLANs no longer match.
- If a voice device is authorized and is using a downloaded voice VLAN, the removal of the voice VLAN configuration, or modifying the configuration value to *dot1p* or *untagged* results in voice device un-authorization and the disablement of multi-domain host mode.

The 802.1x authentication with VLAN assignment feature is not supported on trunk ports, dynamic ports, or with dynamic-access port assignment through a VLAN Membership Policy Server (VMPS).

To configure VLAN assignment you need to perform these tasks:

- Enable AAA authorization by using the **network** keyword to allow interface configuration from the RADIUS server.
- Enable 802.1x authentication. (The VLAN assignment feature is automatically enabled when you configure 802.1x authentication on an access port).
- Assign vendor-specific tunnel attributes in the RADIUS server. The RADIUS server must return these attributes to the switch:
  - [64] Tunnel-Type = VLAN
  - [65] Tunnel-Medium-Type = 802
  - [81] Tunnel-Private-Group-ID = VLAN name or VLAN ID

Attribute [64] must contain the value *VLAN* (type 13). Attribute [65] must contain the value *802* (type 6). Attribute [81] specifies the *VLAN name* or *VLAN ID* assigned to the IEEE 802.1x-authenticated user.

802.1x Authentication with Per-User ACLs

You can enable per-user access control lists (ACLs) to provide different levels of network access and service to an 802.1x-authenticated user. When the RADIUS server authenticates a user connected to an 802.1x port, it retrieves the ACL attributes based on the user identity and sends them to the switch. The switch applies the attributes to the 802.1x port for the duration of the user session. The switch removes the per-user ACL configuration when the session is over, if authentication fails, or if a link-down condition occurs. The switch does not save RADIUS-specified ACLs in the running configuration. When the port is unauthorized, the switch removes the ACL from the port.

You can configure router ACLs and input port ACLs on the same switch. However, a port ACL takes precedence over a router ACL. If you apply input port ACL to an interface that belongs to a VLAN, the port ACL takes precedence over an input router ACL applied to the VLAN interface. Incoming packets received on the port to which a port ACL is applied are filtered by the port ACL. Incoming routed packets received on other ports are filtered by the router ACL. Outgoing routed packets are filtered by the router ACL. To avoid configuration conflicts, you should carefully plan the user profiles stored on the RADIUS server.

RADIUS supports per-user attributes, including vendor-specific attributes. These vendor-specific attributes (VSAs) are in octet-string format and are passed to the switch during the authentication process. The VSAs used for per-user ACLs are inacl#<*n*> for the ingress direction and outacl#<*n*> for the egress direction. MAC ACLs are supported only in the ingress direction. The switch supports VSAs only in the ingress direction. It does not support port ACLs in the egress direction on Layer 2 ports.

Use only the extended ACL syntax style to define the per-user configuration stored on the RADIUS server. When the definitions are passed from the RADIUS server, they are created by using the extended naming convention. However, if you use the Filter-Id attribute, it can point to a standard ACL.

You can use the Filter-Id attribute to specify an inbound or outbound ACL that is already configured on the switch. The attribute contains the ACL number followed by *.in* for ingress filtering or *.out* for egress filtering. If the RADIUS server does not allow the *.in* or *.out* syntax, the access list is applied to the outbound ACL by default. Because of limited support of Cisco IOS access lists on the switch, the Filter-Id attribute is supported only for IP ACLs numbered 1 to 199 and 1300 to 2699 (IP standard and IP extended ACLs).

Only one 802.1x-authenticated user is supported on a port. If the multiple-hosts mode is enabled on the port, the per-user ACL attribute is disabled for the associated port.

The maximum size of the per-user ACL is 4000 ASCII characters but is limited by the maximum size of RADIUS-server per-user ACLs.

To configure per-user ACLs:

- Enable AAA authentication.
- Enable AAA authorization by using the **network** keyword to allow interface configuration from the RADIUS server.
- Enable 802.1x authentication.
- Configure the user profile and VSAs on the RADIUS server.
- Configure the 802.1x port for single-host mode.

Note Per-user ACLs are supported only in single-host mode.

802.1x Authentication with Downloadable ACLs and Redirect URLs

You can download ACLs and redirect URLs from a RADIUS server to the switch during 802.1x authentication or MAC authentication bypass of the host. You can also download ACLs during web authentication.

Note A downloadable ACL is also referred to as a *dACL*.

If more than one host is authenticated and the host is in single-host, MDA, or multiple-authentication mode, the switch changes the source address of the ACL to the host IP address.

You can apply the ACLs and redirect URLs to all the devices connected to the 802.1x-enabled port.

If no ACLs are downloaded during 802.1x authentication, the switch applies the static default ACL on the port to the host. On a voice VLAN port configured in multi-auth or MDA mode, the switch applies the ACL only to the phone as part of the authorization policies.

Beginning with Cisco IOS Release 12.2(55)SE, if there is no static ACL on a port, a dynamic auth-default ACL is created, and policies are enforced before dACLs are downloaded and applied.

Note The auth-default-ACL does not appear in the running configuration.

The auth-default ACL is created when at least one host with an authorization policy is detected on the port. The auth-default ACL is removed from the port when the last authenticated session ends. You can configure the auth-default ACL by using the **ip access-list extended auth-default-acl** global configuration command.

Note The auth-default-ACL does not support Cisco Discovery Protocol (CDP) bypass in the single host mode. You must configure a static ACL on the interface to support CDP bypass.

The 802.1x and MAB authentication methods support two authentication modes, *open* and *closed*. If there is no static ACL on a port in *closed* authentication mode:

- An auth-default-ACL is created.
- The auth-default-ACL allows only DHCP traffic until policies are enforced.
- When the first host authenticates, the authorization policy is applied without IP address insertion.
- When a second host is detected, the policies for the first host are refreshed, and policies for the first and subsequent sessions are enforced with IP address insertion.

If there is no static ACL on a port in *open* authentication mode:

- An auth-default-ACL-OPEN is created and allows all traffic.
- Policies are enforced with IP address insertion to prevent security breaches.

- Web authentication is subject to the auth-default-ACL-OPEN.

To control access for hosts with no authorization policy, you can configure a directive. The supported values for the directive are *open* and *default*. When you configure the *open* directive, all traffic is allowed. The *default* directive subjects traffic to the access provided by the port. You can configure the directive either in the user profile on the AAA server or on the switch. To configure the directive on the AAA server, use the **authz-directive =<open/default>** global command. To configure the directive on the switch, use the **epm access-control open** global configuration command.

Note The default value of the directive is *default*.

If a host falls back to web authentication on a port without a configured ACL:

- If the port is in open authentication mode, the auth-default-ACL-OPEN is created.
- If the port is in closed authentication mode, the auth-default-ACL is created.

The access control entries (ACEs) in the fallback ACL are converted to per-user entries. If the configured fallback profile does not include a fallback ACL, the host is subject to the auth-default-ACL associated with the port.

Note If you use a custom logo with web authentication and it is stored on an external server, the port ACL must allow access to the external server before authentication. You must either configure a static port ACL or change the auth-default-ACL to provide appropriate access to the external server.

Cisco Secure ACS and Attribute-Value Pairs for the Redirect URL

The switch uses these *cisco-av-pair* VSAs:

- url-redirect is the HTTP or HTTPS URL.
- url-redirect-acl is the switch ACL name or number.

The switch uses the CiscoSecure-defined-ACL attribute value pair to intercept an HTTP or HTTPS request from the end point. The switch then forwards the client web browser to the specified redirect address. The url-redirect AV pair on the Cisco Secure ACS contains the URL to which the web browser is redirected. The url-redirect-acl attribute value pair contains the name or number of an ACL that specifies the HTTP or HTTPS traffic to redirect.

Note

- Traffic that matches a permit ACE in the ACL is redirected.
- Define the URL redirect ACL and the default port ACL on the switch.

If a redirect URL is configured for a client on the authentication server, a default port ACL on the connected client switch port must also be configured

Cisco Secure ACS and Attribute-Value Pairs for Downloadable ACLs

You can set the CiscoSecure-Defined-ACL Attribute-Value (AV) pair on the Cisco Secure ACS with the RADIUS cisco-av-pair vendor-specific attributes (VSAs). This pair specifies the names of the downloadable ACLs on the Cisco Secure ACS with the #ACL#-IP-name-number attribute.

- The *name* is the ACL name.
- The *number* is the version number (for example, 3f783768).

If a downloadable ACL is configured for a client on the authentication server, a default port ACL on the connected client switch port must also be configured.

If the default ACL is configured on the switch and the Cisco Secure ACS sends a host-access-policy to the switch, it applies the policy to traffic from the host connected to a switch port. If the policy does not apply, the switch applies the default ACL. If the Cisco Secure ACS sends the switch a downloadable ACL, this ACL takes precedence over the default ACL that is configured on the switch port. However, if the switch receives an host access policy from the Cisco Secure ACS but the default ACL is not configured, the authorization failure is declared.

VLAN ID-based MAC Authentication

You can use VLAN ID-based MAC authentication if you wish to authenticate hosts based on a static VLAN ID instead of a downloadable VLAN. When you have a static VLAN policy configured on your switch, VLAN information is sent to an IAS (Microsoft) RADIUS server along with the MAC address of each host for authentication. The VLAN ID configured on the connected port is used for MAC authentication. By using VLAN ID-based MAC authentication with an IAS server, you can have a fixed number of VLANs in the network.

The feature also limits the number of VLANs monitored and handled by STP. The network can be managed as a fixed VLAN.

Note This feature is not supported on Cisco ACS Server. (The ACS server ignores the sent VLAN-IDs for new hosts and only authenticates based on the MAC address.)

802.1x Authentication with Guest VLAN

You can configure a guest VLAN for each 802.1x port on the switch to provide limited services to clients, such as downloading the 802.1x client. These clients might be upgrading their system for 802.1x authentication, and some hosts, such as Windows 98 systems, might not be IEEE 802.1x-capable.

When you enable a guest VLAN on an 802.1x port, the switch assigns clients to a guest VLAN when the switch does not receive a response to its EAP request/identity frame or when EAPOL packets are not sent by the client.

The switch maintains the EAPOL packet history. If an EAPOL packet is detected on the interface during the lifetime of the link, the switch determines that the device connected to that interface is an IEEE 802.1x-capable supplicant, and the interface does not change to the guest VLAN state. EAPOL history is cleared if the interface link status goes down. If no EAPOL packet is detected on the interface, the interface changes to the guest VLAN state.

If the switch is trying to authorize an 802.1x-capable voice device and the AAA server is unavailable, the authorization attempt fails, but the detection of the EAPOL packet is saved in the EAPOL history. When the AAA server becomes available, the switch authorizes the voice device. However, the switch no longer allows other devices access to the guest VLAN. To prevent this situation, use one of these command sequences:

- Enter the **authentication event no-response action authorize vlan** *vlan-id* interface configuration command to allow access to the guest VLAN.
- Enter the **shutdown** interface configuration command followed by the **no shutdown** interface configuration command to restart the port.

Use a restricted VLAN to allow clients that failed authentication access to the network by entering the **dot1x auth-fail vlan** *vlan-id* interface configuration command.

If devices send EAPOL packets to the switch during the lifetime of the link, the switch no longer allows clients that fail authentication access to the guest VLAN.

Note If an EAPOL packet is detected after the interface has changed to the guest VLAN, the interface reverts to an unauthorized state, and 802.1x authentication restarts.

When the switch port is moved to the guest VLAN, the number of allowed 802.1x-incapable hosts is determined by the configured host-mode. If an 802.1x-capable client joins the same port on which the guest VLAN is configured, the port is put into the unauthorized state in the user-configured access VLAN, and authentication is restarted.

Guest VLANs are supported on 802.1x ports in single host, multiple host, multi-auth and multi-domain modes.

You can configure any active VLAN except an RSPAN VLAN, a private VLAN, or a voice VLAN as an 802.1x guest VLAN. The guest VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.

The switch supports *MAC authentication bypass*. When MAC authentication bypass is enabled on an 802.1x port, the switch can authorize clients based on the client MAC address when IEEE 802.1x authentication times out while waiting for an EAPOL message exchange. After detecting a client on an 802.1x port, the switch waits for an Ethernet packet from the client. The switch sends the authentication server a RADIUS-access/request frame with a username and password based on the MAC address. If authorization succeeds, the switch grants the client access to the network. If authorization fails, the switch assigns the port to the guest VLAN if one is specified.

802.1x Authentication with Restricted VLAN

You can configure a restricted VLAN (also referred to as an *authentication failed VLAN*) for each IEEE 802.1x port on a switch stack or a switch to provide limited services to clients that cannot access the guest VLAN. These clients are 802.1x-compliant and cannot access another VLAN because they fail the authentication process. A restricted VLAN allows users without valid credentials in an authentication server (typically, visitors to an enterprise) to access a limited set of services. The administrator can control the services available to the restricted VLAN.

Note You can configure a VLAN to be both the guest VLAN and the restricted VLAN if you want to provide the same services to both types of users.

Without this feature, the client attempts and fails authentication indefinitely, and the switch port remains in the spanning-tree blocking state. With this feature, you can configure the switch port to be in the restricted VLAN after a specified number of authentication attempts (the default value is 3 attempts).

The authenticator counts the failed authentication attempts for the client. When this count exceeds the configured maximum number of authentication attempts, the port moves to the restricted VLAN. The failed attempt count increments when the RADIUS server replies with either an *EAP failure* or an empty response without an EAP packet. When the port moves into the restricted VLAN, the failed attempt counter resets.

Users who fail authentication remain in the restricted VLAN until the next re-authentication attempt. A port in the restricted VLAN tries to re-authenticate at configured intervals (the default is 60 seconds). If re-authentication fails, the port remains in the restricted VLAN. If re-authentication is successful, the port moves either to the configured VLAN or to a VLAN sent by the RADIUS server. You can disable re-authentication. If you do this, the only way to restart the authentication process is for the port to receive a *link down* or *EAP logoff* event. We recommend that you keep re-authentication enabled if a client might connect through a hub. When a client disconnects from the hub, the port might not receive the *link down* or *EAP logoff* event.

After a port moves to the restricted VLAN, a simulated EAP success message is sent to the client. This prevents clients from indefinitely attempting authentication. Some clients (for example, devices running Windows XP) cannot implement DHCP without EAP success.

Restricted VLANs are supported on 802.1x ports in all host modes and on Layer 2 ports.

You can configure any active VLAN except an RSPAN VLAN, a primary private VLAN, or a voice VLAN as an 802.1x restricted VLAN. The restricted VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.

Other security port features such as dynamic ARP Inspection, DHCP snooping, and IP source guard can be configured independently on a restricted VLAN.

802.1x Authentication with Inaccessible Authentication Bypass

Use the inaccessible authentication bypass feature, also referred to as *critical authentication* or the *AAA fail policy*, when the switch cannot reach the configured RADIUS servers and new hosts cannot be authenticated. You can configure the switch to connect those hosts to *critical ports.*

When a new host tries to connect to the critical port, that host is moved to a user-specified access VLAN, the *critical VLAN*. The administrator gives limited authentication to the hosts.

When the switch tries to authenticate a host connected to a critical port, the switch checks the status of the configured RADIUS server. If a server is available, the switch can authenticate the host. However, if all the RADIUS servers are unavailable, the switch grants network access to the host and puts the port in the *critical-authentication* state, which is a special case of the authentication state.

Inaccessible Authentication Bypass Support on Multiple-Authentication Ports

When a port is configured on any host mode and the AAA server is unavailable, the port is then configured to multi-host mode and moved to the critical VLAN. To support this inaccessible bypass on multiple-authentication (multiauth) ports, use the **authentication event server dead action reinitialize vlan** *vlan-id* command. When a new host tries to connect to the critical port, that port is reinitialized and all the connected hosts are moved to the user-specified access VLAN.

This command is supported on all host modes.

Inaccessible Authentication Bypass Authentication Results

The behavior of the inaccessible authentication bypass feature depends on the authorization state of the port:

- If the port is unauthorized when a host connected to a critical port tries to authenticate and all servers are unavailable, the switch puts the port in the critical-authentication state in the RADIUS-configured or user-specified access VLAN.
- If the port is already authorized and reauthentication occurs, the switch puts the critical port in the critical-authentication state in the current VLAN, which might be the one previously assigned by the RADIUS server.
- If the RADIUS server becomes unavailable during an authentication exchange, the current exchange times out, and the switch puts the critical port in the critical-authentication state during the next authentication attempt.

You can configure the critical port to reinitialize hosts and move them out of the critical VLAN when the RADIUS server is again available. When this is configured, all critical ports in the critical-authentication state are automatically re-authenticated.

Inaccessible Authentication Bypass Feature Interactions

Inaccessible authentication bypass interacts with these features:

- Guest VLAN—Inaccessible authentication bypass is compatible with guest VLAN. When a guest VLAN is enabled on 8021.x port, the features interact as follows:
  - If at least one RADIUS server is available, the switch assigns a client to a guest VLAN when the switch does not receive a response to its EAP request/identity frame or when EAPOL packets are not sent by the client.
  - If all the RADIUS servers are not available and the client is connected to a critical port, the switch authenticates the client and puts the critical port in the critical-authentication state in the RADIUS-configured or user-specified access VLAN.
  - If all the RADIUS servers are not available and the client is not connected to a critical port, the switch might not assign clients to the guest VLAN if one is configured.
  - If all the RADIUS servers are not available and if a client is connected to a critical port and was previously assigned to a guest VLAN, the switch keeps the port in the guest VLAN.
- Restricted VLAN—If the port is already authorized in a restricted VLAN and the RADIUS servers are unavailable, the switch puts the critical port in the critical-authentication state in the restricted VLAN.
- 802.1x accounting—Accounting is not affected if the RADIUS servers are unavailable.
-
- Voice VLAN—Inaccessible authentication bypass is compatible with voice VLAN, but the RADIUS-configured or user-specified access VLAN and the voice VLAN must be different.
- Remote Switched Port Analyzer (RSPAN)—Do not configure an RSPAN VLAN as the RADIUS-configured or user-specified access VLAN for inaccessible authentication bypass.

In a switch stack, the stack master checks the status of the RADIUS servers by sending keepalive packets. When the status of a RADIUS server changes, the stack master sends the information to the stack members. The stack members can then check the status of RADIUS servers when re-authenticating critical ports.

If the new stack master is elected, the link between the switch stack and RADIUS server might change, and the new stack immediately sends keepalive packets to update the status of the RADIUS servers. If the server status changes from *dead* to *alive*, the switch re-authenticates all switch ports in the critical-authentication state.

When a member is added to the stack, the stack master sends the member the server status.

802.1x Critical Voice VLAN

When an IP phone connected to a port is authenticated by the access control server (ACS), the phone is put into the voice domain. If the ACS is not reachable, the switch cannot determine if the device is a voice device. If the server is unavailable, the phone cannot access the voice network and therefore cannot operate.

For data traffic, you can configure inaccessible authentication bypass, or critical authentication, to allow traffic to pass through on the native VLAN when the server is not available. If the RADIUS authentication server is unavailable (down) and inaccessible authentication bypass is enabled, the switch grants the client access to the network and puts the port in the critical-authentication state in the RADIUS-configured or the user-specified access VLAN. When the switch cannot reach the configured RADIUS servers and new hosts cannot be authenticated, the switch connects those hosts to critical ports. A new host trying to connect to the critical port is moved to a user-specified access VLAN, the critical VLAN, and granted limited authentication.

You can enter the authentication event server dead action authorize voice interface configuration command to configure the critical voice VLAN feature. When the ACS does not respond, the port goes into critical authentication mode. When traffic coming from the host is tagged with the voice VLAN, the connected device (the phone) is put in the configured voice VLAN for the port. The IP phones learn the voice VLAN identification through CDP (Cisco devices) or through LLDP or DHCP.

You can configure the voice VLAN for a port by entering the **switchport voice vlan** *vlan-id* interface configuration command.

This feature is supported in multidomain and multi-auth host modes. Although you can enter the command when the switch in single-host or multi-host mode, the command has no effect unless the device changes to multidomain or multi-auth host mode.

802.1x User Distribution

You can configure 802.1x user distribution to load-balance users with the same group name across multiple different VLANs.

The VLANs are either supplied by the RADIUS server or configured through the switch CLI under a VLAN group name.

- Configure the RADIUS server to send more than one VLAN name for a user. The multiple VLAN names can be sent as part of the response to the user. The 802.1x user distribution tracks all the users in a particular VLAN and achieves load balancing by moving the authorized user to the least populated VLAN.
- Configure the RADIUS server to send a VLAN group name for a user. The VLAN group name can be sent as part of the response to the user. You can search for the selected VLAN group name among the VLAN group names that you configured by using the switch CLI. If the VLAN group name is found,

the corresponding VLANs under this VLAN group name are searched to find the least populated VLAN. Load balancing is achieved by moving the corresponding authorized user to that VLAN.

Note The RADIUS server can send the VLAN information in any combination of VLAN-IDs, VLAN names, or VLAN groups.

802.1x User Distribution Configuration Guidelines

- Confirm that at least one VLAN is mapped to the VLAN group.
- You can map more than one VLAN to a VLAN group.
- You can modify the VLAN group by adding or deleting a VLAN.
- When you clear an existing VLAN from the VLAN group name, none of the authenticated ports in the VLAN are cleared, but the mappings are removed from the existing VLAN group.
- If you clear the last VLAN from the VLAN group name, the VLAN group is cleared.
- You can clear a VLAN group even when the active VLANs are mapped to the group. When you clear a VLAN group, none of the ports or users that are in the authenticated state in any VLAN within the group are cleared, but the VLAN mappings to the VLAN group are cleared.

IEEE 802.1x Authentication with Voice VLAN Ports

A voice VLAN port is a special access port associated with two VLAN identifiers:

- VVID to carry voice traffic to and from the IP phone. The VVID is used to configure the IP phone connected to the port.
- PVID to carry the data traffic to and from the workstation connected to the switch through the IP phone. The PVID is the native VLAN of the port.

The IP phone uses the VVID for its voice traffic, regardless of the authorization state of the port. This allows the phone to work independently of IEEE 802.1x authentication.

In single-host mode, only the IP phone is allowed on the voice VLAN. In multiple-hosts mode, additional clients can send traffic on the voice VLAN after a supplicant is authenticated on the PVID. When multiple-hosts mode is enabled, the supplicant authentication affects both the PVID and the VVID.

Note If an IP phone and PC are connected to a switchport, and the port is configured in single- or multi-host mode, we do not recommend configuring that port in standalone MAC authentication bypass mode. We recommend only using MAC authentication bypass as a fallback method to 802.1x authentication with the timeout period set to the default of five seconds.

A voice VLAN port becomes active when there is a link, and the device MAC address appears after the first CDP message from the IP phone. Cisco IP phones do not relay CDP messages from other devices. As a result, if several IP phones are connected in series, the switch recognizes only the one directly connected to it. When

IEEE 802.1x authentication is enabled on a voice VLAN port, the switch drops packets from unrecognized IP phones more than one hop away.

When IEEE 802.1x authentication is enabled on a switch port, you can configure an access port VLAN that is also a voice VLAN.

Note If you enable IEEE 802.1x authentication on an access port on which a voice VLAN is configured and to which a Cisco IP Phone is connected, the Cisco IP phone loses connectivity to the switch for up to 30 seconds.

IEEE 802.1x Authentication with Port Security

In general, Cisco does not recommend enabling port security when IEEE 802.1x is enabled. Since IEEE 802.1x enforces a single MAC address per port (or per VLAN when MDA is configured for IP telephony), port security is redundant and in some cases may interfere with expected IEEE 802.1x operations.

IEEE 802.1x Authentication with Wake-on-LAN

The IEEE 802.1x authentication with wake-on-LAN (WoL) feature allows dormant PCs to be powered when the switch receives a specific Ethernet frame, known as the *magic packet*. You can use this feature in environments where administrators need to connect to systems that have been powered down.

When a host that uses WoL is attached through an IEEE 802.1x port and the host powers off, the IEEE 802.1x port becomes unauthorized. The port can only receive and send EAPOL packets, and WoL magic packets cannot reach the host. When the PC is powered off, it is not authorized, and the switch port is not opened.

When the switch uses IEEE 802.1x authentication with WoL, the switch forwards traffic to unauthorized IEEE 802.1x ports, including magic packets. While the port is unauthorized, the switch continues to block ingress traffic other than EAPOL packets. The host can receive packets but cannot send packets to other devices in the network.

Note If PortFast is not enabled on the port, the port is forced to the bidirectional state.

When you configure a port as unidirectional by using the **authentication control-direction in** interface configuration command, the port changes to the spanning-tree forwarding state. The port can send packets to the host but cannot receive packets from the host.

When you configure a port as bidirectional by using the **authentication control-direction both** interface configuration command, the port is access-controlled in both directions. The port does not receive packets from or send packets to the host.

IEEE 802.1x Authentication with MAC Authentication Bypass

You can configure the switch to authorize clients based on the client MAC address by using the MAC authentication bypass feature. For example, you can enable this feature on IEEE 802.1x ports connected to devices such as printers.

If IEEE 802.1x authentication times out while waiting for an EAPOL response from the client, the switch tries to authorize the client by using MAC authentication bypass.

When the MAC authentication bypass feature is enabled on an IEEE 802.1x port, the switch uses the MAC address as the client identity. The authentication server has a database of client MAC addresses that are allowed network access. After detecting a client on an IEEE 802.1x port, the switch waits for an Ethernet packet from the client. The switch sends the authentication server a RADIUS-access/request frame with a username and password based on the MAC address. If authorization succeeds, the switch grants the client access to the network. If authorization fails, the switch assigns the port to the guest VLAN if one is configured.

If the switch already authorized a port by using MAC authentication bypass and detects an IEEE 802.1x supplicant, the switch does not unauthorize the client connected to the port. When re-authentication occurs, the switch uses the authentication or re-authentication methods configured on the port, if the previous session ended because the Termination-Action RADIUS attribute value is DEFAULT.

Clients that were authorized with MAC authentication bypass can be re-authenticated. The re-authentication process is the same as that for clients that were authenticated with IEEE 802.1x. During re-authentication, the port remains in the previously assigned VLAN. If re-authentication is successful, the switch keeps the port in the same VLAN. If re-authentication fails, the switch assigns the port to the guest VLAN, if one is configured.

If re-authentication is based on the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute [29]) and if the Termination-Action RADIUS attribute (Attribute [29]) action is *Initialize* (the attribute value is *DEFAULT*), the MAC authentication bypass session ends, and connectivity is lost during re-authentication. If MAC authentication bypass is enabled and the IEEE 802.1x authentication times out, the switch uses the MAC authentication bypass feature to initiate re-authorization. For more information about these AV pairs, see RFC 3580, "IEEE 802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines."

MAC authentication bypass interacts with the features:

- IEEE 802.1x authentication—MAC authentication bypass and IEEE 802.1x authentication are configured independently on the port.
- Guest VLAN—If a client has an invalid MAC address identity, the switch assigns the client to a guest VLAN if one is configured.
- Restricted VLAN—This feature is not supported when the client connected to an IEEE 802.lx port is authenticated with MAC authentication bypass.
- Port security
- Voice VLAN
- VLAN Membership Policy Server (VMPS)—IEEE802.1x and VMPS are mutually exclusive.
- Network admission control (NAC) Layer 2 IP validation—This feature takes effect after an IEEE 802.1x port is authenticated with MAC authentication bypass, including hosts in the exception list.
- Network Edge Access Topology (NEAT)—MAB and NEAT are mutually exclusive. You cannot enable MAB when NEAT is enabled on an interface, and you cannot enable NEAT when MAB is enabled on an interface.

Network Admission Control Layer 2 IEEE 802.1x Validation

The switch supports the Network Admission Control (NAC) Layer 2 IEEE 802.1x validation, which checks the antivirus condition or *posture* of endpoint systems or clients before granting the devices network access. With NAC Layer 2 IEEE 802.1x validation, you can do these tasks:

- Download the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute[29]) from the authentication server.
- Set the number of seconds between re-authentication attempts as the value of the Session-Timeout RADIUS attribute (Attribute[27]) and get an access policy against the client from the RADIUS server.
- Set the action to be taken when the switch tries to re-authenticate the client by using the Termination-Action RADIUS attribute (Attribute[29]). If the value is the *DEFAULT* or is not set, the session ends. If the value is RADIUS-Request, the re-authentication process starts.
- View the NAC posture token, which shows the posture of the client, by using the **show authentication** privileged EXEC command.

Configuring NAC Layer 2 IEEE 802.1x validation is similar to configuring IEEE 802.1x port-based authentication except that you must configure a posture token on the RADIUS server.

Flexible Authentication Ordering

You can use flexible authentication ordering to configure the order of methods that a port uses to authenticate a new host. MAC authentication bypass and 802.1x can be the primary or secondary authentication methods, and web authentication can be the fallback method if either or both of those authentication attempts fail.

Related Topics

Configuring Flexible Authentication Ordering, on page 1213

Open1x Authentication

Open1x authentication allows a device access to a port before that device is authenticated. When open authentication is configured, a new host can pass traffic according to the access control list (ACL) defined on the port. After the host is authenticated, the policies configured on the RADIUS server are applied to that host.

You can configure open authentication with these scenarios:

- Single-host mode with open authentication–Only one user is allowed network access before and after authentication.
- MDA mode with open authentication–Only one user in the voice domain and one user in the data domain are allowed.
- Multiple-hosts mode with open authentication–Any host can access the network.
- Multiple-authentication mode with open authentication–Similar to MDA, except multiple hosts can be authenticated.

Note If open authentication is configured, it takes precedence over other authentication controls. This means that if you use the **authentication open** interface configuration command, the port will grant access to the host irrespective of the **authentication port-control** interface configuration command.

In Session Aware Networking mode, to enable open authentication, use **no access-session closed**. To disable open authentication, use **access-session closed**.

Related Topics

Configuring Open1x, on page 1214

Multidomain Authentication

The switch supports multidomain authentication (MDA), which allows both a data device and voice device, such as an IP phone (Cisco or non-Cisco), to authenticate on the same switch port. The port is divided into a data domain and a voice domain.

MDA does not enforce the order of device authentication. However, for best results, we recommend that a voice device is authenticated before a data device on an MDA-enabled port.

Follow these guidelines for configuring MDA:

- You must configure a switch port for MDA.
- You must configure the voice VLAN for the IP phone when the host mode is set to multidomain.
- Voice VLAN assignment on an MDA-enabled port is supported.

> **Note** You can assign a dynamic VLAN to a voice device on an MDA-enabled switch port, but the voice device fails authorization if a static voice VLAN configured on the switchport is the same as the dynamic VLAN assigned for the voice device in the RADIUS server.

- To authorize a voice device, the AAA server must be configured to send a Cisco Attribute-Value (AV) pair attribute with a value of *device-traffic-class=voice*. Without this value, the switch treats the voice device as a data device.
- The guest VLAN and restricted VLAN features only apply to the data devices on an MDA-enabled port. The switch treats a voice device that fails authorization as a data device.
- If more than one device attempts authorization on either the voice or the data domain of a port, it is error disabled.
- Until a device is authorized, the port drops its traffic. Non-Cisco IP phones or voice devices are allowed into both the data and voice VLANs. The data VLAN allows the voice device to contact a DHCP server to obtain an IP address and acquire the voice VLAN information. After the voice device starts sending on the voice VLAN, its access to the data VLAN is blocked.
- A voice device MAC address that is binding on the data VLAN is not counted towards the port security MAC address limit.
- You can use dynamic VLAN assignment from a RADIUS server only for data devices.
- MDA can use MAC authentication bypass as a fallback mechanism to allow the switch port to connect to devices that do not support IEEE 802.1x authentication.
- When a *data* or a *voice* device is detected on a port, its MAC address is blocked until authorization succeeds. If the authorization fails, the MAC address remains blocked for 5 minutes.
- If more than five devices are detected on the *data* VLAN or more than one voice device is detected on the *voice* VLAN while a port is unauthorized, the port is error disabled.

- When a port host mode is changed from single- or multihost to multidomain mode, an authorized data device remains authorized on the port. However, a Cisco IP phone that has been allowed on the port voice VLAN is automatically removed and must be reauthenticated on that port.
- Active fallback mechanisms such as guest VLAN and restricted VLAN remain configured after a port changes from single- or multihost mode to multidomain mode.
- Switching a port host mode from multidomain to single- or multihost mode removes all authorized devices from the port.
- If a data domain is authorized first and placed in the guest VLAN, non-IEEE 802.1x-capable voice devices need to tag their packets on the voice VLAN to trigger authentication.
- We do not recommend per-user ACLs with an MDA-enabled port. An authorized device with a per-user ACL policy might impact traffic on both the voice and data VLANs of the port. If used, only one device on the port should enforce per-user ACLs.

802.1x Supplicant and Authenticator Switches with Network Edge Access Topology (NEAT)

The Network Edge Access Topology (NEAT) feature extends identity to areas outside the wiring closet (such as conference rooms). This allows any type of device to authenticate on the port.

- 802.1x switch supplicant: You can configure a switch to act as a supplicant to another switch by using the 802.1x supplicant feature. This configuration is helpful in a scenario, where, for example, a switch is outside a wiring closet and is connected to an upstream switch through a trunk port. A switch configured with the 802.1x switch supplicant feature authenticates with the upstream switch for secure connectivity. Once the supplicant switch authenticates successfully the port mode changes from access to trunk.
- If the access VLAN is configured on the authenticator switch, it becomes the native VLAN for the trunk port after successful authentication.

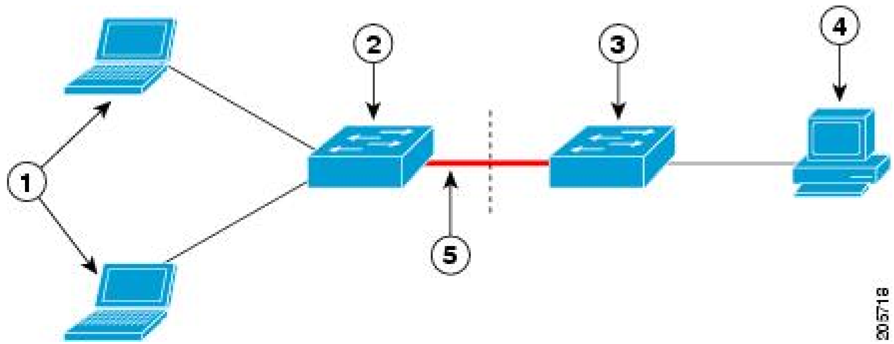
You can enable MDA or multiauth mode on the authenticator switch interface that connects to one more supplicant switches. Multihost mode is not supported on the authenticator switch interface.

Use the **dot1x supplicant force-multicast** global configuration command on the supplicant switch for Network Edge Access Topology (NEAT) to work in all host modes.

- Host Authorization: Ensures that only traffic from authorized hosts (connecting to the switch with supplicant) is allowed on the network. The switches use Client Information Signalling Protocol (CISP) to send the MAC addresses connecting to the supplicant switch to the authenticator switch.

- Auto enablement: Automatically enables trunk configuration on the authenticator switch, allowing user traffic from multiple VLANs coming from supplicant switches. Configure the cisco-av-pair as *device-traffic-class=switch* at the ACS. (You can configure this under the *group* or the *user* settings.)

Figure 54: Authenticator and Supplicant Switch using CISP



1	Workstations (clients)	2	Supplicant switch (outside wiring closet)
3	Authenticator switch	4	Access control server (ACS)
5	Trunk port		

Voice Aware 802.1x Security

You use the voice aware 802.1x security feature to configure the switch to disable only the VLAN on which a security violation occurs, whether it is a data or voice VLAN. In previous releases, when an attempt to authenticate the data client caused a security violation, the entire port shut down, resulting in a complete loss of connectivity.

You can use this feature in IP phone deployments where a PC is connected to the IP phone. A security violation found on the data VLAN results in the shutdown of only the data VLAN. The traffic on the voice VLAN flows through the switch without interruption.

Related Topics

Configuring Voice Aware 802.1x Security, on page 1168

Common Session ID

Authentication manager uses a single session ID (referred to as a common session ID) for a client no matter which authentication method is used. This ID is used for all reporting purposes, such as the show commands and MIBs. The session ID appears with all per-session syslog messages.

The session ID includes:

- The IP address of the Network Access Device (NAD)
- A monotonically increasing unique 32 bit integer

- The session start time stamp (a 32 bit integer)

This example shows how the session ID appears in the output of the show authentication command. The session ID in this example is 160000050000000B288508E5:

```
Switch# show authentication sessions
Interface  MAC Address     Method   Domain   Status         Session ID
Fa4/0/4    0000.0000.0203  mab      DATA     Authz Success  160000050000000B288508E5
```

This is an example of how the session ID appears in the syslog output. The session ID in this example is also160000050000000B288508E5:

```
1w0d: %AUTHMGR-5-START: Starting 'mab' for client (0000.0000.0203) on Interface Fa4/0/4
AuditSessionID 160000050000000B288508E5
1w0d: %MAB-5-SUCCESS: Authentication successful for client (0000.0000.0203) on Interface
Fa4/0/4 AuditSessionID 160000050000000B288508E5
1w0d: %AUTHMGR-7-RESULT: Authentication result 'success' from 'mab' for client
(0000.0000.0203) on Interface Fa4/0/4 AuditSessionID 160000050000000B288508E5
```

The session ID is used by the NAD, the AAA server, and other report-analyzing applications to identify the client. The ID appears automatically. No configuration is required.

How to Configure 802.1x Port-Based Authentication

Default 802.1x Authentication Configuration

Table 135: Default 802.1x Authentication Configuration

Feature	Default Setting
Switch 802.1x enable state	Disabled.
Per-port 802.1x enable state	Disabled (force-authorized). The port sends and receives normal traffic without 802.1x-based authentication of the client.
AAA	Disabled.
RADIUS server • IP address • UDP authentication port • Key	• None specified. • 1812. • None specified.
Host mode	Single-host mode.
Control direction	Bidirectional control.
Periodic re-authentication	Disabled.

Feature	Default Setting
Number of seconds between re-authentication attempts	3600 seconds.
Re-authentication number	2 times (number of times that the switch restarts the authentication process before the port changes to the unauthorized state).
Quiet period	60 seconds (number of seconds that the switch remains in the quiet state following a failed authentication exchange with the client).
Retransmission time	30 seconds (number of seconds that the switch should wait for a response to an EAP request/identity frame from the client before resending the request).
Maximum retransmission number	2 times (number of times that the switch will send an EAP-request/identity frame before restarting the authentication process).
Client timeout period	30 seconds (when relaying a request from the authentication server to the client, the amount of time the switch waits for a response before resending the request to the client.)
Authentication server timeout period	30 seconds (when relaying a response from the client to the authentication server, the amount of time the switch waits for a reply before resending the response to the server.) You can change this timeout period by using the dot1x timeout server-timeout interface configuration command.
Guest VLAN	None specified.
Inaccessible authentication bypass	Disabled.
Restricted VLAN	None specified.
Authenticator (switch) mode	None specified.
MAC authentication bypass	Disabled.

802.1x Authentication Configuration Guidelines

802.1x Authentication

These are the 802.1x authentication configuration guidelines:

- When 802.1x authentication is enabled, ports are authenticated before any other Layer 2 or Layer 3 features are enabled.

- If the VLAN to which an 802.1x-enabled port is assigned changes, this change is transparent and does not affect the switch. For example, this change occurs if a port is assigned to a RADIUS server-assigned VLAN and is then assigned to a different VLAN after re-authentication.

  If the VLAN to which an 802.1x port is assigned to shut down, disabled, or removed, the port becomes unauthorized. For example, the port is unauthorized after the access VLAN to which a port is assigned shuts down or is removed.

- The 802.1x protocol is supported on Layer 2 static-access ports, voice VLAN ports, and Layer 3 routed ports, but it is not supported on these port types:
  - Trunk port—If you try to enable 802.1x authentication on a trunk port, an error message appears, and 802.1x authentication is not enabled. If you try to change the mode of an 802.1x-enabled port to trunk, an error message appears, and the port mode is not changed.
  - Dynamic ports—A port in dynamic mode can negotiate with its neighbor to become a trunk port. If you try to enable 802.1x authentication on a dynamic port, an error message appears, and 802.1x authentication is not enabled. If you try to change the mode of an 802.1x-enabled port to dynamic, an error message appears, and the port mode is not changed.
  - Dynamic-access ports—If you try to enable 802.1x authentication on a dynamic-access (VLAN Query Protocol [VQP]) port, an error message appears, and 802.1x authentication is not enabled. If you try to change an 802.1x-enabled port to dynamic VLAN assignment, an error message appears, and the VLAN configuration is not changed.
  - EtherChannel port—Do not configure a port that is an active or a not-yet-active member of an EtherChannel as an 802.1x port. If you try to enable 802.1x authentication on an EtherChannel port, an error message appears, and 802.1x authentication is not enabled.
  - Switched Port Analyzer (SPAN) and Remote SPAN (RSPAN) destination ports—You can enable 802.1x authentication on a port that is a SPAN or RSPAN destination port. However, 802.1x authentication is disabled until the port is removed as a SPAN or RSPAN destination port. You can enable 802.1x authentication on a SPAN or RSPAN source port.

- Before globally enabling 802.1x authentication on a switch by entering the **dot1x system-auth-control** global configuration command, remove the EtherChannel configuration from the interfaces on which 802.1x authentication and EtherChannel are configured.

- If you are using a device running the Cisco Access Control Server (ACS) application for IEEE 802.1x authentication with EAP-Transparent LAN Services (TLS) and EAP-MD5, make sure that the device is running ACS Version 3.2.1 or later.

- When IP phones are connected to an 802.1x-enabled switch port that is in single host mode, the switch grants the phones network access without authenticating them. We recommend that you use multidomain authentication (MDA) on the port to authenticate both a data device and a voice device, such as an IP phone.

Note Only Catalyst 3750, 3560, and 2960 switches support CDP bypass. The Catalyst 3750-X, 3560-X, 3750-E, and 3560-E switches do not support CDP bypass.

- Cisco IOS Release 12.2(55)SE and later supports filtering of system messages related to 802.1x authentication.

VLAN Assignment, Guest VLAN, Restricted VLAN, and Inaccessible Authentication Bypass

These are the configuration guidelines for VLAN assignment, guest VLAN, restricted VLAN, and inaccessible authentication bypass:

- When 802.1x authentication is enabled on a port, you cannot configure a port VLAN that is equal to a voice VLAN.
- The 802.1x authentication with VLAN assignment feature is not supported on trunk ports, dynamic ports, or with dynamic-access port assignment through a VMPS.
- You can configure 802.1x authentication on a private-VLAN port, but do not configure IEEE 802.1x authentication with port security, a voice VLAN, a guest VLAN, a restricted VLAN, or a per-user ACL on private-VLAN ports.
- You can configure any VLAN except an RSPAN VLAN or a voice VLAN as an 802.1x guest VLAN. The guest VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.
- After you configure a guest VLAN for an 802.1x port to which a DHCP client is connected, you might need to get a host IP address from a DHCP server. You can change the settings for restarting the 802.1x authentication process on the switch before the DHCP process on the client times out and tries to get a host IP address from the DHCP server. Decrease the settings for the 802.1x authentication process (**authentication timer inactivity** and **authentication timer reauthentication** interface configuration commands). The amount to decrease the settings depends on the connected 802.1x client type.
- When configuring the inaccessible authentication bypass feature, follow these guidelines:
  - The feature is supported on 802.1x port in single-host mode and multihosts mode.
  - If the client is running Windows XP and the port to which the client is connected is in the critical-authentication state, Windows XP might report that the interface is not authenticated.
  - If the Windows XP client is configured for DHCP and has an IP address from the DHCP server, receiving an EAP-Success message on a critical port might not re-initiate the DHCP configuration process.
  - You can configure the inaccessible authentication bypass feature and the restricted VLAN on an 802.1x port. If the switch tries to re-authenticate a critical port in a restricted VLAN and all the RADIUS servers are unavailable, switch changes the port state to the critical authentication state and remains in the restricted VLAN.
- You can configure any VLAN except an RSPAN VLAN or a voice VLAN as an 802.1x restricted VLAN. The restricted VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.

MAC Authentication Bypass

These are the MAC authentication bypass configuration guidelines:

- Unless otherwise stated, the MAC authentication bypass guidelines are the same as the 802.1x authentication guidelines.
- If you disable MAC authentication bypass from a port after the port has been authorized with its MAC address, the port state is not affected.

- If the port is in the unauthorized state and the client MAC address is not the authentication-server database, the port remains in the unauthorized state. However, if the client MAC address is added to the database, the switch can use MAC authentication bypass to re-authorize the port.
- If the port is in the authorized state, the port remains in this state until re-authorization occurs.

Maximum Number of Allowed Devices Per Port

This is the maximum number of devices allowed on an 802.1x-enabled port:

- In single-host mode, only one device is allowed on the access VLAN. If the port is also configured with a voice VLAN, an unlimited number of Cisco IP phones can send and receive traffic through the voice VLAN.
- In multidomain authentication (MDA) mode, one device is allowed for the access VLAN, and one IP phone is allowed for the voice VLAN.
- In multihost mode, only one 802.1x supplicant is allowed on the port, but an unlimited number of non-802.1x hosts are allowed on the access VLAN. An unlimited number of devices are allowed on the voice VLAN.

Configuring 802.1x Readiness Check

Beginning in privileged EXEC mode, follow these steps to enable the 802.1x readiness check on the switch:

SUMMARY STEPS

1. **dot1x test eapol-capable** [**interface** *interface-id*]
2. **configure terminal**
3. **dot1x test timeout** *timeout*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**dot1x test eapol-capable** [**interface** *interface-id*] **Example:** `Switch# dot1x test eapol-capable interface gigabitethernet1/0/13`	Enables the 802.1x readiness check on the switch. (Optional) For *interface-id* specify the port on which to check for IEEE 802.1x readiness. **Note** If you omit the optional **interface** keyword, all interfaces on the switch are tested.
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	(Optional) Enters global configuration mode.

	Command or Action	Purpose
Step 3	**dot1x test timeout** *timeout* **Example:** `Switch(config)# dot1x test timeout 300`	(Optional) Configures the timeout used to wait for EAPOL response. The range is from 1 to 65535 seconds. The default is 10 seconds.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Related Topics

802.1x Readiness Check, on page 1145

Configuring Voice Aware 802.1x Security

Follow these guidelines to configure voice aware 802.1x voice security on the switch:

- You enable voice aware 802.1x security by entering the **errdisable detect cause security-violation shutdown vlan** global configuration command. You disable voice aware 802.1x security by entering the **no** version of this command. This command applies to all 802.1x-configured ports in the switch.

> **Note** If you do not include the **shutdown vlan** keywords, the entire port is shut down when it enters the error-disabled state.

- If you use the **errdisable recovery cause security-violation** global configuration command to configure error-disabled recovery, the port is automatically re-enabled. If error-disabled recovery is not configured for the port, you re-enable it by using the **shutdown** and **no shutdown** interface configuration commands.
- You can re-enable individual VLANs by using the **clear errdisable interface** *interface-id* **vlan** [*vlan-list*] privileged EXEC command. If you do not specify a range, all VLANs on the port are enabled.

Beginning in privileged EXEC mode, follow these steps to enable voice aware 802.1x security:

SUMMARY STEPS

1. **configure terminal**
2. **errdisable detect cause security-violation shutdown vlan**
3. **errdisable recovery cause security-violation**
4. **clear errdisable interface** *interface-id* **vlan** [*vlan-list*]
5. Enter the following:
   - **shutdown**
   - **no shutdown**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**errdisable detect cause security-violation shutdown vlan** **Example:** `Switch(config)# errdisable detect cause security-violation shutdown vlan`	Shuts down any VLAN on which a security violation error occurs. **Note** If the **shutdown vlan** keywords are not included, the entire port enters the error-disabled state and shuts down.
Step 3	**errdisable recovery cause security-violation** **Example:** `Switch(config)# errdisable recovery cause security-violation`	(Optional) Enables automatic per-VLAN error recovery.
Step 4	**clear errdisable interface** *interface-id* **vlan** [*vlan-list*] **Example:** `Switch(config)# clear errdisable interface GigabitEthernet4/0/2 vlan`	(Optional) Reenables individual VLANs that have been error disabled. • For *interface-id*, specify the port on which to reenable individual VLANs. • (Optional) For *vlan-list*, specify a list of VLANs to be re-enabled. If *vlan-list* is not specified, all VLANs are re-enabled.

	Command or Action	Purpose

Step 5	Enter the following:
	• **shutdown**
	• **no shutdown**
	
	Example:
	`Switch(config-if)# shutdown`
	`Switch(config-if)# no shutdown`
		(Optional) Re-enables an error-disabled VLAN, and clear all error-disable indications.

Step 6	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Related Topics

Voice Aware 802.1x Security, on page 1162

Configuring 802.1x Violation Modes

You can configure an 802.1x port so that it shuts down, generates a syslog error, or discards packets from a new device when:

- a device connects to an 802.1x-enabled port
- the maximum number of allowed about devices have been authenticated on the port

Beginning in privileged EXEC mode, follow these steps to configure the security violation actions on the switch:

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **aaa authentication dot1x** {**default**} *method1*
4. **interface** *interface-id*
5. **switchport mode access**
6. **authentication violation** {**shutdown** | **restrict** | **protect** | **replace**}
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**aaa new-model** **Example:** `Switch(config)# aaa new-model`	Enables AAA.
Step 3	**aaa authentication dot1x** {**default**} *method1* **Example:** `Switch(config)# aaa authentication dot1x default group radius`	Creates an 802.1x authentication method list. To create a default list that is used when a named list is *not* specified in the **authentication** command, use the **default** keyword followed by the method that is to be used in default situations. The default method list is automatically applied to all ports. For *method1*, enter the **group radius** keywords to use the list of all RADIUS servers for authentication. **Note** Though other keywords are visible in the command-line help string, only the **group radius** keywords are supported.
Step 4	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/4`	Specifies the port connected to the client that is to be enabled for IEEE 802.1x authentication, and enter interface configuration mode.
Step 5	**switchport mode access** **Example:** `Switch(config-if)# switchport mode access`	Sets the port to access mode.
Step 6	**authentication violation** {**shutdown** | **restrict** | **protect** | **replace**} **Example:** `Switch(config-if)# authentication violation restrict`	Configures the violation mode. The keywords have these meanings: • **shutdown**–Error disable the port. • **restrict**–Generate a syslog error. • **protect**–Drop packets from any new device that sends traffic to the port. • **replace**–Removes the current session and authenticates with the new host.

	Command or Action	Purpose
Step 7	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Configuring 802.1x Authentication

To allow per-user ACLs or VLAN assignment, you must enable AAA authorization to configure the switch for all network-related service requests.

This is the 802.1x AAA process:

Before You Begin

To configure 802.1x port-based authentication, you must enable authentication, authorization, and accounting (AAA) and specify the authentication method list. A method list describes the sequence and authentication method to be queried to authenticate a user.

SUMMARY STEPS

1. A user connects to a port on the switch.
2. Authentication is performed.
3. VLAN assignment is enabled, as appropriate, based on the RADIUS server configuration.
4. The switch sends a start message to an accounting server.
5. Re-authentication is performed, as necessary.
6. The switch sends an interim accounting update to the accounting server that is based on the result of re-authentication.
7. The user disconnects from the port.
8. The switch sends a stop message to the accounting server.

DETAILED STEPS

	Command or Action	Purpose
Step 1	A user connects to a port on the switch.	
Step 2	Authentication is performed.	
Step 3	VLAN assignment is enabled, as appropriate, based on the RADIUS server configuration.	
Step 4	The switch sends a start message to an accounting server.	
Step 5	Re-authentication is performed, as necessary.	
Step 6	The switch sends an interim accounting update to the accounting server that is based on the result of re-authentication.	

	Command or Action	Purpose
Step 7	The user disconnects from the port.	
Step 8	The switch sends a stop message to the accounting server.	

Configuring 802.1x Port-Based Authentication

Beginning in privileged EXEC mode, follow these steps to configure 802.1x port-based authentication:

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **aaa authentication dot1x** {**default**} *method1*
4. **dot1x system-auth-control**
5. **aaa authorization network** {**default**} **group radius**
6. **radius-server host** *ip-address*
7. **radius-server key** *string*
8. **interface** *interface-i*d
9. **switchport mode access**
10. **authentication port-control auto**
11. **dot1x pae authenticator**
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**aaa new-model** **Example:** `Switch(config)# aaa new-model`	Enables AAA.
Step 3	**aaa authentication dot1x** {**default**} *method1*	Creates an 802.1x authentication method list.

	Command or Action	Purpose
	Example: `Switch(config)# aaa authentication dot1x default group radius`	To create a default list that is used when a named list is *not* specified in the **authentication** command, use the **default** keyword followed by the method that is to be used in default situations. The default method list is automatically applied to all ports. For *method1*, enter the **group radius** keywords to use the list of all RADIUS servers for authentication. **Note** Though other keywords are visible in the command-line help string, only the **group radius** keywords are supported.
Step 4	**dot1x system-auth-control** **Example:** `Switch(config)# dot1x system-auth-control`	Enables 802.1x authentication globally on the switch.
Step 5	**aaa authorization network** {**default**} **group radius** **Example:** `Switch(config)# aaa authorization network default group radius`	(Optional) Configures the switch to use user-RADIUS authorization for all network-related service requests, such as per-user ACLs or VLAN assignment. **Note** For per-user ACLs, single-host mode must be configured. This setting is the default.
Step 6	**radius-server host** *ip-address* **Example:** `Switch(config)# radius-server host 124.2.2.12`	(Optional) Specifies the IP address of the RADIUS server.
Step 7	**radius-server key** *string* **Example:** `Switch(config)# radius-server key abc1234`	(Optional) Specifies the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server.
Step 8	**interface** *interface-i*d **Example:** `Switch(config)# interface gigabitethernet1/0/2`	Specifies the port connected to the client that is to be enabled for IEEE 802.1x authentication, and enter interface configuration mode.
Step 9	**switchport mode access** **Example:** `Switch(config-if)# switchport mode access`	(Optional) Sets the port to access mode only if you configured the RADIUS server in Step 6 and Step 7.

	Command or Action	Purpose
Step 10	**authentication port-control auto** **Example:** `Switch(config-if)# authentication port-control auto`	Enables 802.1x authentication on the port.
Step 11	**dot1x pae authenticator** **Example:** `Switch(config-if)# dot1x pae authenticator`	Sets the interface Port Access Entity to act only as an authenticator and ignore messages meant for a supplicant.
Step 12	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Configuring the Switch-to-RADIUS-Server Communication

You can globally configure the timeout, retransmission, and encryption key values for all RADIUS servers by using the **radius-server host** global configuration command. If you want to configure these options on a per-server basis, use the **radius-server timeout**, the **radius-server retransmit**, and the **radius-server key** global configuration commands.

You also need to configure some settings on the RADIUS server. These settings include the IP address of the switch and the key string to be shared by both the server and the switch. For more information, see the RADIUS server documentation.

Beginning in privileged EXEC mode, follow these steps to configure the RADIUS server parameters on the switch. This procedure is required.

Before You Begin

You must enable authentication, authorization, and accounting (AAA) and specify the authentication method list. A method list describes the sequence and authentication method to be queried to authenticate a user.

SUMMARY STEPS

1. **configure terminal**
2. **radius-server host** {*hostname* | *ip-address*} **auth-port** *port-number* **key** *string*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**radius-server host** {*hostname* | *ip-address*} **auth-port** *port-number* **key** *string* **Example:** `Switch(config)# radius-server host 125.5.5.43 auth-port 1812 key string`	Configures the RADIUS server parameters. For *hostname* | *ip-address,* specify the hostname or IP address of the remote RADIUS server. For **auth-port** *port-number*, specify the UDP destination port for authentication requests. The default is 1812. The range is 0 to 65536. For **key** *string*, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. The key is a text string that must match the encryption key used on the RADIUS server. **Note** Always configure the key as the last item in the **radius-server host** command syntax because leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in the key, do not enclose the key in quotation marks unless the quotation marks are part of the key. This key must match the encryption used on the RADIUS daemon. If you want to use multiple RADIUS servers, re-enter this command.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Related Topics

Switch-to-RADIUS-Server Communication, on page 1146

Configuring the Host Mode

Beginning in privileged EXEC mode, follow these steps to allow multiple hosts (clients) on an IEEE 802.1x-authorized port that has the **authentication port-control** interface configuration command set to **auto**. Use the **multi-domain** keyword to configure and enable multidomain authentication (MDA), which allows both a host and a voice device, such as an IP phone (Cisco or non-Cisco), on the same switch port. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication host-mode** [**multi-auth** | **multi-domain** | **multi-host** | **single-host**]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/1`	Specifies the port to which multiple hosts are indirectly attached, and enter interface configuration mode.
Step 3	**authentication host-mode** [**multi-auth** | **multi-domain** | **multi-host** | **single-host**] **Example:** `Switch(config-if)# authentication host-mode multi-host`	Allows multiple hosts (clients) on an 802.1x-authorized port. The keywords have these meanings: • **multi-auth**–Allow one client on the voice VLAN and multiple authenticated clients on the data VLAN. **Note** The **multi-auth** keyword is only available with the **authentication host-mode** command. • **multi-host**–Allow multiple hosts on an 802.1x-authorized port after a single host has been authenticated. • **multi-domain**–Allow both a host and a voice device, such as an IP phone (Cisco or non-Cisco), to be authenticated on an IEEE 802.1x-authorized port. **Note** You must configure the voice VLAN for the IP phone when the host mode is set to **multi-domain**. Make sure that the **authentication port-control** interface configuration command is set to **auto** for the specified interface.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Configuring Periodic Re-Authentication

You can enable periodic 802.1x client re-authentication and specify how often it occurs. If you do not specify a time period before enabling re-authentication, the number of seconds between attempts is 3600.

Beginning in privileged EXEC mode, follow these steps to enable periodic re-authentication of the client and to configure the number of seconds between re-authentication attempts. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication periodic**
4. **authentication timer** {{[**inactivity** | **reauthenticate** | **restart**]} {*value*}}
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/1`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**authentication periodic** **Example:** `Switch(config-if)# authentication periodic`	Enables periodic re-authentication of the client, which is disabled by default. **Note** The default value is 3600 seconds. To change the value of the reauthentication timer or to have the switch use a RADIUS-provided session timeout, enter the **authentication timer reauthenticate** command.
Step 4	**authentication timer** {{[**inactivity** | **reauthenticate** | **restart**]} {*value*}} **Example:** `Switch(config-if)# authentication`	Sets the number of seconds between re-authentication attempts. The **authentication timer** keywords have these meanings: • **inactivity—**Interval in seconds after which if there is no activity from the client then it is unauthorized

	Command or Action	Purpose
	`timer reauthenticate 180`	• **reauthenticate—**Time in seconds after which an automatic re-authentication attempt is initiated • **restart** *value***—**Interval in seconds after which an attempt is made to authenticate an unauthorized port This command affects the behavior of the switch only if periodic re-authentication is enabled.
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Changing the Quiet Period

When the switch cannot authenticate the client, the switch remains idle for a set period of time and then tries again. The **authentication timer inactivity** interface configuration command controls the idle period. A failed authentication of the client might occur because the client provided an invalid password. You can provide a faster response time to the user by entering a number smaller than the default.

Beginning in privileged EXEC mode, follow these steps to change the quiet period. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication timer inactivity** *seconds*
4. **end**
5. **show authentication sessions interface** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/1`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**authentication timer inactivity** *seconds* **Example:** `Switch(config-if)# authentication timer inactivity 30`	Sets the number of seconds that the switch remains in the quiet state following a failed authentication exchange with the client. The range is 1 to 65535 seconds; the default is 60.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show authentication sessions interface** *interface-id* **Example:** `Switch# show authentication sessions interface gigabitethernet2/0/1`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Changing the Switch-to-Client Retransmission Time

The client responds to the EAP-request/identity frame from the switch with an EAP-response/identity frame. If the switch does not receive this response, it waits a set period of time (known as the retransmission time) and then resends the frame.

Note You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

Beginning in privileged EXEC mode, follow these steps to change the amount of time that the switch waits for client notification. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication timer reauthenticate** *seconds*
4. **end**
5. **show authentication sessions interface** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/1`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**authentication timer reauthenticate** *seconds* **Example:** `Switch(config-if)# authentication timer reauthenticate 60`	Sets the number of seconds that the switch waits for a response to an EAP-request/identity frame from the client before resending the request. The range is 1 to 65535 seconds; the default is 5.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show authentication sessions interface** *interface-id* **Example:** `Switch# show authentication sessions interface gigabitethernet2/0/1`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Setting the Switch-to-Client Frame-Retransmission Number

In addition to changing the switch-to-client retransmission time, you can change the number of times that the switch sends an EAP-request/identity frame (assuming no response is received) to the client before restarting the authentication process.

Note You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

Beginning in privileged EXEC mode, follow these steps to set the switch-to-client frame-retransmission number. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **dot1x max-reauth-req** *count*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/1`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**dot1x max-reauth-req** *count* **Example:** `Switch(config-if)# dot1x max-reauth-req 5`	Sets the number of times that the switch sends an EAP-request/identity frame to the client before restarting the authentication process. The range is 1 to 10; the default is 2.

	Command or Action	Purpose
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Setting the Re-Authentication Number

You can also change the number of times that the switch restarts the authentication process before the port changes to the unauthorized state.

Note You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

Beginning in privileged EXEC mode, follow these steps to set the re-authentication number. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **dot1x max-req** *count*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch# interface gigabitethernet2/0/1`	Specifies the port to be configured, and enter interface configuration mode.

	Command or Action	Purpose
Step 3	**switchport mode access** **Example:** `Switch(config-if)# switchport mode access`	Sets the port to access mode only if you previously configured the RADIUS server.
Step 4	**dot1x max-req** *count* **Example:** `Switch(config-if)# dot1x max-req 4`	Sets the number of times that the switch restarts the authentication process before the port changes to the unauthorized state. The range is 0 to 10; the default is 2.
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Enabling MAC Move

MAC move allows an authenticated host to move from one port on the switch to another.

Beginning in privileged EXEC mode, follow these steps to globally enable MAC move on the switch. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **authentication mac-move permit**
3. **end**
4. **show running-config**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**authentication mac-move permit**	Enables MAC move on the switch. Default is deny.

	Command or Action	Purpose
	Example: `Switch(config)# authentication mac-move permit`	In Session Aware Networking mode, the default CLI is **access-session mac-move deny**. To enable Mac Move in Session Aware Networking, use the **no access-session mac-move** global configuration command.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Enabling MAC Replace

MAC replace allows a host to replace an authenticated host on a port.

Beginning in privileged EXEC mode, follow these steps to enable MAC replace on an interface. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication violation** {**protect** | **replace** | **restrict** | **shutdown**}
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/2`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**authentication violation** {**protect** | **replace** | **restrict** | **shutdown**} **Example:** `Switch(config-if)# authentication violation replace`	Use the **replace** keyword to enable MAC replace on the interface. The port removes the current session and initiates authentication with the new host. The other keywords have these effects: • **protect**: the port drops packets with unexpected MAC addresses without generating a system message. • **restrict**: violating packets are dropped by the CPU and a system message is generated. • **shutdown**: the port is error disabled when it receives an unexpected MAC address.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring 802.1x Accounting

Enabling AAA system accounting with 802.1x accounting allows system reload events to be sent to the accounting RADIUS server for logging. The server can then infer that all active 802.1x sessions are closed.

Because RADIUS uses the unreliable UDP transport protocol, accounting messages might be lost due to poor network conditions. If the switch does not receive the accounting response message from the RADIUS server after a configurable number of retransmissions of an accounting request, this system message appears:

```
Accounting message %s for session %s failed to receive Accounting Response.
```

When the stop message is not sent successfully, this message appears:

```
00:09:55: %RADIUS-4-RADIUS_DEAD: RADIUS server 172.20.246.201:1645,1646 is not responding.
```

Note You must configure the RADIUS server to perform accounting tasks, such as logging start, stop, and interim-update messages and time stamps. To turn on these functions, enable logging of "Update/Watchdog packets from this AAA client" in your RADIUS server Network Configuration tab. Next, enable "CVS RADIUS Accounting" in your RADIUS server System Configuration tab.

Beginning in privileged EXEC mode, follow these steps to configure 802.1x accounting after AAA is enabled on your switch. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **aaa accounting dot1x default start-stop group radius**
4. **aaa accounting system default start-stop group radius**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/3`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**aaa accounting dot1x default start-stop group radius** **Example:** `Switch(config-if)# aaa accounting dot1x default start-stop group radius`	Enables 802.1x accounting using the list of all RADIUS servers.
Step 4	**aaa accounting system default start-stop group radius** **Example:** `Switch(config-if)# aaa accounting system default start-stop group radius`	(Optional) Enables system accounting (using the list of all RADIUS servers) and generates system accounting reload event messages when the switch reloads.
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEc mode.
Step 6	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring a Guest VLAN

When you configure a guest VLAN, clients that are not 802.1x-capable are put into the guest VLAN when the server does not receive a response to its EAP request/identity frame. Clients that are 802.1x-capable but that fail authentication are not granted network access. The switch supports guest VLANs in single-host or multiple-hosts mode.

Beginning in privileged EXEC mode, follow these steps to configure a guest VLAN. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. Use one of the following:
   - **switchport mode access**
   - **switchport mode private-vlan host**
4. **authentication event no-response action authorize vlan** *vlan-id*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/2`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	Use one of the following: • **switchport mode access** • **switchport mode private-vlan host** **Example:** `Switch(config-if)# switchport mode private-vlan host`	• Sets the port to access mode. • Configures the Layer 2 port as a private-VLAN host port.
Step 4	**authentication event no-response action authorize vlan** *vlan-id* **Example:** `Switch(config-if)# authentication event no-response action authorize vlan 2`	Specifies an active VLAN as an 802.1x guest VLAN. The range is 1 to 4094. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN or a voice VLAN as an 802.1x guest VLAN.

	Command or Action	Purpose
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Configuring a Restricted VLAN

When you configure a restricted VLAN on a switch stack or a switch, clients that are IEEE 802.1x-compliant are moved into the restricted VLAN when the authentication server does not receive a valid username and password. The switch supports restricted VLANs only in single-host mode.

Beginning in privileged EXEC mode, follow these steps to configure a restricted VLAN. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. Use one of the following:
    - **switchport mode access**
    - **switchport mode private-vlan host**
4. **authentication port-control auto**
5. **authentication event fail action authorize vlan** *vlan-id*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/2`	Specifies the port to be configured, and enter interface configuration mode.

	Command or Action	Purpose
Step 3	Use one of the following: • **switchport mode access** • **switchport mode private-vlan host** **Example:** `Switch(config-if)# switchport mode access`	• Sets the port to access mode. • Configures the Layer 2 port as a private-VLAN host port.
Step 4	**authentication port-control auto** **Example:** `Switch(config-if)# authentication port-control auto`	Enables 802.1x authentication on the port.
Step 5	**authentication event fail action authorize vlan** *vlan-id* **Example:** `Switch(config-if)# authentication event fail action authorize vlan 2`	Specifies an active VLAN as an 802.1x restricted VLAN. The range is 1 to 4094. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN or a voice VLAN as an 802.1x restricted VLAN.
Step 6	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Configuring Number of Authentication Attempts on a Restricted VLAN

You can configure the maximum number of authentication attempts allowed before a user is assigned to the restricted VLAN by using the **authentication event retry** *retry count* interface configuration command. The range of allowable authentication attempts is 1 to 3. The default is 3 attempts.

Beginning in privileged EXEC mode, follow these steps to configure the maximum number of allowed authentication attempts. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. Use one of the following:
   - **switchport mode access**
   - **switchport mode private-vlan host**
4. **authentication port-control auto**
5. **authentication event fail action authorize vlan** *vlan-id*
6. **authentication event retry** *retry count*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/3`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	Use one of the following: • **switchport mode access** • **switchport mode private-vlan host** **Example:** or `Switch(config-if)# switchport mode access`	• Sets the port to access mode. • Configures the Layer 2 port as a private-VLAN host port.
Step 4	**authentication port-control auto** **Example:** `Switch(config-if)# authentication port-control auto`	Enables 802.1x authentication on the port.

	Command or Action	Purpose
Step 5	**authentication event fail action authorize vlan** *vlan-id* **Example:** `Switch(config-if)#` **`authentication event fail action authorize vlan 8`**	Specifies an active VLAN as an 802.1x restricted VLAN. The range is 1 to 4094. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN or a voice VLAN as an 802.1x restricted VLAN.
Step 6	**authentication event retry** *retry count* **Example:** `Switch(config-if)#` **`authentication event retry 2`**	Specifies a number of authentication attempts to allow before a port moves to the restricted VLAN. The range is 1 to 3, and the default is 3.
Step 7	**end** **Example:** `Switch(config-if)#` **`end`**	Returns to privileged EXEC mode.

Configuring 802.1x Inaccessible Authentication Bypass with Critical Voice VLAN

Beginning in privileged EXEC mode, follow these steps to configure critical voice VLAN on a port and enable the inaccessible authentication bypass feature.

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **radius-server dead-criteria**{**time** *seconds* } [**tries** *number*]
4. **radius-serverdeadtime***minutes*
5. **radius-server host ip-address** *address*[**acct-port** *udp-port*][**auth-port** *udp-port*] [**testusername** *name*[**idle-time** *time*] [**ignore-acct-port**][**ignore auth-port**]] [**key** *string*]
6. **dot1x critical** {**eapol** | **recovery delay** *milliseconds*}
7. **interface** *interface-id*
8. **authentication event server dead action** {**authorize** | **reinitialize**} **vlan** *vlan-id*]
9. **switchport voice vlan** *vlan-id*
10. **authentication event server dead action authorize voice**
11. **show authentication interface** *interface-id*
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**aaa new-model** **Example:** `Switch(config)# aaa new-model`	Enables AAA.
Step 3	**radius-server dead-criteria**{**time** *seconds* } [**tries** *number*] **Example:** `Switch(config)# radius-server dead-criteria time 20 tries 10`	Sets the conditions that determine when a RADIUS server is considered un-available or down (dead). • **time—** 1 to 120 seconds. The switch dynamically determines a default *seconds* value between 10 and 60. • **number—**1 to 100 tries. The switch dynamically determines a default **tries***number* between 10 and 100.
Step 4	**radius-serverdeadtime***minutes* **Example:** `Switch(config)# radius-server deadtime 60`	(Optional) Sets the number of minutes during which a RADIUS server is not sent requests. The range is from 0 to 1440 minutes (24 hours). The default is 0 minutes.
Step 5	**radius-server host ip-address** *address*[**acct-port** *udp-port*][**auth-port** *udp-port*] [**testusername** *name*[**idle-time** *time*] [**ignore-acct-port**][**ignore auth-port**]] [**key** *string*] **Example:** `Switch(config)# radius-server host 1.1.1.2 acct-port 1550 auth-port 1560 test username user1 idle-time 30 key abc1234`	(Optional) Configure the RADIUS server parameters by using these keywords: • **acct-port***udp-port***—**Specify the UDP port for the RADIUS accounting server. The range for the UDP port number is from 0 to 65536. The default is 1646. • **auth-port***udp-port***—**Specify the UDP port for the RADIUS authentication server. The range for the UDP port number is from 0 to 65536. The default is 1645. **Note** You should configure the UDP port for the RADIUS accounting server and the UDP port for the RADIUS authentication server to nondefault values. • **test username***name***—**Enable automated testing of the RADIUS server status, and specify the username to be used. • **idle-time** *time***—**Set the interval of time in minutes after which the switch sends test packets to the server. The range is from 1 to 35791 minutes. The default is 60 minutes (1 hour).

	Command or Action	Purpose
		• **ignore-acct-port—**Disable testing on the RADIUS-server accounting port. • **ignore-auth-port—**Disable testing on the RADIUS-server authentication port. • For **key***string*, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. The key is a text string that must match the encryption key used on the RADIUS server. **Note** Always configure the key as the last item in the **radius-server host** command syntax because leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in the key, do not enclose the key in quotation marks unless the quotation marks are part of the key. This key must match the encryption used on the RADIUS daemon. You can also configure the authentication and encryption key by using the**radius-server key** {**0**string | **7***string* | *string*} global configuration command.

Step 6	**dot1x critical** {**eapol** | **recovery delay** *milliseconds*}
	
	Example:
	`Switch(config)# dot1x critical eapol`
	`(config)# dot1x critical recovery delay 2000`
		(Optional) Configure the parameters for inaccessible authentication bypass:
		• **eapol—**Specify that the switch sends an EAPOL-Success message when the switch successfully authenticates the critical port.
		• **recovery delay***milliseconds***—**Set the recovery delay period during which the switch waits to re-initialize a critical port when a RADIUS server that was unavailable becomes available. The range is from 1 to 10000 milliseconds. The default is 1000 milliseconds (a port can be re-initialized every second).

Step 7	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Specify the port to be configured, and enter interface configuration mode.
Step 8	**authentication event server dead action** {**authorize** | **reinitialize**} **vlan** *vlan-id*] **Example:** `Switch(config-if)# authentication event server dead action reinitialicze vlan 20`	Use these keywords to move hosts on the port if the RADIUS server is unreachable: • **authorize—**Move any new hosts trying to authenticate to the user-specified critical VLAN. • **reinitialize—**Move all authorized hosts on the port to the user-specified critical VLAN.

	Command or Action	Purpose
Step 9	**switchport voice vlan** *vlan-id* **Example:** `Switch(config-if)#` **`switchport voice vlan`**	Specifies the voice VLAN for the port. The voice VLAN cannot be the same as the critical data VLAN configured in Step 6.
Step 10	**authentication event server dead action authorize voice** **Example:** `Switch(config-if)#` **`authentication event server dead action authorize voice`**	Configures critical voice VLAN to move data traffic on the port to the voice VLAN if the RADIUS server is unreachable.
Step 11	**show authentication interface** *interface-id* **Example:** `Switch(config-if)#` **`do show authentication interface gigabit 1/0/1`**	(Optional) Verify your entries.
Step 12	**copy running-config startup-config** **Example:** `Switch(config-if)#` **`do copy running-config startup-config`**	(Optional) Verify your entries.

To return to the RADIUS server default settings, use the **no radius-server dead-criteria**, the **no radius-server deadtime**, and the **no radius-server host** global configuration commands. To disable inaccessible authentication bypass, use the **no authentication event server dead action** interface configuration command. To disable critical voice VLAN, use the no **authentication event server dead action authorize voice** interface configuration command.

Example of Configuring Inaccessible Authentication Bypass

This example shows how to configure the inaccessible authentication bypass feature:

```
Switch(config)# radius-server dead-criteria time 30 tries 20
Switch(config)# radius-server deadtime 60
Switch(config)# radius-server host 1.1.1.2 acct-port 1550 auth-port 1560 test username user1
 idle-time 30 key abc1234
Switch(config)# dot1x critical eapol
Switch(config)# dot1x critical recovery delay 2000
Switch(config)# interface gigabitethernet 1/0/1
Switch(config-if)# dot1x critical
Switch(config-if)# dot1x critical recovery action reinitialize
```

```
Switch(config-if)# dot1x critical vlan 20
Switch(config-if)# end
```

Configuring 802.1x Authentication with WoL

Beginning in privileged EXEC mode, follow these steps to enable 802.1x authentication with WoL. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication control-direction** {**both** | **in**}
4. **end**
5. **show authentication sessions interface** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/3`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**authentication control-direction** {**both** | **in**} **Example:** `Switch(config-if)# authentication control-direction both`	Enables 802.1x authentication with WoL on the port, and use these keywords to configure the port as bidirectional or unidirectional. • **both**—Sets the port as bidirectional. The port cannot receive packets from or send packets to the host. By default, the port is bidirectional. • **in**—Sets the port as unidirectional. The port can send packets to the host but cannot receive packets from the host.

	Command or Action	Purpose
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show authentication sessions interface** *interface-id* **Example:** `Switch# show authentication sessions interface gigabitethernet2/0/3`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring MAC Authentication Bypass

Beginning in privileged EXEC mode, follow these steps to enable MAC authentication bypass. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication port-control auto**
4. **mab** [**eap**]
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/1`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**authentication port-control auto** **Example:** `Switch(config-if)# authentication port-control auto`	Enables 802.1x authentication on the port.
Step 4	**mab** [**eap**] **Example:** `Switch(config-if)# mab`	Enables MAC authentication bypass. (Optional) Use the **eap** keyword to configure the switch to use EAP for authorization.
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Configuring 802.1x User Distribution

Beginning in privileged EXEC mode, follow these steps to configure a VLAN group and to map a VLAN to it:

SUMMARY STEPS

1. **configure terminal**
2. **vlan group** *vlan-group-name* **vlan-list** *vlan-list*
3. **end**
4. **no vlan group** *vlan-group-name* **vlan-list** *vlan-list*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**vlan group** *vlan-group-name* **vlan-list** *vlan-list* **Example:** `Switch(config)# vlan group eng-dept vlan-list 10`	Configures a VLAN group, and maps a single VLAN or a range of VLANs to it.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**no vlan group** *vlan-group-name* **vlan-list** *vlan-list* **Example:** `Switch(config)# no vlan group eng-dept vlan-list 10`	Clears the VLAN group configuration or elements of the VLAN group configuration.

Example of Configuring VLAN Groups

This example shows how to configure the VLAN groups, to map the VLANs to the groups, to and verify the VLAN group configurations and mapping to the specified VLANs:

```
Switch(config)# vlan group eng-dept vlan-list 10

Switch(config)# show vlan group group-name eng-dept
Group Name                      Vlans Mapped
-------------                   --------------
eng-dept                        10

Switch(config)# show dot1x vlan-group all
Group Name                      Vlans Mapped
-------------                   --------------
eng-dept                        10
hr-dept                         20
```

This example shows how to add a VLAN to an existing VLAN group and to verify that the VLAN was added:

```
Switch(config)# vlan group eng-dept vlan-list 30
Switch(config)# show vlan group eng-dept
Group Name                      Vlans Mapped
-------------                   --------------
```

```
eng-dept                        10,30
```

This example shows how to remove a VLAN from a VLAN group:

```
Switch# no vlan group eng-dept vlan-list 10
```

This example shows that when all the VLANs are cleared from a VLAN group, the VLAN group is cleared:

```
Switch(config)# no vlan group eng-dept vlan-list 30
Vlan 30 is successfully cleared from vlan group eng-dept.

Switch(config)# show vlan group group-name eng-dept
```

This example shows how to clear all the VLAN groups:

```
Switch(config)# no vlan group end-dept vlan-list all
Switch(config)# show vlan-group all
```

For more information about these commands, see the *Cisco IOS Security Command Reference.*

Configuring NAC Layer 2 802.1x Validation

You can configure NAC Layer 2 802.1x validation, which is also referred to as 802.1x authentication with a RADIUS server.

Beginning in privileged EXEC mode, follow these steps to configure NAC Layer 2 802.1x validation. The procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **authentication event no-response action authorize vlan** *vlan-id*
5. **authentication periodic**
6. **authentication timer reauthenticate**
7. **end**
8. **show authentication sessions interface** *interface-id*
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/3`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**switchport mode access** **Example:** `Switch(config-if)# switchport mode access`	Sets the port to access mode only if you configured the RADIUS server.
Step 4	**authentication event no-response action authorize vlan** *vlan-id* **Example:** `Switch(config-if)# authentication event no-response action authorize vlan 8`	Specifies an active VLAN as an 802.1x guest VLAN. The range is 1 to 4094. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN, or a voice VLAN as an 802.1x guest VLAN.
Step 5	**authentication periodic** **Example:** `Switch(config-if)# authentication periodic`	Enables periodic re-authentication of the client, which is disabled by default.
Step 6	**authentication timer reauthenticate** **Example:** `Switch(config-if)# authentication timer reauthenticate`	Sets re-authentication attempt for the client (set to one hour). This command affects the behavior of the switch only if periodic re-authentication is enabled.
Step 7	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 8	**show authentication sessions interface** *interface-id* **Example:** `Switch# show authentication sessions interface gigabitethernet2/0/3`	Verifies your entries.

	Command or Action	Purpose
Step 9	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring an Authenticator Switch with NEAT

Configuring this feature requires that one switch outside a wiring closet is configured as a supplicant and is connected to an authenticator switch.

Note The *cisco-av-pairs* must be configured as *device-traffic-class=switch* on the ACS, which sets the interface as a trunk after the supplicant is successfully authenticated.

Beginning in privileged EXEC mode, follow these steps to configure a switch as an authenticator:

SUMMARY STEPS

1. **configure terminal**
2. **cisp enable**
3. **interface** *interface-id*
4. **switchport mode access**
5. **authentication port-control auto**
6. **dot1x pae authenticator**
7. **spanning-tree portfast**
8. **end**
9. **show running-config interface** *interface-id*
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**cisp enable** **Example:** `Switch(config)# cisp enable`	Enables CISP.
Step 3	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/1`	Specifies the port to be configured, and enter interface configuration mode.
Step 4	**switchport mode access** **Example:** `Switch(config-if)# switchport mode access`	Sets the port mode to **access**.
Step 5	**authentication port-control auto** **Example:** `Switch(config-if)# authentication port-control auto`	Sets the port-authentication mode to auto.
Step 6	**dot1x pae authenticator** **Example:** `Switch(config-if)# dot1x pae authenticator`	Configures the interface as a port access entity (PAE) authenticator.
Step 7	**spanning-tree portfast** **Example:** `Switch(config-if)# spanning-tree portfast trunk`	Enables Port Fast on an access port connected to a single workstation or server..
Step 8	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 9	**show running-config interface** *interface-id* **Example:** `Switch# show running-config interface gigabitethernet2/0/1`	Verifies your configuration.

	Command or Action	Purpose
Step 10	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring a Supplicant Switch with NEAT

Beginning in privileged EXEC mode, follow these steps to configure a switch as a supplicant:

SUMMARY STEPS

1. **configure terminal**
2. **cisp enable**
3. **dot1x credential**s *profile*
4. **username** *suppswitch*
5. **password** *password*
6. **dot1x supplicant force-multicast**
7. **interface** *interface-id*
8. **switchport trunk encapsulation dot1q**
9. **switchport mode trunk**
10. **dot1x pae supplicant**
11. **dot1x credentials** *profile-name*
12. **end**
13. **show running-config interface** *interface-id*
14. **copy running-config startup-config**
15. Configuring NEAT with Auto Smartports Macros

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**cisp enable** **Example:** `Switch(config)# cisp enable`	Enables CISP.
Step 3	**dot1x credentials** *profile* **Example:** `Switch(config)# dot1x credentials test`	Creates 802.1x credentials profile. This must be attached to the port that is configured as supplicant.
Step 4	**username** *suppswitch* **Example:** `Switch(config)# username suppswitch`	Creates a username.
Step 5	**password** *password* **Example:** `Switch(config)# password myswitch`	Creates a password for the new username.
Step 6	**dot1x supplicant force-multicast** **Example:** `Switch(config)# dot1x supplicant force-multicast`	Forces the switch to send only multicast EAPOL packets when it receives either unicast or multicast packets. This also allows NEAT to work on the supplicant switch in all host modes.
Step 7	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Specifies the port to be configured, and enter interface configuration mode.
Step 8	**switchport trunk encapsulation dot1q** **Example:** `Switch(config-if)# switchport trunk encapsulation dot1q`	Sets the port to trunk mode.
Step 9	**switchport mode trunk** **Example:** `Switch(config-if)# switchport mode trunk`	Configures the interface as a VLAN trunk port.

	Command or Action	Purpose
Step 10	**dot1x pae supplicant** **Example:** `Switch(config-if)# dot1x pae supplicant`	Configures the interface as a port access entity (PAE) supplicant.
Step 11	**dot1x credentials** *profile-name* **Example:** `Switch(config-if)# dot1x credentials test`	Attaches the 802.1x credentials profile to the interface.
Step 12	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 13	**show running-config interface** *interface-id* **Example:** `Switch# show running-config interface gigabitethernet1/0/1`	Verifies your configuration.
Step 14	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.
Step 15	Configuring NEAT with Auto Smartports Macros	You can also use an Auto Smartports user-defined macro instead of the switch VSA to configure the authenticator switch. For more information, see the *Auto Smartports Configuration Guide* for this release.

Configuring 802.1x Authentication with Downloadable ACLs and Redirect URLs

In addition to configuring 802.1x authentication on the switch, you need to configure the ACS. For more information, see the *Configuration Guide for Cisco Secure ACS 4.2*:
http://www.cisco.com/en/US/docs/net_mgmt/cisco_secure_access_control_server_for_windows/4.2/configuration/guide/acs_config.pdf

Note You must configure a downloadable ACL on the ACS before downloading it to the switch.

After authentication on the port, you can use the **show ip access-list** privileged EXEC command to display the downloaded ACLs on the port.

Configuring Downloadable ACLs

The policies take effect after client authentication and the client IP address addition to the IP device tracking table. The switch then applies the downloadable ACL to the port.

Beginning in privileged EXEC mode:

SUMMARY STEPS

1. **configure terminal**
2. **ip device tracking**
3. **aaa new-model**
4. **aaa authorization network default local group radius**
5. **radius-server vsa send authentication**
6. **interface** *interface-id*
7. **ip access-group** *acl-id* **in**
8. **show running-config interface** *interface-id*
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip device tracking** **Example:** `Switch(config)# ip device tracking`	Sets the ip device tracking table.
Step 3	**aaa new-model** **Example:** `Switch(config)# aaa new-model`	Enables AAA.
Step 4	**aaa authorization network default local group radius** **Example:** `Switch(config)# aaa authorization network default`	Sets the authorization method to local. To remove the authorization method, use the **no aaa authorization network default local group radius** command.

	Command or Action	Purpose
	`local group radius`	
Step 5	**radius-server vsa send authentication** **Example:** `Switch(config)# radius-server vsa send authentication`	Configures the radius vsa send authentication.
Step 6	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/4`	Specifies the port to be configured, and enter interface configuration mode.
Step 7	**ip access-group** *acl-id* **in** **Example:** `Switch(config-if)# ip access-group default_acl in`	Configures the default ACL on the port in the input direction. **Note** The *acl-id* is an access list name or number.
Step 8	**show running-config interface** *interface-id* **Example:** `Switch(config-if)# show running-config interface gigabitethernet2/0/4`	Verifies your configuration.
Step 9	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring a Downloadable Policy

Beginning in privileged EXEC mode:

SUMMARY STEPS

1. **configure terminal**
2. **access-list** *access-list-number* **{ deny | permit } { hostname | any | host } log**
3. **interface** *interface-id*
4. **ip access-group** *acl-id* **in**
5. **exit**
6. **aaa new-model**
7. **aaa authorization network default group radius**
8. **ip device tracking**
9. **ip device tracking probe** [**count** | **interval** | **use-svi**]
10. **radius-server vsa send authentication**
11. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**access-list** *access-list-number* **{ deny | permit } { hostname | any | host } log** **Example:** `Switch(config)# access-list 1 deny any log`	Defines the default port ACL. The access-list-number is a decimal number from 1 to 99 or 1300 to 1999. Enter **deny** or **permit** to specify whether to deny or permit access if conditions are matched. The source is the source address of the network or host that sends a packet, such as this: • **hostname**: The 32-bit quantity in dotted-decimal format. • **any**: The keyword any as an abbreviation for source and source-wildcard value of 0.0.0.0 255.255.255.255. You do not need to enter a source-wildcard value. • **host**: The keyword host as an abbreviation for source and source-wildcard of source 0.0.0.0. (Optional) Applies the source-wildcard wildcard bits to the source. (Optional) Enters log to cause an informational logging message about the packet that matches the entry to be sent to the console.

	Command or Action	Purpose
Step 3	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/2`	Enters interface configuration mode.
Step 4	**ip access-group** *acl-id* **in** **Example:** `Switch(config-if)# ip access-group default_acl in`	Configures the default ACL on the port in the input direction. **Note** The acl-id is an access list name or number.
Step 5	**exit** **Example:** `Switch(config-if)# exit`	Returns to global configuration mode.
Step 6	**aaa new-model** **Example:** `Switch(config)# aaa new-model`	Enables AAA.
Step 7	**aaa authorization network default group radius** **Example:** `Switch(config)# aaa authorization network default group radius`	Sets the authorization method to local. To remove the authorization method, use the **no aaa authorization network default group radius** command.
Step 8	**ip device tracking** **Example:** `Switch(config)# ip device tracking`	Enables the IP device tracking table. To disable the IP device tracking table, use the **no ip device tracking** global configuration commands.
Step 9	**ip device tracking probe** [**count** | **interval** | **use-svi**] **Example:** `Switch(config)# ip device tracking probe count`	(Optional) Configures the IP device tracking table: • **count** *count*—Sets the number of times that the switch sends the ARP probe. The range is from 1 to 5. The default is 3. • **interval** *interval*—Sets the number of seconds that the switch waits for a response before resending the ARP probe. The range is from 30 to 300 seconds. The default is 30 seconds. • **use-svi**—Uses the switch virtual interface (SVI) IP address as source of ARP probes.

	Command or Action	Purpose
Step 10	**radius-server vsa send authentication** **Example:** `Switch(config)# radius-server vsa send authentication`	Configures the network access server to recognize and use vendor-specific attributes. **Note** The downloadable ACL must be operational.
Step 11	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring VLAN ID-based MAC Authentication

Beginning in privileged EXEC mode, follow these steps:

SUMMARY STEPS

1. **configure terminal**
2. **mab request format attribute 32 vlan access-vlan**
3. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**mab request format attribute 32 vlan access-vlan** **Example:** `Switch(config)# mab request format attribute 32 vlan access-vlan`	Enables VLAN ID-based MAC authentication.

	Command or Action	Purpose
Step 3	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Flexible Authentication Ordering

The examples used in the instructions below changes the order of Flexible Authentication Ordering so that MAB is attempted before IEEE 802.1X authentication (dot1x). MAB is configured as the first authentication method, so MAB will have priority over all other authentication methods.

Note Before changing the default order and priority of these authentication methods, however, you should understand the potential consequences of those changes. See http://www.cisco.com/en/US/prod/collateral/iosswrel/ps6537/ps6586/ps6638/application_note_c27-573287_ps6638_Products_White_Paper.html for details.

Beginning in privileged EXEC mode, follow these steps:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **authentication order** [**dot1x** | **mab**] | {**webauth**}
5. **authentication priority** [**dot1x** | **mab**] | {**webauth**}
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**switchport mode access** **Example:** `Switch(config-if)# switchport mode access`	Sets the port to access mode only if you previously configured the RADIUS server.
Step 4	**authentication order** [**dot1x** | **mab**] | {**webauth**} **Example:** `Switch(config-if)# authentication order mab dot1x`	(Optional) Sets the order of authentication methods used on a port.
Step 5	**authentication priority** [**dot1x** | **mab**] | {**webauth**} **Example:** `Switch(config-if)# authentication priority mab dot1x`	(Optional) Adds an authentication method to the port-priority list.
Step 6	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Related Topics

Flexible Authentication Ordering, on page 1159

Configuring Open1x

Beginning in privileged EXEC mode, follow these steps to enable manual control of the port authorization state:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **authentication control-direction** {**both** | **in**}
5. **authentication fallback** *name*
6. **authentication host-mode** [**multi-auth** | **multi-domain** | **multi-host** | **single-host**]
7. **authentication open**
8. **authentication order** [**dot1x** | **mab**] | {**webauth**}
9. **authentication periodic**
10. **authentication port-control** {**auto** | **force-authorized** | **force-un authorized**}
11. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**switchport mode access** **Example:** `Switch(config-if)# switchport mode access`	Sets the port to access mode only if you configured the RADIUS server.
Step 4	**authentication control-direction** {**both** | **in**} **Example:** `Switch(config-if)# authentication control-direction both`	(Optional) Configures the port control as unidirectional or bidirectional.
Step 5	**authentication fallback** *name* **Example:** `Switch(config-if)# authentication fallback profile1`	(Optional) Configures a port to use web authentication as a fallback method for clients that do not support 802.1x authentication.

	Command or Action	Purpose
Step 6	**authentication host-mode** [**multi-auth** | **multi-domain** | **multi-host** | **single-host**] **Example:** `Switch(config-if)# authentication host-mode multi-auth`	(Optional) Sets the authorization manager mode on a port.
Step 7	**authentication open** **Example:** `Switch(config-if)# authentication open`	(Optional) Enables or disable open access on a port.
Step 8	**authentication order** [**dot1x** | **mab**] | {**webauth**} **Example:** `Switch(config-if)# authentication order dot1x webauth`	(Optional) Sets the order of authentication methods used on a port.
Step 9	**authentication periodic** **Example:** `Switch(config-if)# authentication periodic`	(Optional) Enables or disable reauthentication on a port.
Step 10	**authentication port-control** {**auto** | **force-authorized** | **force-un authorized**} **Example:** `Switch(config-if)# authentication port-control auto`	(Optional) Enables manual control of the port authorization state.
Step 11	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Related Topics

Open1x Authentication, on page 1159

Disabling 802.1x Authentication on the Port

You can disable 802.1x authentication on the port by using the **no dot1x pae** interface configuration command.

Beginning in privileged EXEC mode, follow these steps to disable 802.1x authentication on the port. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **no dot1x pae authenticator**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/1`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**switchport mode access** **Example:** `Switch(config-if)# switchport mode access`	(Optional) Sets the port to access mode only if you configured the RADIUS server.
Step 4	**no dot1x pae authenticator** **Example:** `Switch(config-if)# no dot1x pae authenticator`	Disables 802.1x authentication on the port.
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Resetting the 802.1x Authentication Configuration to the Default Values

Beginning in privileged EXEC mode, follow these steps to reset the 802.1x authentication configuration to the default values. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **dot1x default**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/2`	Enters interface configuration mode, and specify the port to be configured.
Step 3	**dot1x default** **Example:** `Switch(config-if)# dot1x default`	Resets the 802.1x parameters to the default values.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Monitoring 802.1x Statistics and Status

Table 136: Privileged EXEC show Commands

Command	Purpose
show dot1x all statistics	Displays 802.1x statistics for all ports
show dot1x interface *interface-id* **statistics**	Displays 802.1x statistics for a specific port
show dot1x all [**count** | **details** | **statistics** | **summary**]	Displays the 802.1x administrative and operational status for a switch
show dot1x interface *interface-id*	Displays the 802.1x administrative and operational status for a specific port

Table 137: Global Configuration Commands

Command	Purpose
no dot1x logging verbose	Filters verbose 802.1x authentication messages (beginning with Cisco IOS Release 12.2(55)SE)

For detailed information about the fields in these displays, see the command reference for this release.

Additional References

Related Documents

Related Topic	Document Title
Configuring Identity Control policies and Identity Service templates for Session Aware networking.	Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/en/US/docs/ios-xml/ios/san/configuration/xe-3se/3850/san-xe-3se-3850-book.html
Configuring RADIUS, TACACS+, Secure Shell, 802.1X and AAA.	Securing User Services Configuration Guide Library, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/en/US/docs/ios-xml/ios/security/config_library/xe-3se/3850/secuser-xe-3se-3850-library.html

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

CHAPTER 56

Configuring Web-Based Authentication

This chapter describes how to configure web-based authentication on the switch. It contains these sections:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information About Web-Based Authentication

Use the web-based authentication feature, known as web authentication proxy, to authenticate end users on host systems that do not run the IEEE 802.1x supplicant.

Note You can configure web-based authentication on Layer 2 and Layer 3 interfaces.

When you initiate an HTTP session, web-based authentication intercepts ingress HTTP packets from the host and sends an HTML login page to the users. The users enter their credentials, which the web-based authentication feature sends to the authentication, authorization, and accounting (AAA) server for authentication.

If authentication succeeds, web-based authentication sends a Login-Successful HTML page to the host and applies the access policies returned by the AAA server.

If authentication fails, web-based authentication forwards a Login-Fail HTML page to the user, prompting the user to retry the login. If the user exceeds the maximum number of attempts, web-based authentication forwards a Login-Expired HTML page to the host, and the user is placed on a watch list for a waiting period.

These sections describe the role of web-based authentication as part of AAA:

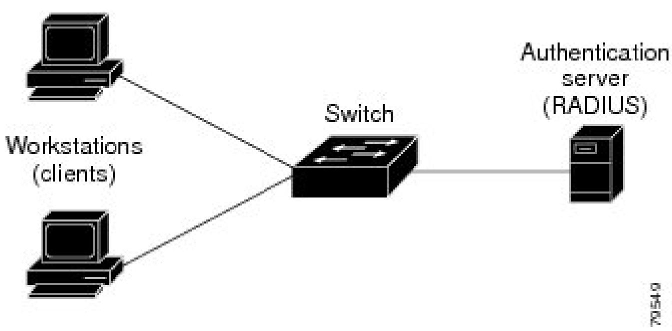
Device Roles

With web-based authentication, the devices in the network have these specific roles:

- *Client*—The device (workstation) that requests access to the LAN and the services and responds to requests from the switch. The workstation must be running an HTML browser with Java Script enabled.
- *Authentication server*—Authenticates the client. The authentication server validates the identity of the client and notifies the switch that the client is authorized to access the LAN and the switch services or that the client is denied.
- *Switch*—Controls the physical access to the network based on the authentication status of the client. The switch acts as an intermediary (proxy) between the client and the authentication server, requesting identity information from the client, verifying that information with the authentication server, and relaying a response to the client.

This figure shows the roles of these devices in a network.

Figure 55: Web-Based Authentication Device Roles



Host Detection

The switch maintains an IP device tracking table to store information about detected hosts.

Note By default, the IP device tracking feature is disabled on a switch. You must enable the IP device tracking feature to use web-based authentication.

For Layer 2 interfaces, web-based authentication detects IP hosts by using these mechanisms:

- ARP based trigger—ARP redirect ACL allows web-based authentication to detect hosts with a static IP address or a dynamic IP address.
- Dynamic ARP inspection

- DHCP snooping—Web-based authentication is notified when the switch creates a DHCP-binding entry for the host.

Session Creation

When web-based authentication detects a new host, it creates a session as follows:

- Reviews the exception list.

  If the host IP is included in the exception list, the policy from the exception list entry is applied, and the session is established.

- Reviews for authorization bypass

  If the host IP is not on the exception list, web-based authentication sends a nonresponsive-host (NRH) request to the server.

  If the server response is access accepted, authorization is bypassed for this host. The session is established.

- Sets up the HTTP intercept ACL

  If the server response to the NRH request is access rejected, the HTTP intercept ACL is activated, and the session waits for HTTP traffic from the host.

Authentication Process

When you enable web-based authentication, these events occur:

- The user initiates an HTTP session.
- The HTTP traffic is intercepted, and authorization is initiated. The switch sends the login page to the user. The user enters a username and password, and the switch sends the entries to the authentication server.
- If the authentication succeeds, the switch downloads and activates the user's access policy from the authentication server. The login success page is sent to the user.
- If the authentication fails, the switch sends the login fail page. The user retries the login. If the maximum number of attempts fails, the switch sends the login expired page, and the host is placed in a watch list. After the watch list times out, the user can retry the authentication process.
- If the authentication server does not respond to the switch, and if an AAA fail policy is configured, the switch applies the failure access policy to the host. The login success page is sent to the user.
- The switch reauthenticates a client when the host does not respond to an ARP probe on a Layer 2 interface, or when the host does not send any traffic within the idle timeout on a Layer 3 interface.
- The feature applies the downloaded timeout or the locally configured session timeout.
- If the terminate action is RADIUS, the feature sends a nonresponsive host (NRH) request to the server. The terminate action is included in the response from the server.
- If the terminate action is default, the session is dismantled, and the applied policy is removed.

Local Web Authentication Banner

With Web Authentication, you can create a default and customized web-browser banners that appears when you log in to a switch.

The banner appears on both the login page and the authentication-result pop-up pages. The default banner messages are as follows:

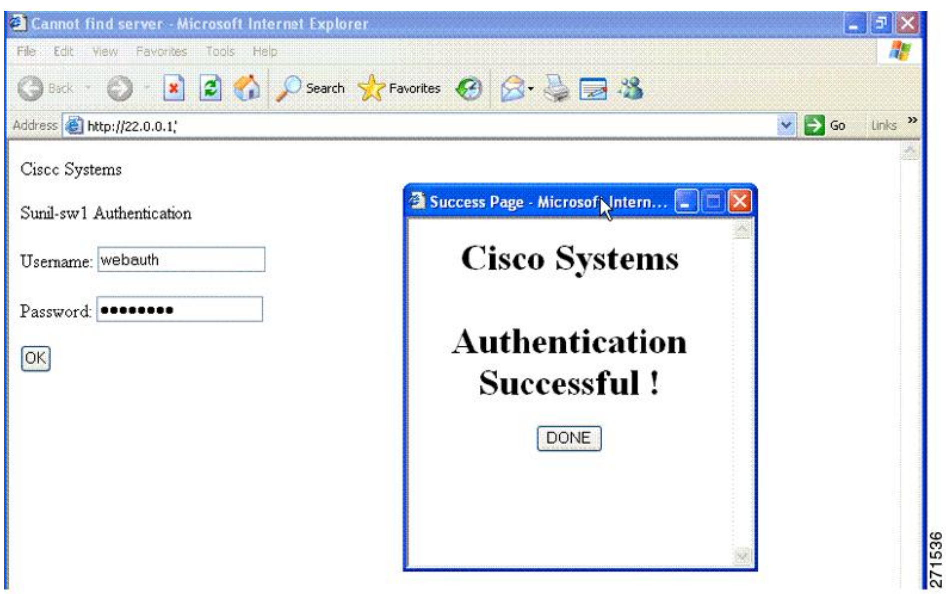
- *Authentication Successful*
- *Authentication Failed*
- *Authentication Expired*

The Local Web Authentication Banner can be configured in legacy and new-style (Session-aware) CLIs as follows:

- Legacy mode—Use the **ip admission auth-proxy-banner http** global configuration command.
- New-style mode—Use the **parameter-map type webauth global banner**global configuration command.

The default banner *Cisco Systems* and *Switch host-name Authentication* appear on the Login Page. *Cisco Systems* appears on the authentication result pop-up page.

Figure 56: Authentication Successful Banner

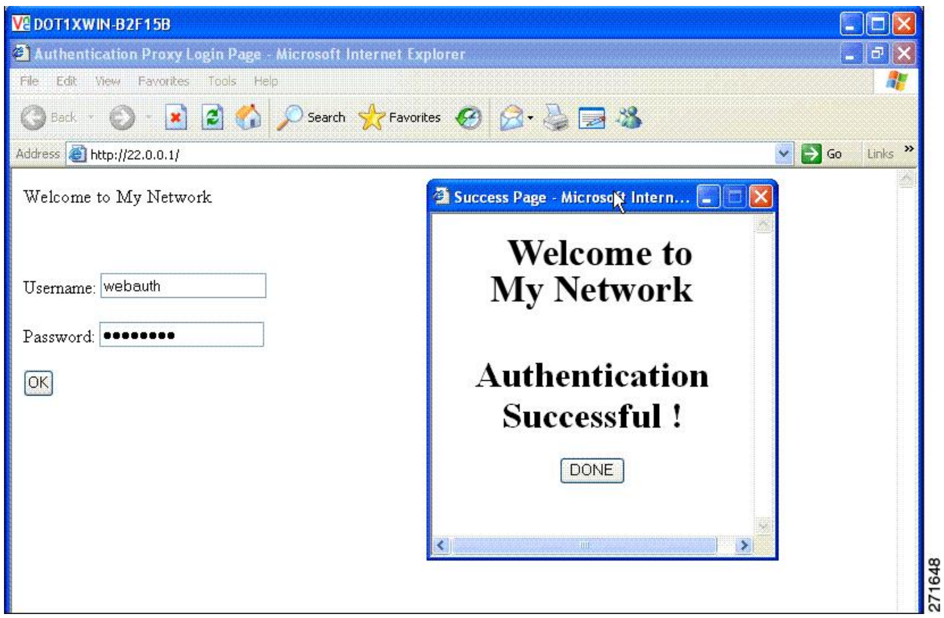


The banner can be customized as follows:

- Add a message, such as switch, router, or company name to the banner:
  - Legacy mode—Use the **ip admission auth-proxy-banner http** *banner-text*global configuration command.
  - New-style mode—Use the **parameter-map type webauth global banner**global configuration command

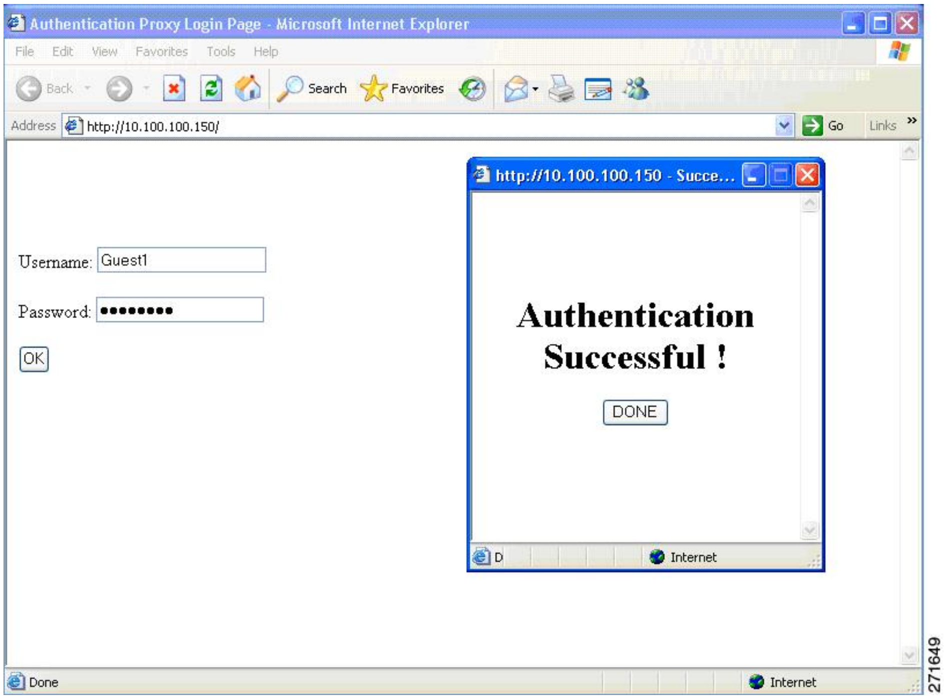
- Add a logo or text file to the banner :
  - Legacy mode—Use the **ip admission auth-proxy-banner http** *file-path* global configuration command.
  - New-style mode—Use the **parameter-map type webauth global banner** global configuration command

Figure 57: Customized Web Banner



If you do not enable a banner, only the username and password dialog boxes appear in the web authentication login screen, and no banner appears when you log into the switch.

Figure 58: Login Screen With No Banner



For more information, see the *Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)* and the *Web Authentication Enhancements - Customizing Authentication Proxy Web Pages*.

Web Authentication Customizable Web Pages

During the web-based authentication process, the switch internal HTTP server hosts four HTML pages to deliver to an authenticating client. The server uses these pages to notify you of these four-authentication process states:

- Login—Your credentials are requested.
- Success—The login was successful.
- Fail—The login failed.
- Expire—The login session has expired because of excessive login failures.

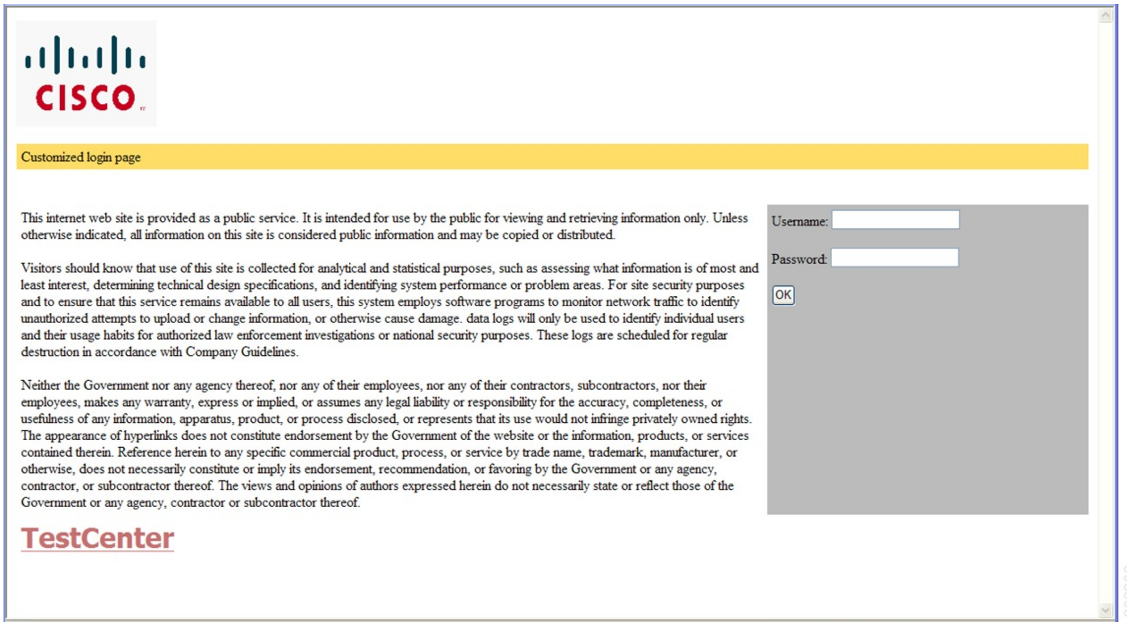
Guidelines

- You can substitute your own HTML pages for the default internal HTML pages.
- You can use a logo or specify text in the *login*, *success*, *failure*, and *expire* web pages.
- On the banner page, you can specify text in the login page.

- The pages are in HTML.
- You must include an HTML redirect command in the success page to access a specific URL.
- The URL string must be a valid URL (for example, http://www.cisco.com). An incomplete URL might cause *page not found* or similar errors on a web browser.
- If you configure web pages for HTTP authentication, they must include the appropriate HTML commands (for example, to set the page time out, to set a hidden password, or to confirm that the same page is not submitted twice).
- The CLI command to redirect users to a specific URL is not available when the configured login form is enabled. The administrator should ensure that the redirection is configured in the web page.
- If the CLI command redirecting users to specific URL after authentication occurs is entered and then the command configuring web pages is entered, the CLI command redirecting users to a specific URL does not take effect.
- Configured web pages can be copied to the switch boot flash or flash.
- On stackable switches, configured pages can be accessed from the flash on the stack master or members.
- The login page can be on one flash, and the success and failure pages can be another flash (for example, the flash on the stack master or a member).
- You must configure all four pages.
- The banner page has no effect if it is configured with the web page.
- All of the logo files (image, flash, audio, video, and so on) that are stored in the system directory (for example, flash, disk0, or disk) and that must be displayed on the login page must use *web_auth_<filename>* as the file name.
- The configured authentication proxy feature supports both HTTP and SSL.

You can substitute your HTML pages for the default internal HTML pages. You can also specify a URL to which users are redirected after authentication occurs, which replaces the internal Success page.

Figure 59: Customizeable Authentication Page



Authentication Proxy Web Page Guidelines

When configuring customized authentication proxy web pages, follow these guidelines:

- To enable the custom web pages feature, specify all four custom HTML files. If you specify fewer than four files, the internal default HTML pages are used.
- The four custom HTML files must be present on the flash memory of the switch. The maximum size of each HTML file is 8 KB.
- Any images on the custom pages must be on an accessible HTTP server. Configure an intercept ACL within the admission rule.
- Any external link from a custom page requires configuration of an intercept ACL within the admission rule.
- To access a valid DNS server, any name resolution required for external links or images requires configuration of an intercept ACL within the admission rule.
- If the custom web pages feature is enabled, a configured auth-proxy-banner is not used.
- If the custom web pages feature is enabled, the redirection URL for successful login feature is not available.
- To remove the specification of a custom file, use the **no** form of the command.

Because the custom login page is a public web form, consider these guidelines for the page:

- The login form must accept user entries for the username and password and must show them as **uname** and **pwd**.
- The custom login page should follow best practices for a web form, such as page timeout, hidden password, and prevention of redundant submissions.

Related Topics

Customizing the Authentication Proxy Web Pages, on page 1237

Redirection URL for Successful Login Guidelines

When configuring a redirection URL for successful login, consider these guidelines:

- If the custom authentication proxy web pages feature is enabled, the redirection URL feature is disabled and is not available in the CLI. You can perform redirection in the custom-login success page.
- If the redirection URL feature is enabled, a configured auth-proxy-banner is not used.
- To remove the specification of a redirection URL, use the **no** form of the command.
- If the redirection URL is required after the web-based authentication client is successfully authenticated, then the URL string must start with a valid URL (for example, http://) followed by the URL information. If only the URL is given without http://, then the redirection URL on successful authentication might cause page not found or similar errors on a web browser.

Related Topics

Specifying a Redirection URL for Successful Login, on page 1239

Web-based Authentication Interactions with Other Features

Port Security

You can configure web-based authentication and port security on the same port. Web-based authentication authenticates the port, and port security manages network access for all MAC addresses, including that of the client. You can then limit the number or group of clients that can access the network through the port.

For more information about enabling port security, see the .

LAN Port IP

You can configure LAN port IP (LPIP) and Layer 2 web-based authentication on the same port. The host is authenticated by using web-based authentication first, followed by LPIP posture validation. The LPIP host policy overrides the web-based authentication host policy.

If the web-based authentication idle timer expires, the NAC policy is removed. The host is authenticated, and posture is validated again.

Gateway IP

You cannot configure Gateway IP (GWIP) on a Layer 3 VLAN interface if web-based authentication is configured on any of the switch ports in the VLAN.

You can configure web-based authentication on the same Layer 3 interface as Gateway IP. The host policies for both features are applied in software. The GWIP policy overrides the web-based authentication host policy.

ACLs

If you configure a VLAN ACL or a Cisco IOS ACL on an interface, the ACL is applied to the host traffic only after the web-based authentication host policy is applied.

For Layer 2 web-based authentication, it is more secure, though not required, to configure a port ACL (PACL) as the default access policy for ingress traffic from hosts connected to the port. After authentication, the web-based authentication host policy overrides the PACL. The Policy ACL is applied to the session even if there is no ACL configured on the port.

You cannot configure a MAC ACL and web-based authentication on the same interface.

You cannot configure web-based authentication on a port whose access VLAN is configured for VACL capture.

Context-Based Access Control

Web-based authentication cannot be configured on a Layer 2 port if context-based access control (CBAC) is configured on the Layer 3 VLAN interface of the port VLAN.

EtherChannel

You can configure web-based authentication on a Layer 2 EtherChannel interface. The web-based authentication configuration applies to all member channels.

How to Configure Web-Based Authentication

Default Web-Based Authentication Configuration

The following table shows the default web-based authentication configuration.

Table 138: Default Web-based Authentication Configuration

Feature	Default Setting
AAA	Disabled
RADIUS server • IP address • UDP authentication port • Key	• None specified • 1645 • None specified
Default value of inactivity timeout	3600 seconds
Inactivity timeout	Enabled

Web-Based Authentication Configuration Guidelines and Restrictions

- Web-based authentication is an ingress-only feature.
- You can configure web-based authentication only on access ports. Web-based authentication is not supported on trunk ports, EtherChannel member ports, or dynamic trunk ports.
- You cannot authenticate hosts on Layer 2 interfaces with static ARP cache assignment. These hosts are not detected by the web-based authentication feature because they do not send ARP messages.
- By default, the IP device tracking feature is disabled on a switch. You must enable the IP device tracking feature to use web-based authentication.
- You must configure at least one IP address to run the switch HTTP server. You must also configure routes to reach each host IP address. The HTTP server sends the HTTP login page to the host.
- Hosts that are more than one hop away might experience traffic disruption if an STP topology change results in the host traffic arriving on a different port. This occurs because the ARP and DHCP updates might not be sent after a Layer 2 (STP) topology change.
- Web-based authentication does not support VLAN assignment as a downloadable-host policy.
- Web-based authentication supports IPv6 in Session-aware policy mode. IPv6 Web-authentication requires at least one IPv6 address configured on the switch and IPv6 Snooping configured on the switchport.
- Web-based authentication and Network Edge Access Topology (NEAT) are mutually exclusive. You cannot use web-based authentication when NEAT is enabled on an interface, and you cannot use NEAT when web-based authentication is running on an interface.

Configuring the Authentication Rule and Interfaces

Examples in this section are legacy-style configurations. For new-style configurations, see the *Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*

This example shows how to verify the configuration:

```
Switch# show ip admission status
IP admission status:
  Enabled interfaces            0
  Total sessions                0
  Init sessions                 0     Max init sessions allowed     100
    Limit reached               0     Hi watermark                  0
  TCP half-open connections     0     Hi watermark                  0
  TCP new connections           0     Hi watermark                  0
  TCP half-open + new           0     Hi watermark                  0
  HTTPD1 Contexts               0     Hi watermark                  0

  Parameter Map: Global
    Custom Pages
      Custom pages not configured
    Banner
      Banner not configured
```

Beginning in privileged EXEC mode, follow these steps to configure the authentication rule and interfaces:

SUMMARY STEPS

1. **configure terminal**
2. **ip admission name** *name* **proxy http**
3. **interface** *type slot/port*
4. **ip access-group** *name*
5. **ip admission** *name*
6. **exit**
7. **ip device tracking**
8. **end**
9. **show ip admission status**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip admission name** *name* **proxy http** **Example:** `Switch(config)# ip admission name webauth1 proxy http`	Configures an authentication rule for web-based authorization.
Step 3	**interface** *type slot/port* **Example:** `Switch(config)# interface gigabitEthernet1/0/1`	Enters interface configuration mode and specifies the ingress Layer 2 or Layer 3 interface to be enabled for web-based authentication. *type* can be fastethernet, gigabit ethernet, or tengigabitethernet.
Step 4	**ip access-group** *name* **Example:** `Switch(config-if)# ip access-group webauthag`	Applies the default ACL.
Step 5	**ip admission** *name* **Example:** `Switch(config-if)# ip admission webauth1`	Configures web-based authentication on the specified interface.

	Command or Action	Purpose
Step 6	**exit** **Example:** `Switch(config-if)# exit`	Returns to configuration mode.
Step 7	**ip device tracking** **Example:** `Switch(config)# ip device tracking`	Enables the IP device tracking table.
Step 8	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 9	**show ip admission status** **Example:** `Switch# show ip admission status`	Displays the configuration.
Step 10	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring AAA Authentication

Beginning in privileged EXEC mode, follow these steps to configure AAA authentication:

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **aaa authentication login default group** {**tacacs+** | **radius**}
4. **aaa authorization auth-proxy default group** {**tacacs+** | **radius**}
5. **tacacs-server host** {*hostname* | *ip_address*}
6. **tacacs-server key** {*key-data*}
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**aaa new-model** **Example:** `Switch(config)# aaa new-model`	Enables AAA functionality.
Step 3	**aaa authentication login default group** {**tacacs+** | **radius**} **Example:** `Switch(config)# aaa authentication login default group tacacs+`	Defines the list of authentication methods at login.
Step 4	**aaa authorization auth-proxy default group** {**tacacs+** | **radius**} **Example:** `Switch(config)# aaa authorization auth-proxy default group tacacs+`	Creates an authorization method list for web-based authorization.
Step 5	**tacacs-server host** {*hostname* | *ip_address*} **Example:** `Switch(config)# tacacs-server host 10.1.1.1`	Specifies an AAA server.
Step 6	**tacacs-server key** {*key-data*} **Example:** `Switch(config)# tacacs-server key`	Configures the authorization and encryption key used between the switch and the TACACS server.
Step 7	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring Switch-to-RADIUS-Server Communication

Beginning in privileged EXEC mode, follow these steps to configure the RADIUS server parameters:

Before You Begin

Identify the following RADIUS security server settings that will be used in theses instructions:

- Host name
- Host IP address
- Host name and specific UDP port numbers
- IP address and specific UDP port numbers

The combination of the IP address and UDP port number creates a unique identifier, that enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address. If two different host entries on the same RADIUS server are configured for the same service (for example, authentication) the second host entry that is configured functions as the failover backup to the first one. The RADIUS host entries are chosen in the order that they were configured.

SUMMARY STEPS

1. **configure terminal**
2. **ip radius source-interface vlan** *vlan interface number*
3. **radius-server host** {*hostname* | *ip-address*} **test username** *username*
4. **radius-server key** *string*
5. **radius-server dead-criteria tries** *num-tries*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip radius source-interface vlan** *vlan interface number* **Example:** `Switch(config)# ip radius source-interface vlan 80`	Specifies that the RADIUS packets have the IP address of the indicated interface.
Step 3	**radius-server host** {*hostname* | *ip-address*} **test username** *username*	Specifies the host name or IP address of the remote RADIUS server.

	Command or Action	Purpose
	Example: `Switch(config)# radius-server host 172.120.39.46 test username user1`	The **test username** *username* option enables automated testing of the RADIUS server connection. The specified *username* does not need to be a valid user name. The **key** option specifies an authentication and encryption key to use between the switch and the RADIUS server. To use multiple RADIUS servers, reenter this command for each server.
Step 4	**radius-server key** *string* **Example:** `Switch(config)# radius-server key rad123`	Configures the authorization and encryption key used between the switch and the RADIUS daemon running on the RADIUS server.
Step 5	**radius-server dead-criteria tries** *num-tries* **Example:** `Switch(config)# radius-server dead-criteria tries 30`	Specifies the number of unanswered sent messages to a RADIUS server before considering the server to be inactive. The range of *num-tries* is 1 to 100. When you configure the RADIUS server parameters: • Specify the **key** *string* on a separate command line. • For **key** *string*, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. The key is a text string that must match the encryption key used on the RADIUS server. • When you specify the **key** *string*, use spaces within and at the end of the key. If you use spaces in the key, do not enclose the key in quotation marks unless the quotation marks are part of the key. This key must match the encryption used on the RADIUS daemon. • You can globally configure the timeout, retransmission, and encryption key values for all RADIUS servers by using with the **radius-server host** global configuration command. If you want to configure these options on a per-server basis, use the **radius-server timeout**, radius-server transmit, and the radius-server key global configuration commands. For more information, see the *Cisco IOS Security Configuration Guide*, Release 12.4 and the *Cisco IOS Security Command Reference*, Release 12.4. **Note** You need to configure some settings on the RADIUS server, including: the switch IP address, the key string to be shared by both the server and the switch, and the downloadable ACL (DACL). For more information, see the RADIUS server documentation.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring the HTTP Server

To use web-based authentication, you must enable the HTTP server within the switch. You can enable the server for either HTTP or HTTPS.

Beginning in privileged EXEC mode, follow these steps to enable the server for either HTTP or HTTPS:

SUMMARY STEPS

1. **configure terminal**
2. **ip http server**
3. **ip http secure-server**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip http server** **Example:** `Switch(config)# ip http server`	Enables the HTTP server. The web-based authentication feature uses the HTTP server to communicate with the hosts for user authentication.
Step 3	**ip http secure-server** **Example:** `Switch(config)# ip http secure-server`	Enables HTTPS. You can configure custom authentication proxy web pages or specify a redirection URL for successful login. **Note** To ensure secure authentication when you enter the **ip http secure-server** command, the login page is always in HTTPS (secure HTTP) even if the user sends an HTTP request.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Customizing the Authentication Proxy Web Pages

You can configure web authentication to display four substitute HTML pages to the user in place of the switch default HTML pages during web-based authentication.

For the equivalent Session Aware Networking configuration example for this feature, see the section "Configuring a Parameter Map for Web-Based Authentication" in the chapter, "Configuring Identity Control Policies." of the book, "*Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*."

Beginning in privileged EXEC mode, follow these steps to specify the use of your custom authentication proxy web pages:

Before You Begin

Store your custom HTML files on the switch flash memory.

SUMMARY STEPS

1. **configure terminal**
2. **ip admission proxy http login page file** *device:login-filename*
3. **ip admission proxy http success page file** *device:success-filename*
4. **ip admission proxy http failure page file** *device:fail-filename*
5. **ip admission proxy http login expired page file** *device:expired-filename*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip admission proxy http login page file** *device:login-filename* **Example:** `Switch(config)# ip admission proxy http login page file disk1:login.htm`	Specifies the location in the switch memory file system of the custom HTML file to use in place of the default login page. The *device:* is flash memory.
Step 3	**ip admission proxy http success page file** *device:success-filename* **Example:** `Switch(config)# ip admission proxy http success page file disk1:success.htm`	Specifies the location of the custom HTML file to use in place of the default login success page.
Step 4	**ip admission proxy http failure page file** *device:fail-filename* **Example:** `Switch(config)# ip admission proxy http fail page`	Specifies the location of the custom HTML file to use in place of the default login failure page.

	Command or Action	Purpose
	`file disk1:fail.htm`	
Step 5	**ip admission proxy http login expired page file** *device:expired-filename* **Example:** `Switch(config)# ip admission proxy http login expired page file disk1:expired.htm`	Specifies the location of the custom HTML file to use in place of the default login expired page.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Verifying Custom Authentication Proxy Web Pages

This example shows how to verify the configuration of a custom authentication proxy web page:

```
Switch# show ip admission status
IP admission status:
  Enabled interfaces            0
  Total sessions                0
  Init sessions                 0     Max init sessions allowed     100
    Limit reached               0     Hi watermark                  0
  TCP half-open connections     0     Hi watermark                  0
  TCP new connections           0     Hi watermark                  0
  TCP half-open + new           0     Hi watermark                  0
  HTTPD1 Contexts               0     Hi watermark                  0

  Parameter Map: Global
    Custom Pages
      Custom pages not configured
    Banner
      Banner not configured
```

Related Topics

Authentication Proxy Web Page Guidelines, on page 1228

Specifying a Redirection URL for Successful Login

Beginning in privileged EXEC mode, follow these steps to specify a URL to which the user is redirected after authentication, effectively replacing the internal Success HTML page:

SUMMARY STEPS

1. **configure terminal**
2. **ip admission proxy http success redirect** *url-string*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip admission proxy http success redirect** *url-string* **Example:** `Switch(config)# ip admission proxy http success redirect www.example.com`	Specifies a URL for redirection of the user in place of the default login success page.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Verifying Redirection URL for Successful Login

```
Switch# show ip admission status
 Enabled interfaces             0
 Total sessions                 0
 Init sessions                  0      Max init sessions allowed     100
   Limit reached                0      Hi watermark                  0
 TCP half-open connections      0      Hi watermark                  0
 TCP new connections            0      Hi watermark                  0
 TCP half-open + new            0      Hi watermark                  0
 HTTPD1 Contexts                0      Hi watermark                  0

 Parameter Map: Global
   Custom Pages
     Custom pages not configured
   Banner
     Banner not configured
```

Related Topics

Redirection URL for Successful Login Guidelines, on page 1229

Configuring the Web-Based Authentication Parameters

Beginning in privileged EXEC mode, follow these steps to configure the maximum number of failed login attempts before the client is placed in a watch list for a waiting period:

SUMMARY STEPS

1. **configure terminal**
2. **ip admission max-login-attempts** *number*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip admission max-login-attempts** *number* **Example:** `Switch(config)# ip admission max-login-attempts 10`	Set sthe maximum number of failed login attempts. The range is 1 to 2147483647 attempts. The default is 5.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring a Web Authentication Local Banner

Beginning in privileged EXEC mode, follow these steps to configure a local banner on a switch that has web authentication configured.

SUMMARY STEPS

1. **configure terminal**
2. **ip admission auth-proxy-banner http** [*banner-text* | *file-path*]
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip admission auth-proxy-banner http** [*banner-text* | *file-path*] **Example:** `Switch(config)# ip admission auth-proxy-banner http C My Switch C`	Enables the local banner. (Optional) Create a custom banner by entering *C banner-text C* (where *C* is a delimiting character), or *file-path* that indicates a file (for example, a logo or text file) that appears in the banner.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Removing Web-Based Authentication Cache Entries

Beginning in privileged EXEC mode, follow these steps to remove web-based authentication cache entries:

SUMMARY STEPS

1. **clear ip auth-proxy cache** {* | *host ip address*}
2. **clear ip admission cache** {* | *host ip address*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**clear ip auth-proxy cache** {* | *host ip address*} **Example:** `Switch# clear ip auth-proxy cache 192.168.4.5`	Delete authentication proxy entries. Use an asterisk to delete all cache entries. Enter a specific IP address to delete the entry for a single host.

	Command or Action	Purpose
Step 2	**clear ip admission cache** {* | *host ip address*} **Example:** `Switch# clear ip admission cache 192.168.4.5`	Delete authentication proxy entries. Use an asterisk to delete all cache entries. Enter a specific IP address to delete the entry for a single host.

Monitoring Web-Based Authentication Status

Use the commands in this topic to display the web-based authentication settings for all interfaces or for specific ports.

Table 139: Privileged EXEC show Commands

Command	Purpose
show authentication sessions method webauth	Displays the web-based authentication settings for all interfaces for fastethernet, gigabitethernet, or tengigabitethernet
show authentication sessions interface *type slot/port*[**details**]	Displays the web-based authentication settings for the specified interface for fastethernet, gigabitethernet, or tengigabitethernet. In Session Aware Networking mode, use the **show access-session interface** command.

CHAPTER 57

Configuring Port-Based Traffic Control

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information About Storm Control

Storm Control

Storm control prevents traffic on a LAN from being disrupted by a broadcast, multicast, or unicast storm on one of the physical interfaces. A LAN storm occurs when packets flood the LAN, creating excessive traffic and degrading network performance. Errors in the protocol-stack implementation, mistakes in network configurations, or users issuing a denial-of-service attack can cause a storm.

Storm control (or traffic suppression) monitors packets passing from an interface to the switching bus and determines if the packet is unicast, multicast, or broadcast. The switch counts the number of packets of a specified type received within the 1-second time interval and compares the measurement with a predefined suppression-level threshold.

How Traffic Activity is Measured

Storm control uses one of these methods to measure traffic activity:

- Bandwidth as a percentage of the total available bandwidth of the port that can be used by the broadcast, multicast, or unicast traffic
- Traffic rate in packets per second at which broadcast, multicast, or unicast packets are received
- Traffic rate in bits per second at which broadcast, multicast, or unicast packets are received
- Traffic rate in packets per second and for small frames. This feature is enabled globally. The threshold for small frames is configured for each interface.

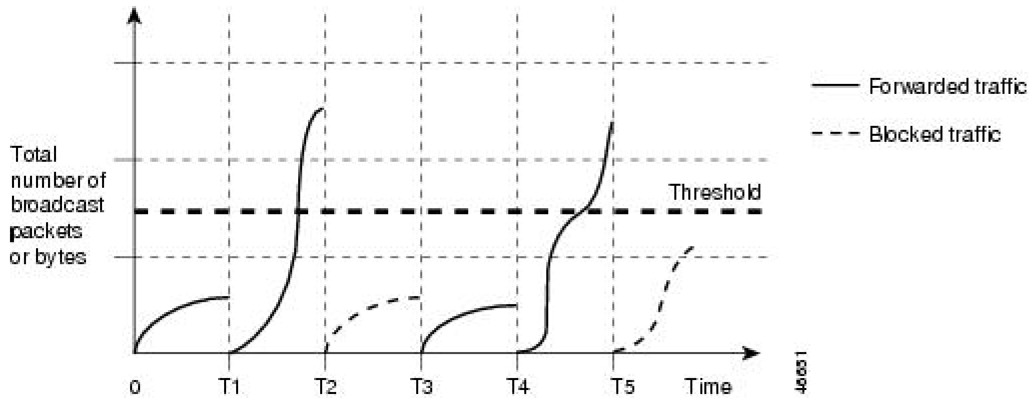
With each method, the port blocks traffic when the rising threshold is reached. The port remains blocked until the traffic rate drops below the falling threshold (if one is specified) and then resumes normal forwarding. If the falling suppression level is not specified, the switch blocks all traffic until the traffic rate drops below the rising suppression level. In general, the higher the level, the less effective the protection against broadcast storms.

Note When the storm control threshold for multicast traffic is reached, all multicast traffic except control traffic, such as bridge protocol data unit (BDPU) and Cisco Discovery Protocol (CDP) frames, are blocked. However, the switch does not differentiate between routing updates, such as OSPF, and regular multicast data traffic, so both types of traffic are blocked.

Traffic Patterns

This example shows broadcast traffic patterns on an interface over a given period of time.

Figure 60: Broadcast Storm Control Example



Broadcast traffic being forwarded exceeded the configured threshold between time intervals T1 and T2 and between T4 and T5. When the amount of specified traffic exceeds the threshold, all traffic of that kind is dropped for the next time period. Therefore, broadcast traffic is blocked during the intervals following T2 and T5. At the next time interval (for example, T3), if broadcast traffic does not exceed the threshold, it is again forwarded.

The combination of the storm-control suppression level and the 1-second time interval controls the way the storm control algorithm works. A higher threshold allows more packets to pass through. A threshold value of 100 percent means that no limit is placed on the traffic. A value of 0.0 means that all broadcast, multicast, or unicast traffic on that port is blocked.

Note Because packets do not arrive at uniform intervals, the 1-second time interval during which traffic activity is measured can affect the behavior of storm control.

You use the **storm-control** interface configuration commands to set the threshold value for each traffic type.

How to Configure Storm Control

Configuring Storm Control and Threshold Levels

You configure storm control on a port and enter the threshold level that you want to be used for a particular type of traffic.

However, because of hardware limitations and the way in which packets of different sizes are counted, threshold percentages are approximations. Depending on the sizes of the packets making up the incoming traffic, the actual enforced threshold might differ from the configured level by several percentage points.

Before You Begin

Storm control is supported on physical interfaces. You can also configure storm control on an EtherChannel. When storm control is configured on an EtherChannel, the storm control settings propagate to the EtherChannel physical interfaces.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **storm-control** {**broadcast** | **multicast** | **unicast**} **level** {*level* [*level-low*] | **bps** *bps* [*bps-low*] | **pps** *pps* [*pps-low*]}
4. **storm-control action** {**shutdown** | **trap**}
5. **end**
6. **show storm-control** [*interface-id*] [**broadcast** | **multicast** | **unicast**]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Specifies the interface to be configured, and enter interface configuration mode.
Step 3	**storm-control** {**broadcast** | **multicast** | **unicast**} **level** {*level* [*level-low*] | **bps** *bps* [*bps-low*] | **pps** *pps* [*pps-low*]} **Example:** `Switch(config-if)# storm-control unicast level 87 65`	Configures broadcast, multicast, or unicast storm control. By default, storm control is disabled. The keywords have these meanings: • For *level*, specifies the rising threshold level for broadcast, multicast, or unicast traffic as a percentage (up to two decimal places) of the bandwidth. The port blocks traffic when the rising threshold is reached. The range is 0.00 to 100.00. • (Optional) For *level-low*, specifies the falling threshold level as a percentage (up to two decimal places) of the bandwidth. This value must be less than or equal to the rising suppression value. The port forwards traffic when traffic drops below this level. If you do not configure a falling suppression level, it is set to the rising suppression level. The range is 0.00 to 100.00. If you set the threshold to the maximum value (100 percent), no limit is placed on the traffic. If you set the threshold to 0.0, all broadcast, multicast, and unicast traffic on that port is blocked.

	Command or Action	Purpose
		• For **bps** *bps*, specifies the rising threshold level for broadcast, multicast, or unicast traffic in bits per second (up to one decimal place). The port blocks traffic when the rising threshold is reached. The range is 0.0 to 10000000000.0. • (Optional) For *bps-low*, specifies the falling threshold level in bits per second (up to one decimal place). It can be less than or equal to the rising threshold level. The port forwards traffic when traffic drops below this level. The range is 0.0 to 10000000000.0. • For **pps** *pps*, specifies the rising threshold level for broadcast, multicast, or unicast traffic in packets per second (up to one decimal place). The port blocks traffic when the rising threshold is reached. The range is 0.0 to 10000000000.0. • (Optional) For *pps-low*, specifies the falling threshold level in packets per second (up to one decimal place). It can be less than or equal to the rising threshold level. The port forwards traffic when traffic drops below this level. The range is **0.0 to** 10000000000.0. For BPS and PPS settings, you can use metric suffixes such as k, m, and g for large number thresholds.
Step 4	**storm-control action** {**shutdown** | **trap**} **Example:** `Switch(config-if)# storm-control action trap`	Specifies the action to be taken when a storm is detected. The default is to filter out the traffic and not to send traps. • Select the **shutdown** keyword to error-disable the port during a storm. • Select the **trap** keyword to generate an SNMP trap when a storm is detected.
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**show storm-control** [*interface-id*] [**broadcast** | **multicast** | **unicast**] **Example:** `Switch# show storm-control gigabitethernet1/0/1 unicast`	Verifies the storm control suppression levels set on the interface for the specified traffic type. If you do not enter a traffic type, broadcast storm control settings are displayed.
Step 7	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring Storm Control

Table 140: Commands for Displaying Storm Control Status and Configuration

Command	Purpose
show interfaces [*interface-id*] **switchport**	Displays the administrative and operational status of all switching (nonrouting) ports or the specified port, including port blocking and port protection settings.
show storm-control [*interface-id*] [**broadcast** | **multicast** | **unicast**]	Displays storm control suppression levels set on all interfaces or the specified interface for the specified traffic type or for broadcast traffic if no traffic type is entered.

Information About Protected Ports

Protected Ports

Some applications require that no traffic be forwarded at Layer 2 between ports on the same switch so that one neighbor does not see the traffic generated by another neighbor. In such an environment, the use of protected ports ensures that there is no exchange of unicast, broadcast, or multicast traffic between these ports on the switch.

Protected ports have these features:

- A protected port does not forward any traffic (unicast, multicast, or broadcast) to any other port that is also a protected port. Data traffic cannot be forwarded between protected ports at Layer 2; only control traffic, such as PIM packets, is forwarded because these packets are processed by the CPU and forwarded in software. All data traffic passing between protected ports must be forwarded through a Layer 3 device.
- Forwarding behavior between a protected port and a nonprotected port proceeds as usual.

Because a switch stack represents a single logical switch, Layer 2 traffic is not forwarded between any protected ports in the switch stack, whether they are on the same or different switches in the stack.

Default Protected Port Configuration

The default is to have no protected ports defined.

Protected Ports Guidelines

You can configure protected ports on a physical interface (for example, Gigabit Ethernet port 1) or an EtherChannel group (for example, port-channel 5). When you enable protected ports for a port channel, it is enabled for all ports in the port-channel group.

How to Configure Protected Ports

Configuring a Protected Port

Before You Begin

Protected ports are not pre-defined. This is the task to configure one.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport protected**
4. **end**
5. **show interfaces** *interface-id* **switchport**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Specifies the interface to be configured, and enter interface configuration mode.
Step 3	**switchport protected** **Example:** `Switch(config-if)# switchport protected`	Configures the interface to be a protected port.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	**show interfaces** *interface-id* **switchport** **Example:** `Switch#` **`show interfaces gigabitethernet1/0/1 switchport`**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch#` **`copy running-config startup-config`**	(Optional) Saves your entries in the configuration file.

Monitoring Protected Ports

Table 141: Commands for Displaying Protected Port Settings

Command	Purpose
show interfaces [*interface-id*] **switchport**	Displays the administrative and operational status of all switching (nonrouting) ports or the specified port, including port blocking and port protection settings.

Information About Port Blocking

Port Blocking

By default, the switch floods packets with unknown destination MAC addresses out of all ports. If unknown unicast and multicast traffic is forwarded to a protected port, there could be security issues. To prevent unknown unicast or multicast traffic from being forwarded from one port to another, you can block a port (protected or nonprotected) from flooding unknown unicast or multicast packets to other ports.

Note With multicast traffic, the port blocking feature blocks only pure Layer 2 packets. Multicast packets that contain IPv4 or IPv6 information in the header are not blocked.

How to Configure Port Blocking

Blocking Flooded Traffic on an Interface

Before You Begin

The interface can be a physical interface or an EtherChannel group. When you block multicast or unicast traffic for a port channel, it is blocked on all ports in the port-channel group.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport block multicast**
4. **switchport block unicast**
5. **end**
6. **show interfaces** *interface-id* **switchport**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Specifies the interface to be configured, and enter interface configuration mode.
Step 3	**switchport block multicast** **Example:** `Switch(config-if)# switchport block multicast`	Blocks unknown multicast forwarding out of the port. **Note** Only pure Layer 2 multicast traffic is blocked. Multicast packets that contain IPv4 or IPv6 information in the header are not blocked.
Step 4	**switchport block unicast** **Example:** `Switch(config-if)# switchport block unicast`	Blocks unknown unicast forwarding out of the port.

	Command or Action	Purpose
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**show interfaces** *interface-id* **switchport** **Example:** `Switch# show interfaces gigabitethernet1/0/1 switchport`	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring Port Blocking

Table 142: Commands for Displaying Port Blocking Settings

Command	Purpose
show interfaces [*interface-id*] **switchport**	Displays the administrative and operational status of all switching (nonrouting) ports or the specified port, including port blocking and port protection settings.

Prerequisites for Port Security

Note If you try to set the maximum value to a number less than the number of secure addresses already configured on an interface, the command is rejected.

Restrictions for Port Security

The maximum number of secure MAC addresses that you can configure on a switch or switch stack is set by the maximum number of available MAC addresses allowed in the system. This number is determined by the active Switch Database Management (SDM) template. This number is the total of available MAC addresses, including those used for other Layer 2 functions and any other secure MAC addresses configured on interfaces.

Information About Port Security

Port Security

You can use the port security feature to restrict input to an interface by limiting and identifying MAC addresses of the stations allowed to access the port. When you assign secure MAC addresses to a secure port, the port does not forward packets with source addresses outside the group of defined addresses. If you limit the number of secure MAC addresses to one and assign a single secure MAC address, the workstation attached to that port is assured the full bandwidth of the port.

If a port is configured as a secure port and the maximum number of secure MAC addresses is reached, when the MAC address of a station attempting to access the port is different from any of the identified secure MAC addresses, a security violation occurs. Also, if a station with a secure MAC address configured or learned on one secure port attempts to access another secure port, a violation is flagged.

Related Topics

Enabling and Configuring Port Security, on page 504
Configuration Examples for Port Security, on page 511
Enabling and Configuring Port Security, on page 504
Configuration Examples for Port Security, on page 511

Types of Secure MAC Addresses

The switch supports these types of secure MAC addresses:

- Static secure MAC addresses—These are manually configured by using the **switchport port-security mac-address** *mac-address* interface configuration command, stored in the address table, and added to the switch running configuration.
- Dynamic secure MAC addresses—These are dynamically configured, stored only in the address table, and removed when the switch restarts.
- Sticky secure MAC addresses—These can be dynamically learned or manually configured, stored in the address table, and added to the running configuration. If these addresses are saved in the configuration file, when the switch restarts, the interface does not need to dynamically reconfigure them.

Sticky Secure MAC Addresses

You can configure an interface to convert the dynamic MAC addresses to sticky secure MAC addresses and to add them to the running configuration by enabling sticky learning. The interface converts all the dynamic secure MAC addresses, including those that were dynamically learned before sticky learning was enabled, to sticky secure MAC addresses. All sticky secure MAC addresses are added to the running configuration.

The sticky secure MAC addresses do not automatically become part of the configuration file, which is the startup configuration used each time the switch restarts. If you save the sticky secure MAC addresses in the configuration file, when the switch restarts, the interface does not need to relearn these addresses. If you do not save the sticky secure addresses, they are lost.

If sticky learning is disabled, the sticky secure MAC addresses are converted to dynamic secure addresses and are removed from the running configuration.

Security Violations

It is a security violation when one of these situations occurs:

- The maximum number of secure MAC addresses have been added to the address table, and a station whose MAC address is not in the address table attempts to access the interface.
- An address learned or configured on one secure interface is seen on another secure interface in the same VLAN.

You can configure the interface for one of three violation modes, based on the action to be taken if a violation occurs:

- protect—when the number of secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. You are not notified that a security violation has occurred.

  Note We do not recommend configuring the protect violation mode on a trunk port. The protect mode disables learning when any VLAN reaches its maximum limit, even if the port has not reached its maximum limit.

- restrict—when the number of secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. In this mode, you are notified that a security violation has occurred. An SNMP trap is sent, a syslog message is logged, and the violation counter increments.
- shutdown—a port security violation causes the interface to become error-disabled and to shut down immediately, and the port LED turns off. When a secure port is in the error-disabled state, you can bring it out of this state by entering the **errdisable recovery cause** *psecure-violation* global configuration command, or you can manually re-enable it by entering the **shutdown** and **no shut down** interface configuration commands. This is the default mode.
- shutdown vlan—Use to set the security violation mode per-VLAN. In this mode, the VLAN is error disabled instead of the entire port when a violation occurs

This table shows the violation mode and the actions taken when you configure an interface for port security.

Table 143: Security Violation Mode Actions

Violation Mode	Traffic is forwarded [23]	Sends SNMP trap	Sends syslog message	Displays error message [24]	Violation counter increments	Shuts down port
protect	No	No	No	No	No	No
restrict	No	Yes	Yes	No	Yes	No
shutdown	No	No	No	No	Yes	Yes
shutdown vlan	No	No	Yes	No	Yes	No [25]

23 Packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses.
24 The switch returns an error message if you manually configure an address that would cause a security violation.
25 Shuts down only the VLAN on which the violation occurred.

Port Security Aging

You can use port security aging to set the aging time for all secure addresses on a port. Two types of aging are supported per port:

- Absolute—The secure addresses on the port are deleted after the specified aging time.
- Inactivity—The secure addresses on the port are deleted only if the secure addresses are inactive for the specified aging time.

Related Topics

Enabling and Configuring Port Security Aging, on page 508
Enabling and Configuring Port Security Aging, on page 508

Port Security and Switch Stacks

When a switch joins a stack, the new switch will get the configured secure addresses. All dynamic secure addresses are downloaded by the new stack member from the other stack members.

When a switch (either the active switch or a stack member) leaves the stack, the remaining stack members are notified, and the secure MAC addresses configured or learned by that switch are deleted from the secure MAC address table.

Default Port Security Configuration

Table 144: Default Port Security Configuration

Feature	Default Setting
Port security	Disabled on a port.
Sticky address learning	Disabled.
Maximum number of secure MAC addresses per port	1.
Violation mode	Shutdown. The port shuts down when the maximum number of secure MAC addresses is exceeded.
Port security aging	Disabled. Aging time is 0. Static aging is disabled. Type is absolute.

Port Security Configuration Guidelines

- Port security can only be configured on static access ports or trunk ports.
- A secure port cannot be a destination port for Switched Port Analyzer (SPAN).
-

  Note Voice VLAN is only supported on access ports and not on trunk ports, even though the configuration is allowed.

- When you enable port security on an interface that is also configured with a voice VLAN, set the maximum allowed secure addresses on the port to two. When the port is connected to a Cisco IP phone, the IP phone requires one MAC address. The Cisco IP phone address is learned on the voice VLAN, but is not learned on the access VLAN. If you connect a single PC to the Cisco IP phone, no additional MAC addresses are required. If you connect more than one PC to the Cisco IP phone, you must configure enough secure addresses to allow one for each PC and one for the phone.
- When a trunk port configured with port security and assigned to an access VLAN for data traffic and to a voice VLAN for voice traffic, entering the **switchport voice** and **switchport priority extend** interface configuration commands has no effect.

  When a connected device uses the same MAC address to request an IP address for the access VLAN and then an IP address for the voice VLAN, only the access VLAN is assigned an IP address.

- When you enter a maximum secure address value for an interface, and the new value is greater than the previous value, the new value overwrites the previously configured value. If the new value is less than the previous value and the number of configured secure addresses on the interface exceeds the new value, the command is rejected.

- The switch does not support port security aging of sticky secure MAC addresses.

This table summarizes port security compatibility with other port-based features.

Table 145: Port Security Compatibility with Other Switch Features

Type of Port or Feature on Port	Compatible with Port Security
DTP [26] port [27]	No
Trunk port	Yes
Routed port	No
SPAN source port	Yes
SPAN destination port	No
EtherChannel	Yes
Tunneling port	Yes
Protected port	Yes
IEEE 802.1x port	Yes
Voice VLAN port [28]	Yes
IP source guard	Yes
Dynamic Address Resolution Protocol (ARP) inspection	Yes
Flex Links	Yes

26 DTP=Dynamic Trunking Protocol
27 A port configured with the **switchport mode dynamic** interface configuration command.
28 You must set the maximum allowed secure addresses on the port to two plus the maximum number of secure addresses allowed on the access VLAN.

How to Configure Port Security

Enabling and Configuring Port Security

Before You Begin

This task restricts input to an interface by limiting and identifying MAC addresses of the stations allowed to access the port:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode** {**access** | **trunk**}
4. **switchport voice vlan** *vlan-id*
5. **switchport port-security**
6. **switchport port-security [maximum** *value* [**vlan** {*vlan-list* | {**access** | **voice**}}]]
7. **switchport port-security violation** {**protect** | **restrict** | **shutdown** | **shutdown vlan**}
8. **switchport port-security [mac-address** *mac-address* [**vlan** {*vlan-id* | {**access** | **voice**}}]
9. **switchport port-security mac-address sticky**
10. **switchport port-security mac-address sticky** [*mac-address* | **vlan** {*vlan-id* | {**access** | **voice**}}]
11. **end**
12. **show port-security**
13. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Specifies the interface to be configured, and enter interface configuration mode.
Step 3	**switchport mode** {**access** | **trunk**} **Example:** `Switch(config-if)# switchport mode access`	Sets the interface switchport mode as access or trunk; an interface in the default mode (dynamic auto) cannot be configured as a secure port.
Step 4	**switchport voice vlan** *vlan-id* **Example:** `Switch(config-if)# switchport voice vlan 22`	Enables voice VLAN on a port. vlan-id—Specifies the VLAN to be used for voice traffic.

	Command or Action	Purpose
Step 5	**switchport port-security** **Example:** `Switch(config-if)# switchport port-security`	Enable port security on the interface.
Step 6	**switchport port-security [maximum** *value* [**vlan** {*vlan-list* | {**access** | **voice**}}]] **Example:** `Switch(config-if)# switchport port-security maximum 20`	(Optional) Sets the maximum number of secure MAC addresses for the interface. The maximum number of secure MAC addresses that you can configure on a switch or switch stack is set by the maximum number of available MAC addresses allowed in the system. This number is set by the active Switch Database Management (SDM) template. This number is the total of available MAC addresses, including those used for other Layer 2 functions and any other secure MAC addresses configured on interfaces. (Optional) **vlan—**sets a per-VLAN maximum value Enter one of these options after you enter the **vlan** keyword: • *vlan-list*—On a trunk port, you can set a per-VLAN maximum value on a range of VLANs separated by a hyphen or a series of VLANs separated by commas. For nonspecified VLANs, the per-VLAN maximum value is used. • **access—**On an access port, specifies the VLAN as an access VLAN. • **voice—**On an access port, specifies the VLAN as a voice VLAN. **Note** The **voice** keyword is available only if a voice VLAN is configured on a port and if that port is not the access VLAN. If an interface is configured for voice VLAN, configure a maximum of two secure MAC addresses.
Step 7	**switchport port-security violation** {**protect** | **restrict** | **shutdown** | **shutdown vlan**} **Example:** `Switch(config-if)# switchport port-security violation restrict`	(Optional) Sets the violation mode, the action to be taken when a security violation is detected, as one of these: • **protect—**When the number of port secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. You are not notified that a security violation has occurred. **Note** We do not recommend configuring the protect mode on a trunk port. The protect mode disables learning when any VLAN reaches its maximum limit, even if the port has not reached its maximum limit. • **restrict—**When the number of secure MAC addresses reaches the limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses or increase the number of maximum allowable addresses. An SNMP trap is sent, a syslog message is logged, and the violation counter increments.

	Command or Action	Purpose
		• **shutdown**—The interface is error-disabled when a violation occurs, and the port LED turns off. An SNMP trap is sent, a syslog message is logged, and the violation counter increments. • **shutdown vlan**—Use to set the security violation mode per VLAN. In this mode, the VLAN is error disabled instead of the entire port when a violation occurs. **Note** When a secure port is in the error-disabled state, you can bring it out of this state by entering the **errdisable recovery cause psecure-violation** global configuration command. You can manually re-enable it by entering the **shutdown** and **no shutdown** interface configuration commands or by using the **clear errdisable interface vlan** privileged EXEC command.
Step 8	**switchport port-security [mac-address** *mac-address* [**vlan** {*vlan-id* | {**access** | **voice**}}] **Example:** `Switch(config-if)# `**`switchport port-security mac-address 00:A0:C7:12:C9:25 vlan 3 voice`**	(Optional) Enters a secure MAC address for the interface. You can use this command to enter the maximum number of secure MAC addresses. If you configure fewer secure MAC addresses than the maximum, the remaining MAC addresses are dynamically learned. **Note** If you enable sticky learning after you enter this command, the secure addresses that were dynamically learned are converted to sticky secure MAC addresses and are added to the running configuration. (Optional) **vlan**—sets a per-VLAN maximum value. Enter one of these options after you enter the **vlan** keyword: • *vlan-id*—On a trunk port, you can specify the VLAN ID and the MAC address. If you do not specify a VLAN ID, the native VLAN is used. • **access**—On an access port, specifies the VLAN as an access VLAN. • **voice**—On an access port, specifies the VLAN as a voice VLAN. **Note** The **voice** keyword is available only if a voice VLAN is configured on a port and if that port is not the access VLAN. If an interface is configured for voice VLAN, configure a maximum of two secure MAC addresses.
Step 9	**switchport port-security mac-address sticky** **Example:** `Switch(config-if)# `**`switchport port-security mac-address sticky`**	(Optional) Enables sticky learning on the interface.
Step 10	**switchport port-security mac-address sticky** [*mac-address* | **vlan** {*vlan-id* | {**access** | **voice**}}] **Example:** `Switch(config-if)# `**`switchport port-security mac-address sticky`**	(Optional) Enters a sticky secure MAC address, repeating the command as many times as necessary. If you configure fewer secure MAC addresses than the maximum, the remaining MAC addresses are dynamically learned, are converted to sticky secure MAC addresses, and are added to the running configuration. **Note** If you do not enable sticky learning before this command is entered, an error message appears, and you cannot enter a sticky secure MAC address. (Optional) **vlan**—sets a per-VLAN maximum value.

	Command or Action	Purpose
	`00:A0:C7:12:C9:25 vlan voice`	Enter one of these options after you enter the **vlan** keyword: • *vlan-id*—On a trunk port, you can specify the VLAN ID and the MAC address. If you do not specify a VLAN ID, the native VLAN is used. • **access**—On an access port, specifies the VLAN as an access VLAN. • **voice**—On an access port, specifies the VLAN as a voice VLAN. **Note** The **voice** keyword is available only if a voice VLAN is configured on a port and if that port is not the access VLAN.
Step 11	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 12	**show port-security** **Example:** `Switch# show port-security`	Verifies your entries.
Step 13	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Port Security, on page 500
Configuration Examples for Port Security, on page 511
Port Security, on page 500
Configuration Examples for Port Security, on page 511

Enabling and Configuring Port Security Aging

Use this feature to remove and add devices on a secure port without manually deleting the existing secure MAC addresses and to still limit the number of secure addresses on a port. You can enable or disable the aging of secure addresses on a per-port basis.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport port-security aging** {**static** | **time** *time* | **type** {**absolute** | **inactivity**}}
4. **end**
5. **show port-security** [**interface** *interface-id*] [**address**]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Specifies the interface to be configured, and enter interface configuration mode.
Step 3	**switchport port-security aging** {**static** | **time** *time* | **type** {**absolute** | **inactivity**}} **Example:** `Switch(config-if)# switchport port-security aging time 120`	Enables or disable static aging for the secure port, or set the aging time or type. **Note** The switch does not support port security aging of sticky secure addresses. Enter **static** to enable aging for statically configured secure addresses on this port. For *time*, specifies the aging time for this port. The valid range is from 0 to 1440 minutes. For **type**, select one of these keywords: • **absolute—**Sets the aging type as absolute aging. All the secure addresses on this port age out exactly after the time (minutes) specified lapses and are removed from the secure address list. • **inactivity—**Sets the aging type as inactivity aging. The secure addresses on this port age out only if there is no data traffic from the secure source addresses for the specified time period.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	**show port-security** [**interface** *interface-id*] [**address**] **Example:** `Switch# show port-security interface gigabitethernet1/0/1`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Port Security Aging, on page 502

Port Security Aging, on page 502

Monitoring Port Security

This table displays port security information.

Table 146: Commands for Displaying Port Security Status and Configuration

Command	Purpose
show port-security [**interface** *interface-id*]	Displays port security settings for the switch or for the specified interface, including the maximum allowed number of secure MAC addresses for each interface, the number of secure MAC addresses on the interface, the number of security violations that have occurred, and the violation mode.
show port-security [**interface** *interface-id*] **address**	Displays all secure MAC addresses configured on all switch interfaces or on a specified interface with aging information for each address.
show port-security interface *interface-id* **vlan**	Displays the number of secure MAC addresses configured per VLAN on the specified interface.

Configuration Examples for Port Security

This example shows how to enable port security on a port and to set the maximum number of secure addresses to 50. The violation mode is the default, no static secure MAC addresses are configured, and sticky learning is enabled.

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security maximum 50
Switch(config-if)# switchport port-security mac-address sticky
```

This example shows how to configure a static secure MAC address on VLAN 3 on a port:

```
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security mac-address 0000.0200.0004 vlan 3
```

This example shows how to enable sticky port security on a port, to manually configure MAC addresses for data VLAN and voice VLAN, and to set the total maximum number of secure addresses to 20 (10 for data VLAN and 10 for voice VLAN).

```
Switch(config)# interface tengigabitethernet1/0/1
Switch(config-if)# switchport access vlan 21
Switch(config-if)# switchport mode access
Switch(config-if)# switchport voice vlan 22
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security maximum 20
Switch(config-if)# switchport port-security violation restrict
Switch(config-if)# switchport port-security mac-address sticky
Switch(config-if)# switchport port-security mac-address sticky 0000.0000.0002
Switch(config-if)# switchport port-security mac-address 0000.0000.0003
Switch(config-if)# switchport port-security mac-address sticky 0000.0000.0001 vlan voice
Switch(config-if)# switchport port-security mac-address 0000.0000.0004 vlan voice
Switch(config-if)# switchport port-security maximum 10 vlan access
Switch(config-if)# switchport port-security maximum 10 vlan voice
```

Related Topics

Port Security, on page 500
Enabling and Configuring Port Security, on page 504
Port Security, on page 500
Enabling and Configuring Port Security, on page 504

Information About Protocol Storm Protection

Protocol Storm Protection

When a switch is flooded with Address Resolution Protocol (ARP) or control packets, high CPU utilization can cause the CPU to overload. These issues can occur:

- Routing protocol can flap because the protocol control packets are not received, and neighboring adjacencies are dropped.

- Spanning Tree Protocol (STP) reconverges because the STP bridge protocol data unit (BPDU) cannot be sent or received.
- CLI is slow or unresponsive.

Using protocol storm protection, you can control the rate at which control packets are sent to the switch by specifying the upper threshold for the packet flow rate. The supported protocols are ARP, ARP snooping, Dynamic Host Configuration Protocol (DHCP) v4, DHCP snooping, Internet Group Management Protocol (IGMP), and IGMP snooping.

When the packet rate exceeds the defined threshold, the switch drops all traffic arriving on the specified virtual port for 30 seconds. The packet rate is measured again, and protocol storm protection is again applied if necessary.

For further protection, you can manually error disable the virtual port, blocking all incoming traffic on the virtual port. You can manually enable the virtual port or set a time interval for automatic re-enabling of the virtual port.

Note Excess packets are dropped on no more than two virtual ports.

Virtual port error disabling is not supported for EtherChannel and Flexlink interfaces

Default Protocol Storm Protection Configuration

Protocol storm protection is disabled by default. When it is enabled, auto-recovery of the virtual port is disabled by default.

How to Configure Protocol Storm Protection

Enabling Protocol Storm Protection

SUMMARY STEPS

1. **configure terminal**
2. **psp** {**arp** | **dhcp** | **igmp**} pps *value*
3. **errdisable detect cause psp**
4. **errdisable recovery interval** *time*
5. **end**
6. **show psp config** {**arp** | **dhcp** | **igmp**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**psp** {**arp** | **dhcp** | **igmp**} pps *value* **Example:** `Switch(config)# psp dhcp pps 35`	Configures protocol storm protection for ARP, IGMP, or DHCP. For *value*, specifies the threshold value for the number of packets per second. If the traffic exceeds this value, protocol storm protection is enforced. The range is from 5 to 50 packets per second.
Step 3	**errdisable detect cause psp** **Example:** `Switch(config)# errdisable detect cause psp`	(Optional) Enables error-disable detection for protocol storm protection. If this feature is enabled, the virtual port is error disabled. If this feature is disabled, the port drops excess packets without error disabling the port.
Step 4	**errdisable recovery interval** *time* **Example:** `Switch`	(Optional) Configures an auto-recovery time (in seconds) for error-disabled virtual ports. When a virtual port is error-disabled, the switch auto-recovers after this time. The range is from 30 to 86400 seconds.
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 6	**show psp config** {**arp** | **dhcp** | **igmp**} **Example:** `Switch# show psp config dhcp`	Verifies your entries.

Monitoring Protocol Storm Protection

Command	Purpose
show psp config {**arp** | **dhcp** | **igmp**}	Verify your entries.

CHAPTER 58

Configuring IPv6 First Hop Security

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for First Hop Security in IPv6

- You have configured the necessary IPv6 enabled SDM template.

- You should be familiar with the IPv6 neighbor discovery feature. For information, see the "Implementing IPv6 Addressing and Basic Connectivity" chapter of the Cisco IOS IPv6 Configuration Library on Cisco.com.

Restrictions for First Hop Security in IPv6

The following restrictions apply when applying FHS policies to EtherChannel interfaces (Port Channels):

- An FHS policy can be attached to a Layer 2 EtherChannel interface or to VLANs in an EtherChannel Group.
- An FHS policy cannot be attached to a Layer 3 EtherChannel interface.
- A physical port with an FHS policy attached cannot join an EtherChannel group.
- An FHS policy cannot be attached to an physical port when it is a member of an EtherChannel group.

Information about First Hop Security in IPv6

First Hop Security in IPv6 (FHS IPv6) is a set of IPv6 security features, the policies of which can be attached to a physical interface, an EtherChannel interface, or a VLAN. An IPv6 software policy database service stores and accesses these policies. When a policy is configured or modified, the attributes of the policy are stored or updated in the software policy database, then applied as was specified. The following IPv6 policies are currently supported:

- IPv6 Snooping Policy—IPv6 Snooping Policy acts as a container policy that enables most of the features available with FHS in IPv6.
- IPv6 Binding Table Content—A database table of IPv6 neighbors connected to the switch is created from information sources such as Neighbor Discovery (ND) protocol snooping. This database, or binding, table is used by various IPv6 guard features (such as IPv6 ND Inspection) to validate the link-layer address (LLA), the IPv4 or IPv6 address, and prefix binding of the neighbors to prevent spoofing and redirect attacks.
- IPv6 Neighbor Discovery Inspection—IPv6 ND inspection learns and secures bindings for stateless autoconfiguration addresses in L2 neighbor tables. IPv6 ND inspection analyzes neighbor discovery messages in order to build a trusted binding table database and IPv6 neighbor discovery messages that do not conform are dropped. An ND message is considered trustworthy if its IPv6-to-Media Access Control (MAC) mapping is verifiable.
- IPv6 Router Advertisement Guard—The IPv6 Router Advertisement (RA) guard feature enables the network administrator to block or reject unwanted or rogue RA guard messages that arrive at the network switch platform. RAs are used by routers to announce themselves on the link. The RA Guard feature analyzes the RAs and filters out bogus RAs sent by unauthorized routers. In host mode, all router advertisement and router redirect messages are disallowed on the port. The RA guard feature compares configuration information on the L2 device with the information found in the received RA frame. Once the L2 device has validated the content of the RA frame and router redirect frame against the configuration, it forwards the RA to its unicast or multicast destination. If the RA frame content is not validated, the RA is dropped.
- IPv6 DHCP Guard— The IPv6 DHCP Guard feature blocks reply and advertisement messages that come from unauthorized DHCPv6 servers and relay agents. IPv6 DHCP guard can prevent forged

messages from being entered in the binding table and block DHCPv6 server messages when they are received on ports that are not explicitly configured as facing a DHCPv6 server or DHCP relay. To use this feature, configure a policy and attach it to an interface or a VLAN. To debug DHCP guard packets, use the **debug ipv6 snooping dhcp-guard** privileged EXEC command.

How to Configure an IPv6 Snooping Policy

Beginning in privileged EXEC mode, follow these steps to configure IPv6 Snooping Policy :

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 snooping policy***policy-name*
3. {[**default**] | [**device-role** {**node** | **switch**}] | [**limit address-count** *value*] | [**no**] | [**protocol** {**dhcp** | **ndp**}] | [**security-level** {**glean** | **guard** | **inspect**}] | [**tracking** {**disable** [**stale-lifetime** [*seconds* | **infinite**] | **enable** [**reachable-lifetime** [*seconds* | **infinite**] }] | [**trusted-port**] }
4. **end**
5. **show ipv6 snooping policy** *policy-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ipv6 snooping policy***policy-name* **Example:** `Switch(config)# ipv6 snooping policy example_policy`	Creates a snooping policy and enters IPv6 Snooping Policy Configuration mode.
Step 3	{[**default**] | [**device-role** {**node** | **switch**}] | [**limit address-count** *value*] | [**no**] | [**protocol** {**dhcp** | **ndp**}] | [**security-level** {**glean** | **guard** | **inspect**}] | [**tracking** {**disable** [**stale-lifetime** [*seconds* | **infinite**] | **enable** [**reachable-lifetime** [*seconds* | **infinite**] }] | [**trusted-port**] } **Example:** Switch(config-ipv6-snooping)# security-level inspect **Example:** Switch(config-ipv6-snooping)# trusted-port	Enables data address gleaning, validates messages against various criteria, specifies the security level for messages. • (Optional) **default—**Sets all to default options. • (Optional) **device-role**{**node**] | **switch**}**—**Specifies the role of the device attached to the port. Default is **node**. • (Optional) **limit address-count** *value***—**Limits the number of addresses allowed per target. • (Optional) **no—**Negates a command or sets it to defaults. • (Optional) **protocol**{**dhcp** | **ndp**}**—**Specifies which protocol should be redirected to the snooping feature for analysis. The default, is **dhcp** and **ndp**. To change the default, use the **no protocol** command.

	Command or Action	Purpose
		• (Optional) **security-level**{**glean**|**guard**|**inspect**}**—**Specifies the level of security enforced by the feature. Default is **guard.** **glean—**Gleans addresses from messages and populates the binding table without any verification. **guard—**Gleans addresses and inspects messages. In addition, it rejects RA and DHCP server messages. This is the default option. **inspect—**Gleans addresses, validates messages for consistency and conformance, and enforces address ownership. • (Optional) **tracking** {**disable** | **enable**}**—**Overrides the default tracking behavior and specifies a tracking option. • (Optional) **trusted-port—**Sets up a trusted port. It disables the guard on applicable targets. Bindings learned through a trusted port have preference over bindings learned through any other port. A trusted port is given preference in case of a collision while making an entry in the table.
Step 4	**end** **Example:** `Switch(config-ipv6-snooping)# exit`	Exits configuration modes to Privileged EXEC mode.
Step 5	**show ipv6 snooping policy** *policy-name* **Example:** `Switch#show ipv6 snooping policy example_policy`	Displays the snooping policy configuration.

What to Do Next

Attach an IPv6 Snooping policy to interfaces or VLANs.

How to Attach an IPv6 Snooping Policy to an Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Snooping policy on an interface or VLAN:

SUMMARY STEPS

1. **configure terminal**
2. **interface** Interface_type *stack/module/port*
3. **switchport**
4. **ipv6 snooping** [**attach-policy** *policy_name* [**vlan** {*vlan_id* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids*}] | **vlan** {*vlan_id* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
5. **do show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** Interface_type *stack/module/port* **Example:** `Switch(config)# interface gigabitethernet 1/1/4`	Specifies an interface type and identifier; enters the interface configuration mode.
Step 3	**switchport** **Example:** `Switch(config-if)# switchport`	Enters the Switchport mode. **Note** To configure Layer 2 parameters, if the interface is in Layer 3 mode, you must enter the switchport interface configuration command without any parameters to put the interface into Layer 2 mode. This shuts down the interface and then re-enables it, which might generate messages on the device to which the interface is connected. When you put an interface that is in Layer 3 mode into Layer 2 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration. The command prompt displays as (config-if)# in Switchport configuration mode.

Step 4	**ipv6 snooping** [**attach-policy** *policy_name* [**vlan** {*vlan_id* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids*}] | **vlan** {*vlan_id* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
	
	Example:
	`Switch(config-if)# ipv6 snooping`
	
	`or`
	
	`Switch(config-if)# ipv6 snooping attach-policy example_policy`
	
	`or`
	`Switch(config-if)# ipv6 snooping vlan 111,112`
		Attaches a custom ipv6 snooping policy to the interface or the specified VLANs on the interface. To attach the default policy to the interface, use the **ipv6 snooping** command without the **attach-policy** keyword. To attach the default policy to VLANs on the interface, use the **ipv6 snooping vlan** command. The default policy is, security-level **guard**, device-role **node**, protocol **ndp** and **dhcp.**

	Command or Action	Purpose
	or Switch(config-if)# **ipv6 snooping attach-policy example_policy vlan 111,112**	
Step 5	**do show running-config** **Example:** Switch#(config-if)# **do show running-config**	Verifies that the policy is attached to the specified interface without exiting the interface configuration mode.

How to Attach an IPv6 Snooping Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Snooping policy on an EtherChannel interface or VLAN:

SUMMARY STEPS

1. **configure terminal**
2. **interface range** *Interface_name*
3. **ipv6 snooping** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_id*s | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config interface***portchannel_interface_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters the global configuration mode.
Step 2	**interface range** *Interface_name* **Example:** Switch(config)# **interface range Po11**	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. **Tip** Enter the **do show interfaces summary** command for quick reference to interface names and types.
Step 3	**ipv6 snooping** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_id*s | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]	Attaches the IPv6 Snooping policy to the interface or the specified VLANs on that interface. The default policy is attached if the **attach-policy** option is not used.

	Command or Action	Purpose

	Example:
	`Switch(config-if-range)# ipv6 snooping attach-policy example_policy`
	or
	`Switch(config-if-range)# ipv6 snooping attach-policy example_policy vlan 222,223,224`
	or
	`Switch(config-if-range)#ipv6 snooping vlan 222, 223,224`	

Step 4	**do show running-config interface***portchannel_interface_name* **Example:** `Switch#(config-if-range)#  do show running-config int po11`	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Snooping Policy to VLANs Globally

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Snooping Policy to VLANs across multiple interfaces:

SUMMARY STEPS

1. **configure terminal**
2. **vlan configuration** *vlan_list*
3. **ipv6 snooping** [**attach-policy** *policy_name*]
4. **do show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**vlan configuration** *vlan_list* **Example:** `Switch(config)#  vlan configuration 333`	Specifies the VLANs to which the IPv6 Snooping policy will be attached ; enters the VLAN interface configuration mode.

	Command or Action	Purpose
Step 3	**ipv6 snooping** [**attach-policy** *policy_name*] **Example:** `Switch(config-vlan-config)#`**`ipv6 snooping attach-policy example_policy`**	Attaches the IPv6 Snooping policy to the specified VLANs across all switch and stack interfaces. The default policy is attached if the **attach-policy** option is not used. The default policy is, security-level **guard**, device-role **node**, protocol **ndp** and **dhcp.**
Step 4	**do show running-config** **Example:** `Switch#(config-if)#` **`do show running-config`**	Verifies that the policy is attached to the specified VLANs without exiting the interface configuration mode.

How to Configure the IPv6 Binding Table Content

Beginning in privileged EXEC mode, follow these steps to configure IPv6 Binding Table Content :

SUMMARY STEPS

1. **configure terminal**
2. [**no**] **ipv6 neighbor binding** [**vlan** *vlan-id* {*ipv6-address* **interface** interface_type *stack/module/port hw_address* [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] | [**tracking**{ [default | disable] [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] | [**enable** [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] | [**retry-interval** {*seconds*| **default** [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] }]
3. [**no**] **ipv6 neighbor binding max-entries** *number* [**mac-limit** *number* | **port-limit** *number* [**mac-limit** *number*] | **vlan-limit** *number* [[**mac-limit** *number*] | [**port-limit** *number* [**mac-limit***number*]]]]
4. **ipv6 neighbor binding logging**
5. **exit**
6. **show ipv6 neighbor binding**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch#` **`configure terminal`**	Enters the global configuration mode.
Step 2	[**no**] **ipv6 neighbor binding** [**vlan** *vlan-id* {*ipv6-address* **interface** interface_type *stack/module/port hw_address* [**reachable-lifetimevalue** [*seconds*	**default**	**infinite**]	[**tracking**{ [default	disable] [**reachable-lifetimevalue** [*seconds*	**default**	**infinite**]	[**enable** [**reachable-lifetimevalue** [*seconds*	**default**	**infinite**]	[**retry-interval**	

	Command or Action	Purpose
	{*seconds*| **default** [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] }] **Example:** `Switch(config)#  ipv6 neighbor binding`	
Step 3	[**no**] **ipv6 neighbor binding max-entries** *number* [**mac-limit** *number* | **port-limit** *number* [**mac-limit** *number*] | **vlan-limit** *number* [[**mac-limit** *number*] | [**port-limit** *number* [**mac-limit***number*]]]] **Example:** `Switch(config)#  ipv6 neighbor binding max-entries 30000`	Specifies the maximum number of entries that are allowed to be inserted in the binding table cache.
Step 4	**ipv6 neighbor binding logging** **Example:** `Switch(config)# ipv6 neighbor binding logging`	Enables the logging of binding table main events.
Step 5	**exit** **Example:** `Switch(config)# exit`	Exits global configuration mode, and places the router in privileged EXEC mode.
Step 6	**show ipv6 neighbor binding** **Example:** `Switch#  show ipv6 neighbor binding`	Displays contents of a binding table.

How to Configure an IPv6 Neighbor Discovery Inspection Policy

Beginning in privileged EXEC mode, follow these steps to configure an IPv6 ND Inspection Policy:

SUMMARY STEPS

1. **configure terminal**
2. [**no**]**ipv6 nd inspection policy** *policy-name*
3. **device-role** {**host** | **monitor** | **router** | **switch**}
4. **drop-unsecure**
5. **limit address-count** *value*
6. **sec-level minimum** *value*
7. **tracking** {**enable** [**reachable-lifetime** {*value* | **infinite**}] | **disable** [**stale-lifetime** {*value* | **infinite**}]}
8. **trusted-port**
9. **validate source-mac**
10. **no** {**device-role** | **drop-unsecure** | **limit address-count** | **sec-level minimum** | **tracking** | **trusted-port** | **validate source-mac**}
11. **default** {**device-role** | **drop-unsecure** | **limit address-count** | **sec-level minimum** | **tracking** | **trusted-port** | **validate source-mac**}
12. **do show ipv6 nd inspection policy** *policy_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	[**no**]**ipv6 nd inspection policy** *policy-name* **Example:** `Switch(config)# ipv6 nd inspection policy example_policy`	Specifies the ND inspection policy name and enters ND Inspection Policy configuration mode.
Step 3	**device-role** {**host** | **monitor** | **router** | **switch**} **Example:** `Switch(config-nd-inspection)# device-role switch`	Specifies the role of the device attached to the port. The default is **host**.
Step 4	**drop-unsecure** **Example:** `Switch(config-nd-inspection)# drop-unsecure`	Drops messages with no or invalid options or an invalid signature.
Step 5	**limit address-count** *value* **Example:** `Switch(config-nd-inspection)# limit address-count 1000`	Enter 1–10,000.
Step 6	**sec-level minimum** *value* **Example:** `Switch(config-nd-inspection)# limit address-count 1000`	Specifies the minimum security level parameter value when Cryptographically Generated Address (CGA) options are used.

	Command or Action	Purpose
Step 7	**tracking** {**enable** [**reachable-lifetime** {*value* | **infinite**}] | **disable** [**stale-lifetime** {*value* | **infinite**}]} **Example:** `Switch(config-nd-inspection)# tracking disable stale-lifetime infinite`	Overrides the default tracking policy on a port.
Step 8	**trusted-port** **Example:** `Switch(config-nd-inspection)# trusted-port`	Configures a port to become a trusted port.
Step 9	**validate source-mac** **Example:** `Switch(config-nd-inspection)# validate source-mac`	
Step 10	**no** {**device-role** | **drop-unsecure** | **limit address-count** | **sec-level minimum** | **tracking** | **trusted-port** | **validate source-mac**} **Example:** `Switch(config-nd-inspection)# no validate source-mac`	Remove the current configuration of a parameter with the **no** form of the command.
Step 11	**default** {**device-role** | **drop-unsecure** | **limit address-count** | **sec-level minimum** | **tracking** | **trusted-port** | **validate source-mac**} **Example:** `Switch(config-nd-inspection)# default limit address-count`	Restores configuration to the default values.
Step 12	**do show ipv6 nd inspection policy** *policy_name* **Example:** `Switch(config-nd-inspection)# do show ipv6 nd inspection policy example_policy`	Verifies the ND Inspection Configuration without exiting ND inspection configuration mode.

How to Attach an IPv6 Neighbor Discovery Inspection Policy to an Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 ND Inspection policy to an interface or VLANs on an interface :

SUMMARY STEPS

1. **configure terminal**
2. **interface** Interface_type *stack/module/port*
3. **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** Interface_type *stack/module/port* **Example:** `Switch(config)#  interface gigabitethernet 1/1/4`	Specifies an interface type and identifier; enters the interface configuration mode.

Step 3	**ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
	
	Example:
	`Switch(config-if)# ipv6 nd inspection attach-policy example_policy`
	
	`or`
	
	`Switch(config-if)# ipv6 nd inspection attach-policy example_policy vlan 222,223,224`
	
	`or`
	
	`Switch(config-if)# ipv6 nd inspection vlan 222, 223,224`
		Attaches the Neighbor Discovery Inspection policy to the interface or the specified VLANs on that interface. The default policy is attached if the **attach-policy** option is not used.

Step 4	**do show running-config** **Example:** `Switch#(config-if)#  do show running-config`	Verifies that the policy is attached to the specified interface without exiting the interface configuration mode.

How to Attach an IPv6 Neighbor Discovery Inspection Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Neighbor Discovery Inspection policy on an EtherChannel interface or VLAN:

SUMMARY STEPS

1. **configure terminal**
2. **interface range** *Interface_name*
3. **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_id*s | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config interface***portchannel_interface_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface range** *Interface_name* **Example:** `Switch(config)#  interface range Po11`	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. **Tip** Enter the **do show interfaces summary** command for quick reference to interface names and types.
Step 3	**ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids*	**add** *vlan_ids*	**except** *vlan_ids*	**none**	**remove** *vlan_ids*	**all**}]	**vlan** [{*vlan_ids*	**add** *vlan_ids*	**except***vlan_ids*	**none**	**remove** *vlan_ids*	**all**}] **Example:** `Switch(config-if-range)# ipv6 nd inspection attach-policy example_policy` `or` `Switch(config-if-range)# ipv6 nd inspection attach-policy example_policy vlan 222,223,224` `or` `Switch(config-if-range)#ipv6 nd inspection vlan 222, 223,224`	Attaches the ND Inspection policy to the interface or the specified VLANs on that interface. The default policy is attached if the **attach-policy** option is not used.

	Command or Action	Purpose
Step 4	**do show running-config interface***portchannel_interface_name* **Example:** `Switch#(config-if-range)# ` **`do show running-config int po11`**	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Neighbor Discovery Inspection Policy to VLANs Globally

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 ND Inspection policy to VLANs across multiple interfaces:

SUMMARY STEPS

1. **configure terminal**
2. **vlan configuration** *vlan_list*
3. **ipv6 nd inspection** [**attach-policy** *policy_name*]
4. **do show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch#` **`configure terminal`**	Enters the global configuration mode.
Step 2	**vlan configuration** *vlan_list* **Example:** `Switch(config)#` **`vlan configuration 334`**	Specifies the VLANs to which the IPv6 Snooping policy will be attached ; enters the VLAN interface configuration mode.
Step 3	**ipv6 nd inspection** [**attach-policy** *policy_name*] **Example:** `Switch(config-vlan-config)#`**`ipv6 nd inspection attach-policy example_policy`**	Attaches the IPv6 Neighbor Discovery policy to the specified VLANs across all switch and stack interfaces. The default policy is attached if the **attach-policy** option is not used. The default policy is, device-role **host**, no drop-unsecure, limit address-count disabled, sec-level minimum is disabled, tracking is disabled, no trusted-port, no validate source-mac.
Step 4	**do show running-config** **Example:** `Switch#(config-if)# ` **`do show running-config`**	Confirms that the policy is attached to the specified VLANs without exiting the configuration mode.

How to Configure an IPv6 Router Advertisement Guard Policy

Beginning in privileged EXEC mode, follow these steps to configure an IPv6 Router Advertisement policy :

SUMMARY STEPS

1. **configure terminal**
2. [**no**]**ipv6 nd raguard policy** *policy-name*
3. [**no**]**device-role** {**host** | **monitor** | **router** | **switch**}
4. [**no**]**hop-limit** {**maximum** | **minimum**} *value*
5. [**no**]**managed-config-flag** {**off** | **on**}
6. [**no**]**match** {**ipv6 access-list** *list* | **ra prefix-list** *list*}
7. [**no**]**other-config-flag** {**on** | **off**}
8. [**no**]**router-preference maximum** {**high** | **medium** | **low**}
9. [**no**]**trusted-port**
10. **default** {**device-role** | **hop-limit** {**maximum** | **minimum**} | **managed-config-flag** | **match** {**ipv6 access-list** | **ra prefix-list** } | **other-config-flag** | **router-preference maximum**| **trusted-port**}
11. **do show ipv6 nd raguard policy** *policy_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	[**no**]**ipv6 nd raguard policy** *policy-name* **Example:** `Switch(config)# ipv6 nd raguard policy example_policy`	Specifies the RA Guard policy name and enters RA Guard Policy configuration mode.
Step 3	[**no**]**device-role** {**host**	**monitor**	**router**	**switch**} **Example:** `Switch(config-nd-raguard)# device-role switch`	Specifies the role of the device attached to the port. The default is **host**.
Step 4	[**no**]**hop-limit** {**maximum**	**minimum**} *value* **Example:** `Switch(config-nd-raguard)# hop-limit maximum 33`	(1–255) Range for Maximum and Minimum Hop Limit values. Enables filtering of Router Advertisement messages by the Hop Limit value. A rogue RA message may have a low Hop Limit value (equivalent to the IPv4 Time to Live) that when accepted by the host,

	Command or Action	Purpose
		prevents the host from generating traffic to destinations beyond the rogue RA message generator. An RA message with an unspecified Hop Limit value is blocked. If not configured, this filter is disabled. Configure **minimum** to block RA messages with Hop Limit values lower than the value you specify. Configure **maximum**to block RA messages with Hop Limit values greater than the value you specify.
Step 5	[**no**]**managed-config-flag** {**off** | **on**} **Example:** `Switch(config-nd-raguard)#` **`managed-config-flag on`**	Enables filtering of Router Advertisement messages by the Managed Address Configuration, or "M" flag field. A rouge RA message with an M field of 1 can cause a host to use a rogue DHCPv6 server. If not configured, this filter is disabled. **On—**Accepts and forwards RA messages with an M value of 1, blocks those with 0. **Off—**Accepts and forwards RA messages with an M value of 0, blocks those with 1.
Step 6	[**no**]**match** {**ipv6 access-list** *list* | **ra prefix-list** *list*} **Example:** `Switch(config-nd-raguard)#` **`match ipv6 access-list example_list`**	Matches a specified prefix list or access list.
Step 7	[**no**]**other-config-flag** {**on** | **off**} **Example:** `Switch(config-nd-raguard)#` **`other-config-flag on`**	Enables filtering of Router Advertisement messages by the Other Configuration, or "O" flag field. A rouge RA message with an O field of 1 can cause a host to use a rogue DHCPv6 server. If not configured, this filter is disabled. **On—**Accepts and forwards RA messages with an O value of 1, blocks those with 0. **Off—**Accepts and forwards RA messages with an O value of 0, blocks those with 1.
Step 8	[**no**]**router-preference maximum** {**high** | **medium** | **low**} **Example:** `Switch(config-nd-raguard)#` **`router-preference maximum high`**	Enables filtering of Router Advertisement messages by the Router Preference flag. If not configured, this filter is disabled. • **high—**Accepts RA messages with the Router Preference set to high, medium, or low. • **medium—**Blocks RA messages with the Router Preference set to high. • **low—**Blocks RA messages with the Router Preference set to medium and high.
Step 9	[**no**]**trusted-port** **Example:** `Switch(config-nd-raguard)#` **`trusted-port`**	When configured as a trusted port, all attached devices are trusted, and no further message verification is performed.

	Command or Action	Purpose
Step 10	**default** {**device-role** | **hop-limit** {**maximum** | **minimum**} | **managed-config-flag** | **match** {**ipv6 access-list** | **ra prefix-list** } | **other-config-flag** | **router-preference maximum**| **trusted-port**} **Example:** `Switch(config-nd-raguard)# default hop-limit`	Restores a command to its default value.
Step 11	**do show ipv6 nd raguard policy** *policy_name* **Example:** `Switch(config-nd-raguard)# do show ipv6 nd raguard policy example_policy`	(Optional)—Displays the ND Guard Policy configuration without exiting the RA Guard policy configuration mode.

How to Attach an IPv6 Router Advertisement Guard Policy to an Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Router Advertisement policy to an interface or to VLANs on the interface :

SUMMARY STEPS

1. **configure terminal**
2. **interface** Interface_type *stack/module/port*
3. **ipv6 nd raguard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** Interface_type *stack/module/port* **Example:** `Switch(config)#  interface gigabitethernet 1/1/4`	Specifies an interface type and identifier; enters the interface configuration mode.
Step 3	**ipv6 nd raguard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]	Attaches the Neighbor Discovery Inspection policy to the interface or the specified VLANs on that

	Command or Action	Purpose
	| **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] **Example:** `Switch(config-if)#` **`ipv6 nd raguard attach-policy example_policy`** `or` `Switch(config-if)#` **`ipv6 nd raguard attach-policy example_policy vlan 222,223,224`** **`or`** `Switch(config-if)#` **`ipv6 nd raguard vlan 222, 223,224`**	interface. The default policy is attached if the **attach-policy** option is not used.
Step 4	**do show running-config** **Example:** `Switch#(config-if)#` **`do show running-config`**	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Router Advertisement Guard Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Router Advertisement Guard Policy on an EtherChannel interface or VLAN:

SUMMARY STEPS

1. **configure terminal**
2. **interface range** *Interface_name*
3. **ipv6 nd raguard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* \| **add** *vlan_ids* \| **except** *vlan_ids* \| **none** \| **remove** *vlan_ids* \| **all**}] \| **vlan** [{*vlan_ids* \| **add** *vlan_ids* \| **except***vlan_ids* \| **none** \| **remove** *vlan_ids* \| **all**}]
4. **do show running-config interface***portchannel_interface_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch#` **`configure terminal`**	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	**interface range** *Interface_name* **Example:** `Switch(config)#  interface range Po11`	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. **Tip** Enter the **do show interfaces summary** command for quick reference to interface names and types.

Step 3	**ipv6 nd raguard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_id*s | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
	
	Example:
	`Switch(config-if-range)# ipv6 nd raguard attach-policy example_policy`
	
	`or`
	
	`Switch(config-if-range)# ipv6 nd raguard attach-policy example_policy vlan 222,223,224`
	
	`or`
	
	`Switch(config-if-range)#ipv6 nd raguard vlan 222, 223,224`
		Attaches the RA Guard policy to the interface or the specified VLANs on that interface. The default policy is attached if the **attach-policy** option is not used.

Step 4	**do show running-config interface***portchannel_interface_name* **Example:** `Switch#(config-if-range)#  do show running-config int po11`	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Router Advertisement Guard Policy to VLANs Globally

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Router Advertisement policy to VLANs regardless of interface:

SUMMARY STEPS

1. **configure terminal**
2. **vlan configuration** *vlan_list*
3. **ipv6 dhcp guard** [**attach-policy** *policy_name*]
4. **do show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**vlan configuration** *vlan_list* **Example:** `Switch(config)# vlan configuration 335`	Specifies the VLANs to which the IPv6 RA Guard policy will be attached ; enters the VLAN interface configuration mode.
Step 3	**ipv6 dhcp guard** [**attach-policy** *policy_name*] **Example:** `Switch(config-vlan-config)#ipv6 nd raguard attach-policy example_policy`	Attaches the IPv6 RA Guard policy to the specified VLANs across all switch and stack interfaces. The default policy is attached if the **attach-policy** option is not used.
Step 4	**do show running-config** **Example:** `Switch#(config-if)#  do show running-config`	Confirms that the policy is attached to the specified VLANs without exiting the configuration mode.

How to Configure an IPv6 DHCP Guard Policy

Beginning in privileged EXEC mode, follow these steps to configure an IPv6 DHCP (DHCPv6) Guard policy:

SUMMARY STEPS

1. **configure terminal**
2. [**no**]**ipv6 dhcp guard policy** *policy-name*
3. [**no**]**device-role** {**client** | **server**}
4. [**no**] **match server access-list** *ipv6-access-list-name*
5. [**no**] **match reply prefix-list** *ipv6-prefix-list-name*
6. [**no**]**preference**{ **max** *limit* | **min** *limit* }
7. [**no**] **trusted-port**
8. **default** {**device-role** | **trusted-port**}
9. **do show ipv6 dhcp guard policy** *policy_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	[**no**]**ipv6 dhcp guard policy** *policy-name* **Example:** `Switch(config)# ipv6 dhcp guard policy example_policy`	Specifies the DHCPv6 Guard policy name and enters DHCPv6 Guard Policy configuration mode.
Step 3	[**no**]**device-role** {**client** | **server**} **Example:** `Switch(config-dhcp-guard)# device-role server`	(Optional) Filters out DHCPv6 replies and DHCPv6 advertisements on the port that are not from a device of the specified role. Default is **client**. • **client—**Default value, specifies that the attached device is a client. Server messages are dropped on this port. • **server—**Specifies that the attached device is a DHCPv6 server. Server messages are allowed on this port.

Step 4	[**no**] **match server access-list** *ipv6-access-list-name*
	
	Example:
	`;;Assume a preconfigured IPv6 Access List as follows:`
	`Switch(config)# ipv6 access-list my_acls`
	`Switch(config-ipv6-acl)# permit host FE80::A8BB:CCFF:FE01:F700 any`
	
	`;;configure DCHPv6 Guard to match approved access list.`
	`Switch(config-dhcp-guard)#  match server access-list my_acls`
		(Optional). Enables verification that the advertised DHCPv6 server or relay address is from an authorized server access list (The destination address in the access list is 'any'). If not configured, this check will be bypassed. An empty access list is treated as a permit all.

Step 5	[**no**] **match reply prefix-list** *ipv6-prefix-list-name*
	
	Example:
	`;;Assume a preconfigured IPv6 prefix list as follows:`
	`Switch(config)# ipv6 prefix-list my_prefix permit 2001:0DB8::/64 le 128`
	
	`;; Configure DCHPv6 Guard to match prefix`
	`Switch(config-dhcp-guard)#  match reply prefix-list my_prefix`
		(Optional) Enables verification of the advertised prefixes in DHCPv6 reply messages from the configured authorized prefix list. If not configured, this check will be bypassed. An empty prefix list is treated as a permit.

Step 6	[**no**]**preference**{ **max** *limit* | **min** *limit* }
	
	Example:
	`Switch(config-dhcp-guard)# preference max 250`
	`Switch(config-dhcp-guard)#preference min 150`
		Configure **max** and **min** when **device-role** is **server**to filter DCHPv6 server advertisements by the server preference value. The defaults permit all advertisements.

	Command or Action	Purpose
		max *limit*—(0 to 255) (Optional) Enables verification that the advertised preference (in preference option) is less than the specified limit. Default is 255. If not specified, this check will be bypassed. **min** *limit*—(0 to 255) (Optional) Enables verification that the advertised preference (in preference option) is greater than the specified limit. Default is 0. If not specified, this check will be bypassed.
Step 7	[**no**] **trusted-port** **Example:** `Switch(config-dhcp-guard)#` **`trusted-port`**	(Optional) **trusted-port**—Sets the port to a trusted mode. No further policing takes place on the port. **Note** If you configure a trusted port then the device-role option is not available.
Step 8	**default** {**device-role** | **trusted-port**} **Example:** `Switch(config-dhcp-guard)#` **`default device-role`**	(Optional) **default**—Sets a command to its defaults.
Step 9	**do show ipv6 dhcp guard policy** *policy_name* **Example:** `Switch(config-dhcp-guard)#` **`do show ipv6 dhcp guard policy example_policy`**	(Optional) Displays the configuration of the IPv6 DHCP guard policy without leaving the configuration submode. Omitting the *policy_name* variable displays all DHCPv6 policies.

Example of DHCPv6 Guard Configuration

```
enable
configure terminal
ipv6 access-list acl1
 permit host FE80::A8BB:CCFF:FE01:F700 any
ipv6 prefix-list abc permit 2001:0DB8::/64 le 128
ipv6 dhcp guard policy pol1
 device-role server
 match server access-list acl1
 match reply prefix-list abc
 preference min 0
 preference max 255
 trusted-port
interface GigabitEthernet 0/2/0
 switchport
 ipv6 dhcp guard attach-policy pol1 vlan add 1
 vlan 1
  ipv6 dhcp guard attach-policy pol1
show ipv6 dhcp guard policy pol1
```

How to Attach an IPv6 DHCP Guard Policy to an Interface or a VLAN on an Interface

Beginning in privileged EXEC mode, follow these steps to configure IPv6 Binding Table Content :

SUMMARY STEPS

1. **configure terminal**
2. **interface** Interface_type *stack/module/port*
3. **ipv6 dhcp guard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config interface** Interface_type *stack/module/port*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** Interface_type *stack/module/port* **Example:** `Switch(config)#  interface gigabitethernet 1/1/4`	Specifies an interface type and identifier; enters the interface configuration mode.

Step 3	**ipv6 dhcp guard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
	
	Example:
	`Switch(config-if)# ipv6 dhcp guard attach-policy example_policy`
	
	`or`
	
	`Switch(config-if)# ipv6 dhcp guard attach-policy example_policy vlan 222,223,224`
	
	`or`
	
	`Switch(config-if)# ipv6 dhcp guard vlan 222, 223,224`
		Attaches the DHCP Guard policy to the interface or the specified VLANs on that interface. The default policy is attached if the **attach-policy** option is not used.

Step 4	**do show running-config interface** Interface_type *stack/module/port* **Example:** `Switch#(config-if)#  do show running-config gig 1/1/4`	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 DHCP Guard Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 DHCP Guard policy on an EtherChannel interface or VLAN:

SUMMARY STEPS

1. **configure terminal**
2. **interface range** *Interface_name*
3. **ipv6 dhcp guard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_id*s | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config interface***portchannel_interface_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface range** *Interface_name* **Example:** `Switch(config)#  interface range Po11`	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. **Tip** Enter the **do show interfaces summary** command for quick reference to interface names and types.

Step 3	**ipv6 dhcp guard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_id*s | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
	
	Example:
	`Switch(config-if-range)# ipv6 dhcp guard attach-policy example_policy`
	
	`or`
	
	`Switch(config-if-range)# ipv6 dhcp guard attach-policy example_policy vlan 222,223,224`
	
	`or`
	
	`Switch(config-if-range)#ipv6 dhcp guard vlan 222, 223,224`
		Attaches the DHCP Guard policy to the interface or the specified VLANs on that interface. The default policy is attached if the **attach-policy** option is not used.

Step 4	**do show running-config interface***portchannel_interface_name* **Example:** `Switch#(config-if-range)#  do show running-config int po11`	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 DHCP Guard Policy to VLANs Globally

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 DHCP Guard policy to VLANs across multiple interfaces:

SUMMARY STEPS

1. **configure terminal**
2. **vlan configuration** *vlan_list*
3. **ipv6 dhcp guard** [**attach-policy** *policy_name*]
4. **do show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**vlan configuration** *vlan_list* **Example:** `Switch(config)# vlan configuration 334`	Specifies the VLANs to which the IPv6 Snooping policy will be attached ; enters the VLAN interface configuration mode.
Step 3	**ipv6 dhcp guard** [**attach-policy** *policy_name*] **Example:** `Switch(config-vlan-config)#ipv6 dhcp guard attach-policy example_policy`	Attaches the IPv6 Neighbor Discovery policy to the specified VLANs across all switch and stack interfaces. The default policy is attached if the **attach-policy** option is not used. The default policy is, device-role **client**, **no** trusted-port.
Step 4	**do show running-config** **Example:** `Switch#(config-if)#  do show running-config`	Confirms that the policy is attached to the specified VLANs without exiting the configuration mode.

Additional References

Related Documents

Related Topic	Document Title
IPv6 network management and security topics	IPv6 Configuration Library, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/en/US/docs/ios-xml/ios/ipv6/config_library/xe-3se/3850/ipv6-xe-3se-3850-library.html
IPv6 Command Reference	IPv6 Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/en/US/docs/ios-xml/ios/ipv6/command/ipv6-xe-3se-3850-cr-book.html

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

CHAPTER 59

Configuring Wireless Guest Access

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for Guest Access

- All mobility peers should be configured for hierarchical mobility architecture.
- For Guest Controller Mobility Anchor configuration on WLAN is must on Mobility Agent and Guest Controller.

- Guest Access can be a 3 box solution or 2 box solution. The mobility tunnel link status should be up between:
    - Mobility Agent, Mobility Controller and Guest Controller.

  or

    - Mobility Agent/Mobility Controller and Guest Controller

Restrictions for Guess Access

Guest Controller functionality is not supported on the Catalyst 3850 switch whereas Catalyst 3850 can act as mobility agent.

Information about Wireless Guest Access

Ideally, the implementation of a wireless guest network uses as much of an enterprise's existing wireless and wired infrastructure as possible to avoid the cost and complexity of building a physical overlay network. Assuming this is the case, the following additional elements and functions are needed:

- A dedicated guest WLAN/SSID—Implemented throughout the campus wireless network wherever guest access is required. A guest WLAN is identified by a WLAN with mobility anchor (Guest Controller) configured.
- Guest traffic segregation—Requires implementing Layer 2 or Layer 3 techniques across the campus network to restrict where guests are allowed to go.
- Access control—Involves using imbedded access control functionality within the campus network or implementing an external platform to control guest access to the Internet from the enterprise network.
- Guest user credential management—A process by which a sponsor or lobby administrator can create temporary credentials in behalf of a guest. This function might be resident within an access control platform or it might be a component of AAA or some other management system.

Fast Secure Roaming

Fast secure roaming can be achieved by caching the Pairwise Master Key (PMK) information for Cisco Centralized Key Management (CCKM), 802.11r and 802.11i clients. Cisco Centralized Key Management (CCKM) helps to improve roaming. Only the client can initiate the roaming process, which depends on factors such as:

- Overlap between APs
- Distance between APs
- Channel, signal strength, and load on the AP
- Data rates and output power

Whenever a fast-roaming client 802.11i, [CCKM]) roams to a new device, after fast-roaming the clients go through mobility "handoff" procedure. And new AAA attributes learned through mobility "handoff" procedure get re-applied.

Full L2 authentication must be avoided during roaming if the client uses the 802.11i WPA2, CCKM, 802.11r to achieve the full requirements of fast secure roaming. The PMK cache (802.11i, CCKM, and 802.11r) is used to authenticate and derive the keys for roaming clients to avoid full L2 authentication. This requires all Mobility Anchors (MA) and Mobility Controllers (MC) in the mobility group to have the same PMK cache values.

The session timeout defines when a PMK cache will expire. A PMK cache can also be deleted when a client fails to re-authenticate or when it is manually deleted them from the CLI. The deletion on the original controller or switch shall be propagated to other controllers or switches in the same mobility group.

How to Configure Guest Access

Creating a Lobby Administrator Account

SUMMARY STEPS

1. **configure terminal**
2. **user-name** *user-name*
3. **type lobby-admin**
4. **password 0** *password*
5. **end**
6. **show running-config | section** *user-name* (or) **show running-config | section** *configured lobby admin username*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch # configure terminal`	Enters global configuration mode.
Step 2	**user-name** *user-name* **Example:** `Switch (config)# user-name lobby`	Creates a user account.
Step 3	**type lobby-admin** **Example:** `Switch (config-user-name)# type lobby-admin`	Specifies the account type as lobby admin.

	Command or Action	Purpose
Step 4	**password 0** *password* **Example:** `Switch(config-user-name)# password 0 lobby`	Creates a password for the lobby administrator account.
Step 5	**end** **Example:** `Switch (config-user-name)# end`	Returns to privileged EXEC mode.
Step 6	**show running-config | section** *user-name* (or) **show running-config | section** *configured lobby admin username* **Example:** `Switch # show running-config | section lobby`	Displays the configuration details.

Configuring Guest User Accounts

SUMMARY STEPS

1. **configure terminal**
2. **user-name** *user-name*
3. **password** *unencrypted/hidden-password password*
4. **type network-user description** *description* **guest-user lifetime year** *0-1* **month** *0-11* **day** *0-30* **hour** *0-23* **minute** *0-59* **second** *0-59*
5. **end**
6. **show aaa local netuser all**
7. **show running-config | section***user-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch # configure terminal`	Enters global configuration mode.
Step 2	**user-name** *user-name* **Example:** `Switch (config)# user-name guest`	Creates a username for the lobby ambassador account.
Step 3	**password** *unencrypted/hidden-password password*	Specifies the password for the user.

	Command or Action	Purpose
	Example: `Switch (config-user-name)# password 0 guest`	
Step 4	**type network-user description** *description* **guest-user lifetime year** *0-1* **month** *0-11* **day** *0-30* **hour** *0-23* **minute** *0-59* **second** *0-59* **Example:** `Switch (config-user-name)# type network-user description guest guest-user lifetime year 1 month 10 day 3 hour 1 minute 5 second 30`	Specifies the type of user.
Step 5	**end** **Example:** `Switch (config-user-name)# end`	Returns to privileged EXEC mode.
Step 6	**show aaa local netuser all** **Example:** `Switch # show aaa local netuser all`	Displays the configuration details. After the lifetime, the user-name with guest type will be deleted and the client associated with the guest user-name will be de-authenticated.
Step 7	**show running-config | section***user-name* **Example:** `Switch # show running-config | section guest`	Displays the configuration details.

Configuring Mobility Agent (MA)

SUMMARY STEPS

1. **configure terminal**
2. **wireless mobility controller ip***mc-ipaddress* **public-ip** *mc-publicipaddress*
3. **wlan** *wlan-name wlan-id ssid*
4. **client vlan id***vlan-group name/vlan-id*
5. **no security wpa**
6. **mobility anchor** *ipaddress*
7. **aaa-override**
8. **no shutdown**
9. **end**
10. **show wireless mobility summary**
11. **show wlan name** *wlan-name/id*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch # configure terminal`	Enters global configuration mode.
Step 2	**wireless mobility controller ip***mc-ipaddress* **public-ip** *mc-publicipaddress* **Example:** `Switch (config) # wireless mobility controller ip27.0.0.1 public-ip 27.0.0.1`	Configures the Mobility Controller to which the MA will be associated.
Step 3	**wlan** *wlan-name wlan-id ssid* **Example:** `Switch (config) # wlan mywlan 34 mywlan-ssid`	• For *wlan-name* enter, enter the profile name. The range is 1- 32 characters. • For *wlan-id*, enter the WLAN ID. The range is 1-512. • For *ssid*, enter the Service Set IDentifier (SSID) for this WLAN. If the SSID is not specified, the WLAN profile name is set as the SSID.
Step 4	**client vlan id***vlan-group name/vlan-id* **Example:** `Switch (config-wlan) # client vlan VLAN0136`	Configures the VLAN id or group of the WLAN.
Step 5	**no security wpa** **Example:** `Switch (config-wlan) # no security wpa`	The security configuration must be the same for the WLAN created on the GC. This example is for open authentication. For other security types such as open and webauth, appropriate command should be provided.
Step 6	**mobility anchor** *ipaddress* **Example:** `Switch (config-wlan) # mobility anchor 9.3.32.2`	Configures the Guest Controller as mobility anchor.
Step 7	**aaa-override** **Example:** `Switch (config-wlan) # aaa-override`	(Optional) Enables AAA override. AAA override is required for non open authentication in case AAA attributes are to be prioritized. It is required only in case guest user need to be deauthenticated after lifetime or have to give aaa-override attribute to the user.
Step 8	**no shutdown** **Example:** `Switch(config-wlan) # no shutdown`	Enables the WLAN.
Step 9	**end**	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Example: `Switch (config) # end`	
Step 10	**show wireless mobility summary** **Example:** `Switch  # show wireless mobility summary`	Verifies the mobility controller IP address and mobility tunnel status.
Step 11	**show wlan name** *wlan-name/id* **Example:** `Switch # show wlan name mywlan`	Displays the configuration of mobility anchor.

Configuring Mobility Controller

Mobility Controller mode should be enabled using the **wireless mobility controller** command.

SUMMARY STEPS

1. **configure terminal**
2. **wireless mobility group member ip** *ip-address* **public-ip** *ip-address* **group** *group-name*
3. **wireless mobility controller peer-group** *peer-group-name*
4. **wireless mobility controller peer-group** *peer-group-name* **member ip** *ipaddress* **public-ip** *ipaddress*
5. **end**
6. **show wireless mobility summary**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch # configure terminal`	Enters global configuration mode.
Step 2	**wireless mobility group member ip** *ip-address* **public-ip** *ip-address* **group** *group-name* **Example:** `Switch (config) # wireless mobility group member ip 27.0.0.1 public-ip 23.0.0.1 group test`	Adds all peers within the MC group. The *ip-address* should be the guest controller's IP address.
Step 3	**wireless mobility controller peer-group** *peer-group-name*	Creates the switch peer group.

	Command or Action	Purpose
	Example: `Switch (config) # wireless mobility controller peer-group pg`	
Step 4	**wireless mobility controller peer-group** *peer-group-name* **member ip** *ipaddress* **public-ip** *ipaddress* **Example:** `Switch (config) # wireless mobility controller peer-group pg member ip 9.7.136.10 public-ip 9.7.136.10`	Adds the MA to the switch peer group.
Step 5	**end** **Example:** `Switch (config) # end`	Returns to privileged EXEC mode.
Step 6	**show wireless mobility summary** **Example:** `Switch # show wireless mobility summary`	Displays the configuration details.

Obtaining a Web Authentication Certificate

SUMMARY STEPS

1. **configure terminal**
2. **crypto pki import** *trustpoint name* **pkcs12 tftp:** *passphrase*
3. **end**
4. **show crypto pki** *trustpoints cert*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch # configure terminal`	Enters global configuration mode.
Step 2	**crypto pki import** *trustpoint name* **pkcs12 tftp:** *passphrase* **Example:** `Switch (config)#  crypto pki import cert pkcs12 tftp://9.1.0.100/ldapserver-cert.p12 cisco`	Imports certificate.

	Command or Action	Purpose
Step 3	**end** **Example:** `Switch (config)#  end`	Returns to privileged EXEC mode.
Step 4	**show crypto pki** *trustpoints cert* **Example:** `Switch # show crypto pki trustpoints cert`	Displays the configuration details.

Displaying a Web Authentication Certificate

SUMMARY STEPS

1. **show crypto ca certificate verb**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show crypto ca certificate verb** **Example:** `Switch # show crypto ca certificate verb`	Displays the current web authentication certificate details.

Choosing the Default Web Authentication Login Page

AAA override flag should be enabled on the WLAN for web authentication using local or remote AAA server.

SUMMARY STEPS

1. **configure terminal**
2. **parameter-map type webauth** *parameter-map name*
3. **wlan** *wlan-name*
4. **shutdown**
5. **security web-auth**
6. **security web-auth authentication-list** *authentication list name*
7. **security web-auth parameter-map** *parameter-map name*
8. **no shutdown**
9. **end**
10. **show running-config | section** *wlan-name*
11. **show running-config | section parameter-map type webauth** *parameter-map*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch # configure terminal`	Enters global configuration mode.
Step 2	**parameter-map type webauth** *parameter-map name* **Example:** `Switch (config) # parameter-map type webauth test`	Configures the web-auth parameter-map.
Step 3	**wlan** *wlan-name* **Example:** `Switch (config) # wlan wlan10`	For the wlan-name, enter the profile name. The range is 1- 32 characters.
Step 4	**shutdown** **Example:** `Switch (config) # shutdown`	Disables WLAN.
Step 5	**security web-auth** **Example:** `Controller (config-wlan) # security web-auth`	Enables web-auth on WLAN.
Step 6	**security web-auth authentication-list** *authentication list name* **Example:** `Controller (config-wlan) # security web-auth authentication-list test`	Allows you to map the authentication list name with the web-auth WLAN.

	Command or Action	Purpose
Step 7	**security web-auth parameter-map** *parameter-map name* **Example:** `Switch (config) #` **`security web-auth parameter-map`** `test`	Allows you to map the parameter-map name with the web-auth WLAN.
Step 8	**no shutdown** **Example:** `Switch (config) #` **`no shutdown`**	Enables the WLAN.
Step 9	**end** **Example:** `Switch (config) #` **`end`**	Returns to privileged EXEC mode.
Step 10	**show running-config | section** *wlan-name* **Example:** `Switch#` **`show  running-config | section`** `mywlan`	Displays the configuration details.
Step 11	**show running-config | section parameter-map type webauth** *parameter-map* **Example:** `Switch#` **`show  running-config | section  parameter-map type webauth`** `test`	Displays the configuration details.

Choosing a Customized Web Authentication Login Page from an External Web Server

AAA override flag should be enabled on the WLAN for web authentication using local or remote AAA server.

SUMMARY STEPS

1. **configure terminal**
2. **parameter-map type webauth global**
3. **virtual-ip {ipv4 | ipv6}** *ip-address*
4. **parameter-map type webauth** *parameter-map name*
5. **type {authbypass | consent | webauth | webconsent}**
6. **redirect [for-login|on-success|on-failure]** *URL*
7. **redirect portal {ipv4 | ipv6}** *ip-address*
8. **end**
9. show running-config | section parameter-map

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch #` **`configure terminal`**	Enters global configuration mode.
Step 2	**parameter-map type webauth global** **Example:** `Switch (config) #` **`parameter-map type webauth global`**	Configures a global webauth type parameter.
Step 3	**virtual-ip {ipv4 | ipv6}** *ip-address* **Example:** `Switch (config-params-parameter-map) #` **`virtual-ip ipv4`** `1.1.1.1`	Configures the virtual IP address.
Step 4	**parameter-map type webauth** *parameter-map name* **Example:** `Switch (config-params-parameter-map) #` **`parameter-map type webauth`** `test`	Configures the webauth type parameter.
Step 5	**type {authbypass | consent | webauth | webconsent}** **Example:** `Switch (config-params-parameter-map) #` **`type webauth`**	Configures webauth subtypes such as consent, passthru, webauth, or webconsent.

Step 6	**redirect [for-login|on-success|on-failure]** *URL*
	
	Example:
	`Switch (config-params-parameter-map)`
	`#` **`redirect for-login`** `http://9.1.0.100/login.html`
		Configures the redirect URL for the log in page, success page, and failure page.

Step 7	**redirect portal {ipv4 | ipv6}** *ip-address* **Example:** `Switch (config-params-parameter-map) #` **`redirect portal ipv4`** `23.0.0.1`	Configures the external portal IPv4 address.
Step 8	**end** **Example:** `Switch (config-params-parameter-map) #` **`end`**	Returns to privileged EXEC mode.
Step 9	show running-config | section parameter-map **Example:** `Switch #` **`show running-config | section parameter-map`**	Displays the configuration details.

Assigning Login, Login Failure, and Logout Pages per WLAN

SUMMARY STEPS

1. **configure terminal**
2. **parameter-map type webauth** *parameter-map-name*
3. **custom-page login device** *html-filename*
4. **custom-page login expired** *html-filename*
5. **custom-page failure device** *html-filename*
6. **custom-page success device** *html-filename*
7. **end**
8. **show running-config | section parameter-map type webauth** *parameter-map*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch # configure terminal`	Enters global configuration mode.
Step 2	**parameter-map type webauth** *parameter-map-name* **Example:** `Switch (config) # parameter-map type webauth test`	Configures the webauth type parameter.
Step 3	**custom-page login device** *html-filename* **Example:** `Switch (config-params-parameter-map)# custom-page login device device flash:login.html`	Allows you to specify the filename for web authentication customized login page.
Step 4	**custom-page login expired** *html-filename* **Example:** `Switch (config-params-parameter-map)# custom-page login expired device flash:loginexpired.html`	Allows you to specify the filename for web authentication customized login expiry page.
Step 5	**custom-page failure device** *html-filename* **Example:** `Switch (config-params-parameter-map)# custom-page failure device device flash:loginfail.html`	Allows you to specify the filename for web authentication customized login failure page.
Step 6	**custom-page success device** *html-filename* **Example:** `Switch (config-params-parameter-map)# custom-page success device device flash:loginsuccess.html`	Allows you to specify the filename for web authentication customized login success page.
Step 7	**end**	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Example: `Switch (config-params-parameter-map)# end`	
Step 8	**show running-config | section parameter-map type webauth** *parameter-map* **Example:** `Switch (config) # show  running-config | section parameter-map type webauth test`	Displays the configuration details.

Configuring AAA-Override

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name*
3. **aaa-override**
4. **end**
5. **show running-config | section** *wlan-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch # configure terminal`	Enters global configuration mode.
Step 2	**wlan** *wlan-name* **Example:** `Switch (config) # wlan ramban`	For *wlan-name*, enter the profile name. The range is 1- 32 characters.
Step 3	**aaa-override** **Example:** `Switch (config-wlan) # aaa-override`	Enables AAA override on the WLAN.
Step 4	**end** **Example:** `Switch (config-wlan) # end`	Returns to privileged EXEC mode.
Step 5	**show running-config | section** *wlan-name*	Displays the configuration details.

	Command or Action	Purpose
	Example: `Switch # show running-config | section ramban`	

Configuring Client Load Balancing

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name*
3. **shutdown**
4. **mobility anchor** *ip-address1*
5. **mobility anchor** *ip-address2*
6. **no shutdown wlan**
7. **end**
8. **show running-config | section** *wlan-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch # configure terminal`	Enters global configuration mode.
Step 2	**wlan** *wlan-name* **Example:** `Switch (config)# wlan ramban`	For *wlan-name*, enter the profile name.
Step 3	**shutdown** **Example:** `Switch (config-wlan)# shutdown`	Disables WLAN.
Step 4	**mobility anchor** *ip-address1* **Example:** `Switch (config-wlan) # mobility anchor 9.7.136.15`	Configures a guest controller as mobility anchor.
Step 5	**mobility anchor** *ip-address2* **Example:** `Switch (config-wlan) # mobility anchor 9.7.136.16`	Configures a guest controller as mobility anchor.

	Command or Action	Purpose
Step 6	**no shutdown wlan** **Example:** `Switch (config-wlan) # no shutdown wlan`	Enables the WLAN.
Step 7	**end** **Example:** `Switch (config-wlan) # end`	Returns to privileged EXEC mode.
Step 8	**show running-config | section** *wlan-name* **Example:** `Switch # show running-config | section ramban`	Displays the configuration details.

Configuring Preauthentication ACL

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name*
3. **shutdown**
4. **ip access-group web** *preauthrule*
5. **no shutdown**
6. **end**
7. **show wlan name** *wlan-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *wlan-name* **Example:** `Switch (config)# wlan ramban`	For *wlan-name*, enter the profile name.
Step 3	**shutdown** **Example:** `Switch (config-wlan)# shutdown`	Disables the WLAN.

	Command or Action	Purpose
Step 4	**ip access-group web** *preauthrule* **Example:** `Switch (config-wlan)# ip access-group web preauthrule`	Configures ACL that has to be applied before authentication.
Step 5	**no shutdown** **Example:** `Switch (config)# no shutdown`	Enables the WLAN.
Step 6	**end** **Example:** `Switch (config-wlan)# end`	Returns to privileged EXEC mode.
Step 7	**show wlan name** *wlan-name* **Example:** `Switch# show wlan name ramban`	Displays the configuration details.

Configuring IOS ACL Definition

SUMMARY STEPS

1. **configure terminal**
2. **ip access-list extended** *access-list number*
3. **permit udp any eq** *port number* **any**
4. **end**
5. **show access-lists** *ACL number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch # configure terminal`	Enters global configuration mode.
Step 2	**ip access-list extended** *access-list number* **Example:** `Switch (config) # ip access-list extended 102`	Configures extended IP access-list.
Step 3	**permit udp any eq** *port number* **any**	Configures destination host.

	Command or Action	Purpose
	Example: `Switch (config-ext-nacl) # permit udp any eq 8080 any`	
Step 4	**end** **Example:** `Switch (config-wlan) # end`	Returns to privileged EXEC mode.
Step 5	**show access-lists** *ACL number* **Example:** `Switch # show access-lists 102`	Displays the configuration details.

Configuring Webpassthrough

SUMMARY STEPS

1. **configure terminal**
2. **parameter-map type webauth** *parameter-map name*
3. **type consent**
4. **end**
5. **show running-config | section parameter-map type webauth** *parameter-map*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch # configure terminal`	Enters global configuration mode.
Step 2	**parameter-map type webauth** *parameter-map name* **Example:** `Switch (config)  # parameter-map type webauth webparalocal`	Configures the webauth type parameter.
Step 3	**type consent** **Example:** `Switch (config-params-parameter-map) # type consent`	Configures webauth type as consent.
Step 4	**end**	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Example: `Switch (config-params-parameter-map) # end`	
Step 5	**show running-config | section parameter-map type webauth** *parameter-map* **Example:** `Switch (config) # show  running-config | section parameter-map type webauth test`	Displays the configuration details.

Configuration Examples for Guest Access

Example: Creating a Lobby Ambassador Account

This example shows how to configure a lobby ambassador account.

```
Switch# configure terminal
Switch(config)# user-name lobby
Switch(config)# type lobby-admin
Switch(config)# password 0 lobby
Switch(config)# end
Switch#  show running-config | section lobby
    user-name lobby
    creation-time 1351118727
    password 0 lobby
    type lobby-admin
```

Example: Obtaining Web Authentication Certificate

This example shows how to obtain web authentication certificate.

```
Switch# configure terminal
Switch(config)# crypto pki import cert pkcs12 tftp://9.1.0.100/ldapserver-cert.p12 cisco
Switch(config)# end
Switch# show crypto pki trustpoints cert
 Trustpoint cert:
    Subject Name:
    e=rkannajr@cisco.com
    cn=sthaliya-lnx
    ou=WNBU
    o=Cisco
    l=SanJose
    st=California
    c=US
          Serial Number (hex): 00
    Certificate configured.
Switch# show  crypto pki certificates cert
Certificate
  Status: Available
  Certificate Serial Number (hex): 04
  Certificate Usage: General Purpose
  Issuer:
    e=rkannajr@cisco.com
    cn=sthaliya-lnx
```

```
      ou=WNBU
      o=Cisco
      l=SanJose
      st=California
      c=US
    Subject:
      Name: ldapserver
      e=rkannajr@cisco.com
      cn=ldapserver
      ou=WNBU
      o=Cisco
      st=California
      c=US
    Validity Date:
      start date: 07:35:23 UTC Jan 31 2012
      end   date: 07:35:23 UTC Jan 28 2022
    Associated Trustpoints: cert ldap12
    Storage: nvram:rkannajrcisc#4.cer

  CA Certificate
    Status: Available
    Certificate Serial Number (hex): 00
    Certificate Usage: General Purpose
    Issuer:
      e=rkannajr@cisco.com
      cn=sthaliya-lnx
      ou=WNBU
      o=Cisco
      l=SanJose
      st=California
      c=US
    Subject:
      e=rkannajr@cisco.com
      cn=sthaliya-lnx
      ou=WNBU
      o=Cisco
      l=SanJose
      st=California
      c=US
    Validity Date:
      start date: 07:27:56 UTC Jan 31 2012
      end   date: 07:27:56 UTC Jan 28 2022
    Associated Trustpoints: cert ldap12 ldap
    Storage: nvram:rkannajrcisc#0CA.cer
```

Example: Displaying a Web Authentication Certificate

This example shows how to display a web authentication certificate.

```
Switch# show crypto ca certificate verb
     Certificate
     Status: Available
     Version: 3
     Certificate Serial Number (hex): 2A9636AC00000000858B
     Certificate Usage: General Purpose
     Issuer:
    cn=Cisco Manufacturing CA
    o=Cisco Systems
    Subject:
    Name: WS-C3780-6DS-S-2037064C0E80
    Serial Number: PID:WS-C3780-6DS-S SN:FOC1534X12Q
    cn=WS-C3780-6DS-S-2037064C0E80
    serialNumber=PID:WS-C3780-6DS-S SN:FOC1534X12Q
    CRL Distribution Points:
    http://www.cisco.com/security/pki/crl/cmca.crl
    Validity Date:
    start date: 15:43:22 UTC Aug 21 2011
    end   date: 15:53:22 UTC Aug 21 2021
    Subject Key Info:
```

```
Public Key Algorithm: rsaEncryption
RSA Public Key: (1024 bit)
Signature Algorithm: SHA1 with RSA Encryption
Fingerprint MD5: A310B856 A41565F1 1D9410B5 7284CB21
Fingerprint SHA1: 04F180F6 CA1A67AF 9D7F561A 2BB397A1 0F5EB3C9
X509v3 extensions:
X509v3 Key Usage: F0000000
  Digital Signature
  Non Repudiation
  Key Encipherment
  Data Encipherment
X509v3 Subject Key ID: B9EEB123 5A3764B4 5E9C54A7 46E6EECA 02D283F7
X509v3 Authority Key ID: D0C52226 AB4F4660 ECAE0591 C7DC5AD1 B047F76C
Authority Info Access:
Associated Trustpoints: CISCO_IDEVID_SUDI
Key Label: CISCO_IDEVID_SUDI
```

Example: Configuring Guest User Accounts

This example shows how to configure a guest user account.

```
Switch# configure terminal
Switch(config)# user-name guest
Switch(config-user-name)# password 0 guest
Switch(config-user-name)# type network-user description guest guest-user lifetime year 1
month 10 day 3 hour 1 minute 5 second 30
Switch(config-user-name)# end
Switch# show aaa local netuser all
User-Name           : guest
Type                : guest
Password            : guest
Is_passwd_encrypted : No
Descriptio          : guest
Attribute-List      : Not-Configured
First-Login-Time    : Not-Logged-In
Num-Login           : 0
Lifetime            : 1 years 10 months 3 days 1 hours 5 mins 30 secs
Start-Time          : 20:47:37 chennai Dec 21 2012
```

Example: Configuring Mobility Controller

This example shows how to configure a mobility controller.

```
Switch# configure terminal
Switch(config)# wireless mobility group member ip 27.0.0.1 public-ip 23.0.0.1 group test
Switch(config)# wireless mobility controller peer-group pg
Switch(config)# wireless mobility controller peer-group pg member ip 9.7.136.10 public-ip
9.7.136.10
Switch(config)# end
Switch# show wireless mobility summary

Mobility Controller Summary:

Mobility Role                                   : Mobility Controller
Mobility Protocol Port                          : 16666
Mobility Group Name                             : default
Mobility Oracle                                 : Enabled
DTLS Mode                                       : Enabled
Mobility Domain ID for 802.11r                  : 0xac34
Mobility Keepalive Interval                     : 10
Mobility Keepalive Count                        : 3
Mobility Control Message DSCP Value             : 7
Mobility Domain Member Count                    : 3

Link Status is Control Link Status : Data Link Status
```

```
Controllers configured in the Mobility Domain:

IP                Public IP        Group Name       Multicast IP     Link Status
-------------------------------------------------------------------------------
9.9.9.2           -                default          0.0.0.0          UP   : UP
12.12.11.11       12.13.12.12      rasagna-grp                       DOWN : DOWN
27.0.0.1          23.0.0.1         test                              DOWN : DOWN

Switch Peer Group Name             : spg1
Switch Peer Group Member Count     : 0
Bridge Domain ID                   : 0
Multicast IP Address               : 0.0.0.0

Switch Peer Group Name             : pg
Switch Peer Group Member Count     : 1
Bridge Domain ID                   : 0
Multicast IP Address               : 0.0.0.0

IP                Public IP             Link Status
--------------------------------------------------
9.7.136.10        9.7.136.10            DOWN : DOWN
```

Example: Choosing the Default Web Authentication Login Page

This example shows how to choose a default web authentication login page.

```
Switch# configure terminal
Switch(config)# parameter-map type webauth test
This operation will permanently convert all relevant authentication commands to their CPL
control-policy equivalents. As this conversion is irreversible and will
disable the conversion CLI 'authentication display [legacy|new-style]', you are strongly
advised to back up your current configuration before proceeding.
Do you wish to continue? [yes]: yes
Switch(config)# wlan wlan50
Switch(config-wlan)# shutdown
Switch(config-wlan)# security web-auth authentication-list test
Switch(config-wlan)# security web-auth parameter-map test
Switch(config-wlan)# no shutdown
Switch(config-wlan)# end
Switch# show running-config | section wlan50
wlan wlan50 50 wlan50
 security wpa akm cckm
 security wpa wpa1
 security wpa wpa1 ciphers aes
 security wpa wpa1 ciphers tkip
 security web-auth authentication-list test
 security web-auth parameter-map test
 session-timeout 1800
 no shutdown

Switch# show running-config | section parameter-map type webauth test
parameter-map type webauth test
 type webauth
```

Example: Choosing a Customized Web Authentication Login Page from an External Web Server

This example shows how to choose a customized web authentication login page from an external web server.

```
Switch# configure terminal
Switch(config)# parameter-map type webauth global
Switch(config-params-parameter-map)# virtual-ip ipv4 1.1.1.1
Switch(config-params-parameter-map)# parameter-map type webauth test
Switch(config-params-parameter-map)# type webauth
Switch(config-params-parameter-map)# redirect for-login http://9.1.0.100/login.html
Switch(config-params-parameter-map)# redirect portal ipv4 23.0.0.1
Switch(config-params-parameter-map)# end
Switch# show running-config | section parameter-map
```

```
parameter-map type webauth global
virtual-ip ipv4 1.1.1.1
parameter-map type webauth test
type webauth
redirect for-login http://9.1.0.100/login.html
redirect portal ipv4 23.0.0.1
security web-auth parameter-map rasagna-auth-map
security web-auth parameter-map test
```

Example: Assigning Login, Login Failure, and Logout Pages per WLAN

This example shows how to assign login, login failure and logout pages per WLAN.

```
Switch# configure terminal
Switch(config)# parameter-map type webauth test
Switch(config-params-parameter-map)# custom-page login device flash:loginsantosh.html
Switch(config-params-parameter-map)# custom-page login expired device flash:loginexpire.html
Switch(config-params-parameter-map)# custom-page failure device flash:loginfail.html
Switch(config-params-parameter-map)# custom-page success device flash:loginsucess.html
Switch(config-params-parameter-map)# end
Switch# show running-config | section parameter-map type webauth test
 parameter-map type webauth test
 type webauth
 redirect for-login http://9.1.0.100/login.html
 redirect portal ipv4 23.0.0.1
 custom-page login device flash:loginsantosh.html
 custom-page success device flash:loginsucess.html
 custom-page failure device flash:loginfail.html
 custom-page login expired device flash:loginexpire.html
```

Example: Configuring AAA-Override

This example shows how to configure aaa-override.

```
Switch# configure terminal
Switch(config)# wlan fff
Switch(config-wlan)# aaa-override
Switch(config-wlan)# end
Switch# show running-config | section fff
 wlan fff 44 fff
 aaa-override
 shutdown
```

Example: Configuring Client Load Balancing

This example shows how to configure client load balancing.

```
Switch# configure terminal
Switch(config)# wlan fff
Switch(config-wlan)# shutdown
Switch(config-wlan)# mobility anchor 9.7.136.15
Switch(config-wlan)# mobility anchor 9.7.136.16
Switch(config-wlan)# no shutdown wlan
Switch(config-wlan)# end
Switch# show running-config | section fff
wlan fff 44 fff
 aaa-override
 shutdown
```

Example: Configuring Preauthentication ACL

This example shows how to configure preauthentication ACL.

```
Switch# configure terminal
Switch(config)# wlan fff
Switch(config-wlan)# shutdown
Switch(config-wlan)# ip access-group web preauthrule
Switch(config-wlan)# no shutdown
Switch(config-wlan)# end
Switch# show wlan name fff
```

Example: Configuring IOS ACL Definition

This example shows how to configure IOS ACL definition.

```
Switch# configure terminal
Switch(config)# ip access-list extended 102
Switch(config-ext-nacl)# permit udp any eq 8080 any
Switch(config-ext-nacl)# end
Switch# show access-lists 102
 Extended IP access list 102
    10 permit udp any eq 8080 any
```

Example: Configuring Webpassthrough

This example shows how to configure webpassthrough.

```
Switch# configure terminal
Switch(config)# parameter-map type webauth webparalocal
Switch(config-params-parameter-map)# type consent
Switch(config-params-parameter-map)# end
Switch# show running-config | section parameter-map type webauth test
 parameter-map type webauth test
 type webauth
 redirect for-login http://9.1.0.100/login.html
 redirect portal ipv4 23.0.0.1
```

Additional References for Guest Access

Related Documents

Related Topic	Document Title
Mobility CLI commands	*Mobility Command Reference, Cisco IOS XE 3SE (Cisco WLC 5700 Series)*
Mobility configuration	*Mobility Configuration Guide, Cisco IOS XE 3SE (Cisco WLC 5700 Series)*
Security CLI commands	*Security Command Reference, Cisco IOS Release 3SE (Cisco WLC 5700 Series)*
Configuring web-based authentication on the Catalyst 5700 Series Wireless Controller	*Security Configuration Guide, Cisco IOS Release 3SE (Cisco WLC 5700 Series)*

Related Topic	Document Title
Wired guest access configuration and commands	*Identity Based Networking Services*

Standards and RFCs

Standard/RFC	Title
None	-

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Guest Access

Releases	Feature Information
Cisco IOS XE Release 3.2SE	This feature was introduced.

CHAPTER 60

Configuring Intrusion Detection System

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the <TBD>

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About Intrusion Detection System

The Cisco Intrusion Detection System/Intrusion Prevention System (CIDS/CIPS) instructs switches to block certain clients from accessing the wireless network when attacks involving these clients are detected at Layer 3 through Layer 7. This system offers significant network protection by helping to detect, classify, and stop threats including worms, spyware/adware, network viruses, and application abuse. Two methods are available to detect potential attacks:

- IDS sensors
- IDS signatures

IDS sensors can be configured to detect various types of IP-level attacks in the network. When the sensors identify an attack, they can alert the switch to shun the offending client. When a new IDS sensor is added, the IDS sensor should be registered with the switch so that the switch can query the sensor to get the list of shunned clients.

When an IDS sensor detects a suspicious client, it alerts the switch to shun this client. The shun entry is distributed to all switches within the same mobility group. If the client to be shunned is currently joined to a switch in this mobility group, the anchor switch adds this client to the dynamic exclusion list, and the foreign switch removes the client. The next time that the client tries to connect to a switch, the anchor switch rejects the handoff and informs the foreign switch that the client is being excluded.

How to Configure Intrusion Detection System

Configuring IDS Sensors

SUMMARY STEPS

1. **configure terminal**
2. **wireless wps cids-sensor** *index* [**ip-address** *ip-addr* **username** *username* **password** *password_type password*]
3. **wireless wps cids-sensor** *index*
4. [**default exit fingerprint interval no port shutdown**]
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**wireless wps cids-sensor** *index* [**ip-address** *ip-addr* **username** *username* **password** *password_type password*] **Example:** `Switch(config)# wireless wps cids-sensor 2 231.1.1.1 admin pwd123`	Configures the IDS sensors that holds and internal index number. The index parameter determines the sequence in which the controller consults the IDS sensors. The controller supports up to five IDS sensors. • **ip-address–** [optional] Provide the IP address for the IDS. • **username–** [optional] Configures the username for the IDS. • **password–** [optional] Configures the password for the respective username.
Step 3	**wireless wps cids-sensor** *index* **Example:** `Switch(config)# wireless wps cids-sensor 1`	Enters the IDS configuration submode.
Step 4	[**default exit fingerprint interval no port shutdown**]	Configures various IDS parameters. • **default–** [optional] Sets a command to its default.

	Command or Action	Purpose
	Example: `Switch(config-cids-index)# default`	• **exit–** [optional] Exits the submode. • **fingerprint–** [optional] Configures the sensor's TLS fingerprint. • **interval–** [optional] Configures the sensor's query interval. The range is between 10-3600 seconds. • **no–** [optional] Negates a command or set its defaults. • **port–** [optional] Configures the sensor's port number. • **shutdown–** [optional] Shuts down the intrusion detection sensor.
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Monitoring Intrusion Detection System

Table 147: Commands for Monitoring Wireless Multicast

Commands	Description
show wireless wps cids-sensor *index*	Displays the IDS configuration of the IDS sensor with the mentioned index value.
show wireless wps cids-sensor summary	Displays the list of all the configured IDS with their respective values like index, ip-address, port number, interval value, status and last query.
show wireless wps shun-list	Displays the list of the IDS shun list.

PART VIII

Layer 2 (Link Aggregation)

CHAPTER 61

Configuring EtherChannels

Configuring EtherChannels

You can configure EtherChannels on Layer 2 and Layer 3 ports on the switch. EtherChannel provides fault-tolerant high-speed links between switches, routers, and servers. You can use Etherchannels to increase the bandwidth between the wiring closets and the data center, and you can deploy it anywhere in the network where bottlenecks are likely to occur. EtherChannel provides automatic recovery for the loss of a link by redistributing the load across the remaining links. If a link fails, EtherChannel redirects traffic from the failed link to the remaining links in the channel without intervention.

Feature Information for EtherChannels

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.
Cisco IOS XE 3.3SE	Support for the LACP max-bundle feature and the port channel min-links features was added.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Restrictions for EtherChannels

The following are restrictions for EtherChannels:

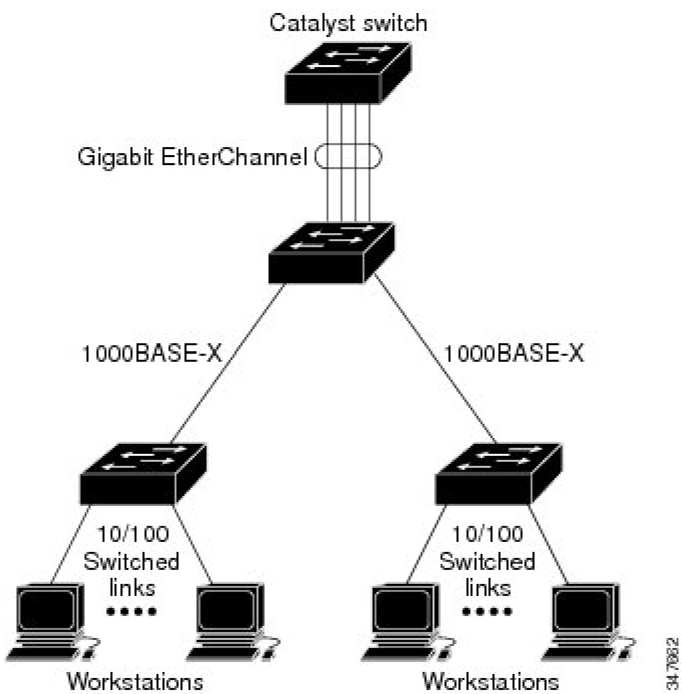
- All ports in an EtherChannel must be assigned to the same VLAN or they must be configured as trunk ports.
- Layer 3 EtherChannels are not supported if running the LAN Base license feature set.
- You cannot have a switch stack containing a mix of Catalyst 3850 and Catalyst 3650 switches.

Information About EtherChannels

EtherChannel Overview

An EtherChannel consists of individual Ethernet links bundled into a single logical link.

Figure 61: Typical EtherChannel Configuration



The EtherChannel provides full-duplex bandwidth up to 8 Gb/s (Gigabit EtherChannel) or 80 Gb/s (10-Gigabit EtherChannel) between your switch and another switch or host.

Each EtherChannel can consist of up to eight compatibly configured Ethernet ports.

The number of EtherChannels is limited to 128.

All ports in each EtherChannel must be configured as either Layer 2 or Layer 3 ports. The EtherChannel Layer 3 ports are made up of routed ports. Routed ports are physical ports configured to be in Layer 3 mode by using the **no switchport** interface configuration command.

Related Topics

Configuring Layer 2 EtherChannels , on page 1341
EtherChannel Configuration Guidelines, on page 1339
Layer 2 EtherChannel Configuration Guidelines, on page 1340
Layer 3 EtherChannel Configuration Guidelines, on page 1340
Default EtherChannel Configuration, on page 1337

EtherChannel Modes

You can configure an EtherChannel in one of these modes: Port Aggregation Protocol (PAgP), Link Aggregation Control Protocol (LACP), or On. Configure both ends of the EtherChannel in the same mode:

- When you configure one end of an EtherChannel in either PAgP or LACP mode, the system negotiates with the other end of the channel to determine which ports should become active. If the remote port cannot negotiate an EtherChannel, the local port is put into an independent state and continues to carry data traffic as would any other single link. The port configuration does not change, but the port does not participate in the EtherChannel.
- When you configure an EtherChannel in the **on** mode, no negotiations take place. The switch forces all compatible ports to become active in the EtherChannel. The other end of the channel (on the other switch) must also be configured in the **on** mode; otherwise, packet loss can occur.

Related Topics

Configuring Layer 2 EtherChannels , on page 1341
EtherChannel Configuration Guidelines, on page 1339
Layer 2 EtherChannel Configuration Guidelines, on page 1340
Layer 3 EtherChannel Configuration Guidelines, on page 1340
Default EtherChannel Configuration, on page 1337

EtherChannel Link Failover

If a link within an EtherChannel fails, traffic previously carried over that failed link moves to the remaining links within the EtherChannel. If traps are enabled on the switch, a trap is sent for a failure that identifies the switch, the EtherChannel, and the failed link. Inbound broadcast and multicast packets on one link in an EtherChannel are blocked from returning on any other link of the EtherChannel.

Related Topics

Configuring Layer 2 EtherChannels , on page 1341
EtherChannel Configuration Guidelines, on page 1339
Layer 2 EtherChannel Configuration Guidelines, on page 1340
Layer 3 EtherChannel Configuration Guidelines, on page 1340
Default EtherChannel Configuration, on page 1337

Port-Channel Interfaces

When you create an EtherChannel, a port-channel logical interface is involved:

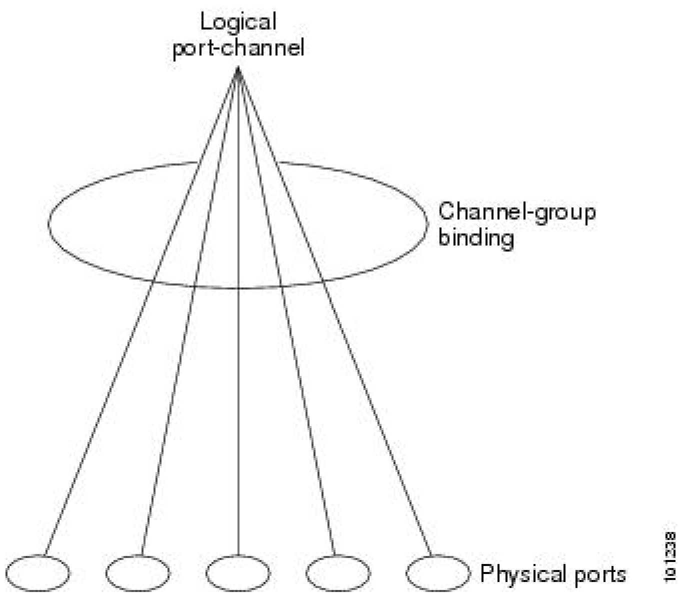
- With Layer 2 ports, use the **channel-group** interface configuration command to dynamically create the port-channel logical interface.

  You also can use the **interface port-channel** *port-channel-number* global configuration command to manually create the port-channel logical interface, but then you must use the **channel-group** *channel-group-number* command to bind the logical interface to a physical port. The *channel-group-number* can be the same as the *port-channel-number,* or you can use a new number. If you use a new number, the **channel-group** command dynamically creates a new port channel.

- With Layer 3 ports, you should manually create the logical interface by using the **interface port-channel** global configuration command followed by the **no switchport** interface configuration command. You then manually assign an interface to the EtherChannel by using the **channel-group** interface configuration command.

The **channel-group** command binds the physical port and the logical interface together. Each EtherChannel has a port-channel logical interface numbered from 1 to 48. This port-channel interface number corresponds to the one specified with the **channel-group** interface configuration command.

Figure 62: Relationship of Physical Ports, Logical Port Channels, and Channel Groups



After you configure an EtherChannel, configuration changes applied to the port-channel interface apply to all the physical ports assigned to the port-channel interface. Configuration changes applied to the physical port affect only the port where you apply the configuration. To change the parameters of all ports in an EtherChannel, apply configuration commands to the port-channel interface, for example, spanning-tree commands or commands to configure a Layer 2 EtherChannel as a trunk.

Related Topics

EtherChannel Configuration Guidelines, on page 1339
Layer 2 EtherChannel Configuration Guidelines, on page 1340
Layer 3 EtherChannel Configuration Guidelines, on page 1340
Default EtherChannel Configuration, on page 1337
EtherChannel Configuration Guidelines, on page 1339
Layer 2 EtherChannel Configuration Guidelines, on page 1340
Layer 3 EtherChannel Configuration Guidelines, on page 1340
Default EtherChannel Configuration, on page 1337

Port Aggregation Protocol

The Port Aggregation Protocol (PAgP) is a Cisco-proprietary protocol that can be run only on Cisco switches and on those switches licensed by vendors to support PAgP. PAgP facilitates the automatic creation of EtherChannels by exchanging PAgP packets between Ethernet ports. PAgP can be enabled on cross-stack EtherChannels.

By using PAgP, the switch or switch stack learns the identity of partners capable of supporting PAgP and the capabilities of each port. It then dynamically groups similarly configured ports (on a single switch in the stack) into a single logical link (channel or aggregate port). Similarly configured ports are grouped based on hardware, administrative, and port parameter constraints. For example, PAgP groups the ports with the same speed, duplex mode, native VLAN, VLAN range, and trunking status and type. After grouping the links into an EtherChannel, PAgP adds the group to the spanning tree as a single switch port.

PAgP Learn Method and Priority

Network devices are classified as PAgP physical learners or aggregate-port learners. A device is a physical learner if it learns addresses by physical ports and directs transmissions based on that knowledge. A device is an aggregate-port learner if it learns addresses by aggregate (logical) ports. The learn method must be configured the same at both ends of the link.

When a device and its partner are both aggregate-port learners, they learn the address on the logical port-channel. The device sends packets to the source by using any of the ports in the EtherChannel. With aggregate-port learning, it is not important on which physical port the packet arrives.

PAgP cannot automatically detect when the partner device is a physical learner and when the local device is an aggregate-port learner. Therefore, you must manually set the learning method on the local device to learn addresses by physical ports. You also must set the load-distribution method to source-based distribution, so that any given source MAC address is always sent on the same physical port.

You also can configure a single port within the group for all transmissions and use other ports for hot-standby. The unused ports in the group can be swapped into operation in just a few seconds if the selected single port loses hardware-signal detection. You can configure which port is always selected for packet transmission by changing its priority with the **pagp port-priority** interface configuration command. The higher the priority, the more likely that the port will be selected.

Note The switch supports address learning only on aggregate ports even though the **physical-port** keyword is provided in the CLI. The **pagp learn-method** command and the **pagp port-priority** command have no effect on the switch hardware, but they are required for PAgP interoperability with devices that only support address learning by physical ports, such as the Catalyst 1900 switch.

When the link partner of the switch is a physical learner, we recommend that you configure the switch as a physical-port learner by using the **pagp learn-method physical-port** interface configuration command. Set the load-distribution method based on the source MAC address by using the **port-channel load-balance src-mac** global configuration command. The switch then sends packets to the physcial learner using the same port in the EtherChannel from which it learned the source address. Only use the **pagp learn-method** command in this situation.

Related Topics

Configuring the PAgP Learn Method and Priority , on page 1345

EtherChannel Configuration Guidelines, on page 1339
Layer 2 EtherChannel Configuration Guidelines, on page 1340
Layer 3 EtherChannel Configuration Guidelines, on page 1340
Default EtherChannel Configuration, on page 1337
Monitoring EtherChannel, PAgP, and LACP Status, on page 1346

PAgP Interaction with Virtual Switches and Dual-Active Detection

A virtual switch can be two or more core switches connected by virtual switch links (VSLs) that carry control and data traffic between them. One of the switches is in active mode. The others are in standby mode. For redundancy, remote switches are connected to the virtual switch by remote satellite links (RSLs).

If the VSL between two switches fails, one switch does not know the status of the other. Both switches could change to the active mode, causing a *dual-active situation* in the network with duplicate configurations (including duplicate IP addresses and bridge identifiers). The network might go down.

To prevent a dual-active situation, the core switches send PAgP protocol data units (PDUs) through the RSLs to the remote switches. The PAgP PDUs identify the active switch, and the remote switches forward the PDUs to core switches so that the core switches are in sync. If the active switch fails or resets, the standby switch takes over as the active switch. If the VSL goes down, one core switch knows the status of the other and does not change its state.

PAgP Interaction with Other Features

The Dynamic Trunking Protocol (DTP) and the Cisco Discovery Protocol (CDP) send and receive packets over the physical ports in the EtherChannel. Trunk ports send and receive PAgP protocol data units (PDUs) on the lowest numbered VLAN.

In Layer 2 EtherChannels, the first port in the channel that comes up provides its MAC address to the EtherChannel. If this port is removed from the bundle, one of the remaining ports in the bundle provides its MAC address to the EtherChannel. For Layer 3 EtherChannels, the MAC address is allocated by the active switch as soon as the interface is created (through the **interface port-channel** global configuration command).

PAgP sends and receives PAgP PDUs only from ports that are up and have PAgP enabled for the auto or desirable mode.

Link Aggregation Control Protocol

The LACP is defined in IEEE 802.3ad and enables Cisco switches to manage Ethernet channels between switches that conform to the IEEE 802.3ad protocol. LACP facilitates the automatic creation of EtherChannels by exchanging LACP packets between Ethernet ports.

By using LACP, the switch or switch stack learns the identity of partners capable of supporting LACP and the capabilities of each port. It then dynamically groups similarly configured ports into a single logical link (channel or aggregate port). Similarly configured ports are grouped based on hardware, administrative, and port parameter constraints. For example, LACP groups the ports with the same speed, duplex mode, native VLAN, VLAN range, and trunking status and type. After grouping the links into an EtherChannel, LACP adds the group to the spanning tree as a single switch port.

LACP Modes

LACP modes specify whether a port can send LACP packets or only receive LACP packets.

Table 148: EtherChannel LACP Modes

Mode	Description
active	Places a port into an active negotiating state in which the port starts negotiations with other ports by sending LACP packets.
passive	Places a port into a passive negotiating state in which the port responds to LACP packets that it receives, but does not start LACP packet negotiation. This setting minimizes the transmission of LACP packets.

Both the **active** and **passive LACP** modes enable ports to negotiate with partner ports to an EtherChannel based on criteria such as port speed, and for Layer 2 EtherChannels, based on trunk state and VLAN numbers.

Ports can form an EtherChannel when they are in different LACP modes as long as the modes are compatible. For example:

- A port in the **active** mode can form an EtherChannel with another port that is in the **active** or **passive** mode.
- A port in the **passive** mode cannot form an EtherChannel with another port that is also in the **passive** mode because neither port starts LACP negotiation.

Related Topics

Configuring Layer 2 EtherChannels , on page 1341
EtherChannel Configuration Guidelines, on page 1339
Layer 2 EtherChannel Configuration Guidelines, on page 1340
Layer 3 EtherChannel Configuration Guidelines, on page 1340
Default EtherChannel Configuration, on page 1337

LACP Interaction with Other Features

The DTP and the CDP send and receive packets over the physical ports in the EtherChannel. Trunk ports send and receive LACP PDUs on the lowest numbered VLAN.

In Layer 2 EtherChannels, the first port in the channel that comes up provides its MAC address to the EtherChannel. If this port is removed from the bundle, one of the remaining ports in the bundle provides its MAC address to the EtherChannel. For Layer 3 EtherChannels, the MAC address is allocated by the active switch as soon as the interface is created through the **interface port-channel** global configuration command.

LACP sends and receives LACP PDUs only from ports that are up and have LACP enabled for the active or passive mode.

EtherChannel On Mode

EtherChannel **on** mode can be used to manually configure an EtherChannel. The **on** mode forces a port to join an EtherChannel without negotiations. The **on** mode can be useful if the remote device does not support PAgP or LACP. In the **on** mode, a usable EtherChannel exists only when the switches at both ends of the link are configured in the **on** mode.

Ports that are configured in the **on** mode in the same channel group must have compatible port characteristics, such as speed and duplex. Ports that are not compatible are suspended, even though they are configured in the **on** mode.

Caution You should use care when using the **on** mode. This is a manual configuration, and ports on both ends of the EtherChannel must have the same configuration. If the group is misconfigured, packet loss or spanning-tree loops can occur.

Load-Balancing and Forwarding Methods

EtherChannel balances the traffic load across the links in a channel by reducing part of the binary pattern formed from the addresses in the frame to a numerical value that selects one of the links in the channel. You can specify one of several different load-balancing modes, including load distribution based on MAC addresses, IP addresses, source addresses, destination addresses, or both source and destination addresses. The selected mode applies to all EtherChannels configured on the switch.

You configure the load-balancing and forwarding method by using the **port-channel load-balance** and the **port-channel load-balance extended**global configuration commands.

Related Topics

Configuring EtherChannel Load-Balancing , on page 1343
EtherChannel Configuration Guidelines, on page 1339
Layer 2 EtherChannel Configuration Guidelines, on page 1340
Layer 3 EtherChannel Configuration Guidelines, on page 1340
Default EtherChannel Configuration, on page 1337

MAC Address Forwarding

With source-MAC address forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the channel based on the source-MAC address of the incoming packet. Therefore, to provide load-balancing, packets from different hosts use different ports in the channel, but packets from the same host use the same port in the channel.

With destination-MAC address forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the channel based on the destination host's MAC address of the incoming packet. Therefore, packets to the same destination are forwarded over the same port, and packets to a different destination are sent on a different port in the channel.

With source-and-destination MAC address forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the channel based on both the source and destination MAC addresses. This forwarding method, a combination source-MAC and destination-MAC address forwarding methods of load distribution, can be used if it is not clear whether source-MAC or destination-MAC address forwarding is better suited on a particular switch. With source-and-destination MAC-address forwarding, packets sent from host A to host B, host A to host C, and host C to host B could all use different ports in the channel.

Related Topics

Configuring EtherChannel Load-Balancing , on page 1343
EtherChannel Configuration Guidelines, on page 1339

[Layer 2 EtherChannel Configuration Guidelines, on page 1340]
[Layer 3 EtherChannel Configuration Guidelines, on page 1340]
[Default EtherChannel Configuration, on page 1337]

IP Address Forwarding

With source-IP address-based forwarding, packets are distributed across the ports in the EtherChannel based on the source-IP address of the incoming packet. To provide load balancing, packets from different IP addresses use different ports in the channel, and packets from the same IP address use the same port in the channel.

With destination-IP address-based forwarding, packets are distributed across the ports in the EtherChannel based on the destination-IP address of the incoming packet. To provide load balancing, packets from the same IP source address sent to different IP destination addresses could be sent on different ports in the channel. Packets sent from different source IP addresses to the same destination IP address are always sent on the same port in the channel.

With source-and-destination IP address-based forwarding, packets are distributed across the ports in the EtherChannel based on both the source and destination IP addresses of the incoming packet. This forwarding method, a combination of source-IP and destination-IP address-based forwarding, can be used if it is not clear whether source-IP or destination-IP address-based forwarding is better suited on a particular switch. In this method, packets sent from the IP address A to IP address B, from IP address A to IP address C, and from IP address C to IP address B could all use different ports in the channel.

Related Topics

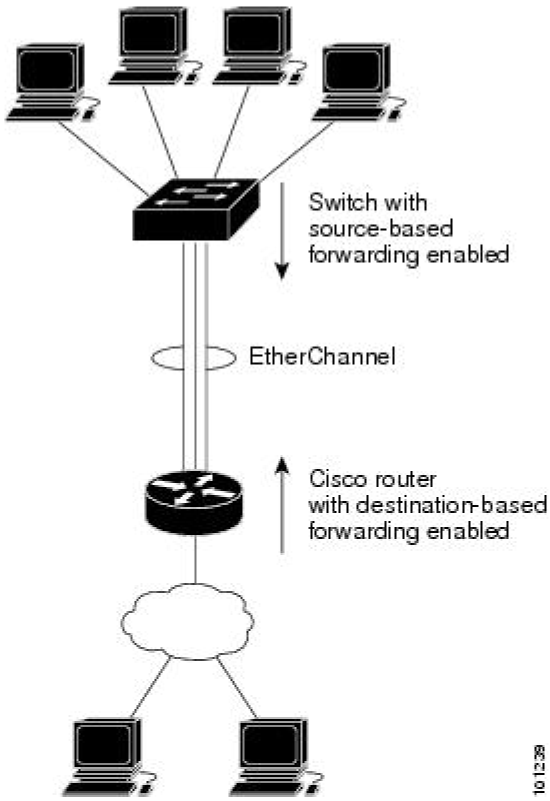
Configuring EtherChannel Load-Balancing , on page 1343
EtherChannel Configuration Guidelines, on page 1339
Layer 2 EtherChannel Configuration Guidelines, on page 1340
Layer 3 EtherChannel Configuration Guidelines, on page 1340
Default EtherChannel Configuration, on page 1337

Load-Balancing Advantages

Different load-balancing methods have different advantages, and the choice of a particular load-balancing method should be based on the position of the switch in the network and the kind of traffic that needs to be load-distributed.

In the following figure, an EtherChannel of four workstations communicates with a router. Because the router is a single MAC-address device, source-based forwarding on the switch EtherChannel ensures that the switch uses all available bandwidth to the router. The router is configured for destination-based forwarding because the large number of workstations ensures that the traffic is evenly distributed from the router EtherChannel.

Figure 63: Load Distribution and Forwarding Methods



Use the option that provides the greatest variety in your configuration. For example, if the traffic on a channel is going only to a single MAC address, using the destination-MAC address always chooses the same link in the channel. Using source addresses or IP addresses might result in better load-balancing.

Related Topics

Configuring EtherChannel Load-Balancing , on page 1343
EtherChannel Configuration Guidelines, on page 1339
Layer 2 EtherChannel Configuration Guidelines, on page 1340
Layer 3 EtherChannel Configuration Guidelines, on page 1340
Default EtherChannel Configuration, on page 1337

Default EtherChannel Configuration

The default EtherChannel configuration is described in this table.

Table 149: Default EtherChannel Configuration

Feature	Default Setting
Channel groups	None assigned.
Port-channel logical interface	None defined.
PAgP mode	No default.
PAgP learn method	Aggregate-port learning on all ports.
PAgP priority	128 on all ports.
LACP mode	No default.
LACP learn method	Aggregate-port learning on all ports.
LACP port priority	32768 on all ports.
LACP system priority	32768.
LACP system ID	LACP system priority and the switch or stack MAC address.
Load-balancing	Load distribution on the switch is based on the source-MAC address of the incoming packet.

Related Topics

Configuring Layer 2 EtherChannels , on page 1341
EtherChannel Overview, on page 1329
EtherChannel Modes, on page 1330
EtherChannel Link Failover, on page 1330
LACP Modes, on page 1333
Port-Channel Interfaces, on page 1330
Port-Channel Interfaces, on page 1330
Configuring EtherChannel Load-Balancing , on page 1343
Load-Balancing and Forwarding Methods, on page 1335
MAC Address Forwarding, on page 1335
IP Address Forwarding, on page 1336
Load-Balancing Advantages, on page 1336
Configuring the PAgP Learn Method and Priority , on page 1345
PAgP Learn Method and Priority, on page 1332

EtherChannel Configuration Guidelines

If improperly configured, some EtherChannel ports are automatically disabled to avoid network loops and other problems. Follow these guidelines to avoid configuration problems:

- Do not try to configure more than 128 EtherChannels on the switch or switch stack.
- Configure a PAgP EtherChannel with up to eight Ethernet ports of the same type.
- Configure a LACP EtherChannel with up to 16 Ethernet ports of the same type. Up to eight ports can be active, and up to eight ports can be in standby mode.
- Configure all ports in an EtherChannel to operate at the same speeds and duplex modes.
- Enable all ports in an EtherChannel. A port in an EtherChannel that is disabled by using the **shutdown** interface configuration command is treated as a link failure, and its traffic is transferred to one of the remaining ports in the EtherChannel.
- When a group is first created, all ports follow the parameters set for the first port to be added to the group. If you change the configuration of one of these parameters, you must also make the changes to all ports in the group:
    - Allowed-VLAN list
    - Spanning-tree path cost for each VLAN
    - Spanning-tree port priority for each VLAN
    - Spanning-tree Port Fast setting
- Do not configure a port to be a member of more than one EtherChannel group.
- Do not configure an EtherChannel in both the PAgP and LACP modes. EtherChannel groups running PAgP and LACP can coexist on the same switch or on different switches in the stack. Individual EtherChannel groups can run either PAgP or LACP, but they cannot interoperate.
- Do not configure a secure port as part of an EtherChannel or the reverse.
- Do not configure a port that is an active or a not-yet-active member of an EtherChannel as an IEEE 802.1x port. If you try to enable IEEE 802.1x on an EtherChannel port, an error message appears, and IEEE 802.1x is not enabled.
- If EtherChannels are configured on switch interfaces, remove the EtherChannel configuration from the interfaces before globally enabling IEEE 802.1x on a switch by using the **dot1x system-auth-control** global configuration command.
- If cross-stack EtherChannel is configured and the switch stack partitions, loops and forwarding issues can occur.

Related Topics

Configuring Layer 2 EtherChannels , on page 1341
EtherChannel Overview, on page 1329
EtherChannel Modes, on page 1330
EtherChannel Link Failover, on page 1330
LACP Modes, on page 1333

Port-Channel Interfaces, on page 1330
Port-Channel Interfaces, on page 1330
Configuring EtherChannel Load-Balancing , on page 1343
Load-Balancing and Forwarding Methods, on page 1335
MAC Address Forwarding, on page 1335
IP Address Forwarding, on page 1336
Load-Balancing Advantages, on page 1336
Configuring the PAgP Learn Method and Priority , on page 1345
PAgP Learn Method and Priority, on page 1332

Layer 2 EtherChannel Configuration Guidelines

When configuring Layer 2 EtherChannels, follow these guidelines:

- Assign all ports in the EtherChannel to the same VLAN, or configure them as trunks. Ports with different native VLANs cannot form an EtherChannel.
- An EtherChannel supports the same allowed range of VLANs on all the ports in a trunking Layer 2 EtherChannel. If the allowed range of VLANs is not the same, the ports do not form an EtherChannel even when PAgP is set to the **auto** or **desirable** mode.
- Ports with different spanning-tree path costs can form an EtherChannel if they are otherwise compatibly configured. Setting different spanning-tree path costs does not, by itself, make ports incompatible for the formation of an EtherChannel.

Related Topics

Configuring Layer 2 EtherChannels , on page 1341
EtherChannel Overview, on page 1329
EtherChannel Modes, on page 1330
EtherChannel Link Failover, on page 1330
LACP Modes, on page 1333
Port-Channel Interfaces, on page 1330
Port-Channel Interfaces, on page 1330
Configuring EtherChannel Load-Balancing , on page 1343
Load-Balancing and Forwarding Methods, on page 1335
MAC Address Forwarding, on page 1335
IP Address Forwarding, on page 1336
Load-Balancing Advantages, on page 1336
Configuring the PAgP Learn Method and Priority , on page 1345
PAgP Learn Method and Priority, on page 1332

Layer 3 EtherChannel Configuration Guidelines

- For Layer 3 EtherChannels, assign the Layer 3 address to the port-channel logical interface, not to the physical ports in the channel.

Related Topics

Configuring Layer 2 EtherChannels , on page 1341
EtherChannel Overview, on page 1329
EtherChannel Modes, on page 1330
EtherChannel Link Failover, on page 1330
LACP Modes, on page 1333
Port-Channel Interfaces, on page 1330
Port-Channel Interfaces, on page 1330
Configuring EtherChannel Load-Balancing , on page 1343
Load-Balancing and Forwarding Methods, on page 1335
MAC Address Forwarding, on page 1335
IP Address Forwarding, on page 1336
Load-Balancing Advantages, on page 1336
Configuring the PAgP Learn Method and Priority , on page 1345
PAgP Learn Method and Priority, on page 1332

How to Configure EtherChannels

After you configure an EtherChannel, configuration changes applied to the port-channel interface apply to all the physical ports assigned to the port-channel interface, and configuration changes applied to the physical port affect only the port where you apply the configuration.

Configuring Layer 2 EtherChannels

You configure Layer 2 EtherChannels by assigning ports to a channel group with the **channel-group** interface configuration command. This command automatically creates the port-channel logical interface.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode** {**access** | **trunk**}
4. **switchport access vlan** *vlan-id*
5. **channel-group** *channel-group-number* **mode** {**auto** [**non-silent**] | **desirable** [**non-silent**] | **on** } | { **active** | **passive**}
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/1`	Specifies a physical port, and enters interface configuration mode. Valid interfaces are physical ports. For a PAgP EtherChannel, you can configure up to eight ports of the same type and speed for the same group. For a LACP EtherChannel, you can configure up to 16 Ethernet ports of the same type. Up to eight ports can be active, and up to eight ports can be in standby mode.
Step 3	**switchport mode** {**access** | **trunk**} **Example:** `Switch(config-if)# switchport mode access`	Assigns all ports as static-access ports in the same VLAN, or configure them as trunks. If you configure the port as a static-access port, assign it to only one VLAN. The range is 1 to 4094.
Step 4	**switchport access vlan** *vlan-id* **Example:** `Switch(config-if)# switchport access vlan 22`	(Optional) If you configure the port as a static-access port, assign it to only one VLAN. The range is 1 to 4094.
Step 5	**channel-group** *channel-group-number* **mode** {**auto** [**non-silent**] | **desirable** [**non-silent**] | **on** } | { **active** | **passive**} **Example:** `Switch(config-if)# channel-group 5 mode auto`	Assigns the port to a channel group, and specifies the PAgP or the LACP mode. For *channel-group-number*, the range is 1 to 128. For **mode**, select one of these keywords: • **auto** —Enables PAgP only if a PAgP device is detected. It places the port into a passive negotiating state, in which the port responds to PAgP packets it receives but does not start PAgP packet negotiation. This keyword is not supported when EtherChannel members are from different switches in the switch stack. • **desirable** —Unconditionally enables PAgP. It places the port into an active negotiating state, in which the port starts negotiations with other ports by sending PAgP packets. This keyword is not supported when EtherChannel members are from different switches in the switch stack. • **on** —Forces the port to channel without PAgP or LACP. In the **on** mode, an EtherChannel exists only when a port group in the **on** mode is connected to another port group in the **on** mode.

	Command or Action	Purpose
		• **non-silent** —(Optional) If your switch is connected to a partner that is PAgP-capable, configures the switch port for nonsilent operation when the port is in the **auto** or **desirable** mode. If you do not specify **non-silent**, silent is assumed. The silent setting is for connections to file servers or packet analyzers. This setting allows PAgP to operate, to attach the port to a channel group, and to use the port for transmission. • **active**—Enables LACP only if a LACP device is detected. It places the port into an active negotiating state in which the port starts negotiations with other ports by sending LACP packets. • **passive** —Enables LACP on the port and places it into a passive negotiating state in which the port responds to LACP packets that it receives, but does not start LACP packet negotiation.
Step 6	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Related Topics

EtherChannel Overview, on page 1329
EtherChannel Modes, on page 1330
EtherChannel Link Failover, on page 1330
LACP Modes, on page 1333
EtherChannel Configuration Guidelines, on page 1339
Layer 2 EtherChannel Configuration Guidelines, on page 1340
Layer 3 EtherChannel Configuration Guidelines, on page 1340
Default EtherChannel Configuration, on page 1337

Configuring EtherChannel Load-Balancing

You can configure EtherChannel load-balancing to use one of several different forwarding methods.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **port-channel load-balance** { **dst-ip** | **dst-mac** | **dst-mixed-ip-port** | **dst-port** | **extended** [**dst-ip** | **dst-mac** | **dst-port** | **ipv6-label** | **l3-proto** | **src-ip** | **src-mac** | **src-port**] | **src-dst-ip** | **src-dst-mac** **src-dst-mixed-ip-port** **src-dst-portsrc-ip** | **src-mac** | **src-mixed-ip-port** | **src-port**}
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**port-channel load-balance** { **dst-ip** | **dst-mac** | **dst-mixed-ip-port** | **dst-port** | **extended** [**dst-ip** | **dst-mac** | **dst-port** | **ipv6-label** | **l3-proto** | **src-ip** | **src-mac** | **src-port**] | **src-dst-ip** | **src-dst-mac** **src-dst-mixed-ip-port** **src-dst-portsrc-ip** | **src-mac** | **src-mixed-ip-port** | **src-port**} **Example:** `Switch(config)# port-channel load-balance src-mac`	Configures an EtherChannel load-balancing method. The default is **src-mac**. Select one of these load-distribution methods: • **dst-ip**—Specifies destination-host IP address. • **dst-mac**—Specifies the destination-host MAC address of the incoming packet. • **dst-mixed-ip-port**—Specifies the host IP address and TCP/UDP port. • **dst-port**—Specifies the destination TCP/UDP port. • **extended**—Specifies extended load balance methods--combinations of source and destination methods beyond those available with the standard command. • **ipv6-label**—Specifies the IPv6 flow label. • **l3-proto**—Specifies the Layer 3 protocol. • **src-dst-ip**—Specifies the source and destination host IP address. • **src-dst-mac**—Specifies the source and destination host MAC address. • **src-dst-mixed-ip-port**—Specifies the source and destination host IP address and TCP/UDP port. • **src-dst-port**—Specifies the source and destination TCP/UDP port. • **src-ip**—Specifies the source host IP address. • **src-mac**—Specifies the source MAC address of the incoming packet. • **src-mixed-ip-port**—Specifies the source host IP address and TCP/UDP port.

	Command or Action	Purpose
		• **src-port**—Specifies the source TCP/UDP port.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Related Topics

Load-Balancing and Forwarding Methods, on page 1335
MAC Address Forwarding, on page 1335
IP Address Forwarding, on page 1336
Load-Balancing Advantages, on page 1336
EtherChannel Configuration Guidelines, on page 1339
Layer 2 EtherChannel Configuration Guidelines, on page 1340
Layer 3 EtherChannel Configuration Guidelines, on page 1340
Default EtherChannel Configuration, on page 1337

Configuring the PAgP Learn Method and Priority

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **pagp learn-method physical-port**
4. **pagp port-priority** *priority*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Switch(config)#` **`interface gigabitethernet 1/0/2`**	Specifies the port for transmission, and enters interface configuration mode.
Step 3	**pagp learn-method physical-port** **Example:** `Switch(config-if)#` **`pagp learn-method physical port`**	Selects the PAgP learning method. By default, **aggregation-port learning** is selected, which means the switch sends packets to the source by using any of the ports in the EtherChannel. With aggregate-port learning, it is not important on which physical port the packet arrives. Selects **physical-port** to connect with another switch that is a physical learner. Make sure to configure the **port-channel load-balance** global configuration command to **src-mac**. The learning method must be configured the same at both ends of the link.
Step 4	**pagp port-priority** *priority* **Example:** `Switch(config-if)#` **`pagp port-priority 200`**	Assigns a priority so that the selected port is chosen for packet transmission. For *priority*, the range is 0 to 255. The default is 128. The higher the priority, the more likely that the port will be used for PAgP transmission.
Step 5	**end** **Example:** `Switch(config-if)#` **`end`**	Returns to privileged EXEC mode.

Related Topics

PAgP Learn Method and Priority, on page 1332
EtherChannel Configuration Guidelines, on page 1339
Layer 2 EtherChannel Configuration Guidelines, on page 1340
Layer 3 EtherChannel Configuration Guidelines, on page 1340
Default EtherChannel Configuration, on page 1337
Monitoring EtherChannel, PAgP, and LACP Status, on page 1346

Monitoring EtherChannel, PAgP, and LACP Status

You can display EtherChannel, PAgP, and LACP status using the commands listed in this table.

Table 150: Commands for Monitoring EtherChannel, PAgP, and LACP Status

Command	Description
clear lacp { *channel-group-number* **counters** | **counters** }	Clears LACP channel-group information and traffic counters.
clear pagp { *channel-group-number* **counters** | **counters** }	Clears PAgP channel-group information and traffic counters.
show etherchannel [*channel-group-number* { **detail** | **port** | **port-channel** | **protocol** | **summary** }] [**detail** | **load-balance** | **port** | **port-channel** | **protocol** | **summary**]	Displays EtherChannel information in a brief, detailed, and one-line summary form. Also displays the load-balance or frame-distribution scheme, port, port-channel, and protocol information.
show pagp [*channel-group-number*] { **counters** | **internal** | **neighbor** }	Displays PAgP information such as traffic information, the internal PAgP configuration, and neighbor information.
show pagp [*channel-group-number*] **dual-active**	Displays the dual-active detection status.
show lacp [*channel-group-number*] { **counters** | **internal** | **neighbor** | **sys-id**}	Displays LACP information such as traffic information, the internal LACP configuration, and neighbor information.
show running-config	Verifies your configuration entries.
show etherchannel load-balance	Displays the load balance or frame distribution scheme among ports in the port channel.

Related Topics

Configuring the PAgP Learn Method and Priority , on page 1345
PAgP Learn Method and Priority, on page 1332

Configuration Examples for Configuring EtherChannels

Configuring Layer 2 EtherChannels: Examples

This example shows how to configure an EtherChannel on a single switch in the stack. It assigns two ports as static-access ports in VLAN 10 to channel 5 with the PAgP mode **desirable**:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet2/0/1 -2
Switch(config-if-range)# switchport mode access
Switch(config-if-range)# switchport access vlan 10
Switch(config-if-range)# channel-group 5 mode desirable non-silent
Switch(config-if-range)# end
```

This example shows how to configure an EtherChannel on a single switch in the stack. It assigns two ports as static-access ports in VLAN 10 to channel 5 with the LACP mode **active**:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet2/0/1 -2
Switch(config-if-range)# switchport mode access
Switch(config-if-range)# switchport access vlan 10
Switch(config-if-range)# channel-group 5 mode active
Switch(config-if-range)# end
```

This example shows how to configure a cross-stack EtherChannel. It uses LACP passive mode and assigns two ports on stack member 1 and one port on stack member 2 as static-access ports in VLAN 10 to channel 5:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet2/0/4 -5
Switch(config-if-range)# switchport mode access
Switch(config-if-range)# switchport access vlan 10
Switch(config-if-range)# channel-group 5 mode passive
Switch(config-if-range)# exit
Switch(config)# interface gigabitethernet3/0/3
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 10
Switch(config-if)# channel-group 5 mode passive
Switch(config-if)# exit
```

Configuring Port-Channel Logical Interfaces: Example

This example shows how to create the logical port channel 5 and assign 172.10.20.10 as its IP address:

```
Switch# configure terminal
Switch(config)# interface port-channel 5
Switch(config-if)# no switchport
Switch(config-if)# ip address 172.10.20.10 255.255.255.0
Switch(config-if)# end
```

Configuring EtherChannel Physical Interfaces: Examples

This example shows how to configure an EtherChannel. It assigns two ports to channel 5 with the LACP mode **active**:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet2/0/1 -2
Switch(config-if-range)# no ip address
Switch(config-if-range)# no switchport
Switch(config-if-range)# channel-group 5 mode active
Switch(config-if-range)# end
```

This example shows how to configure a cross-stack EtherChannel. It assigns two ports on stack member 2 and one port on stack member 3 to channel 7 using LACP active mode:

```
Switch# configure terminal
```

```
Switch(config)# interface range gigabitethernet2/0/4 -5
Switch(config-if-range)# no ip address
Switch(config-if-range)# no switchport
Switch(config-if-range)# channel-group 7 mode active
Switch(config-if-range)# exit
Switch(config)# interface gigabitethernet3/0/3
Switch(config-if)# no ip address
Switch(config-if)# no switchport
Switch(config-if)# channel-group 7 mode active
Switch(config-if)# exit
```

Additional References for EtherChannels

Related Documents

Related Topic	Document Title
Layer 2 command reference	*Layer 2/3 Command Reference (Catalyst 3850 Switches)*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for EtherChannels

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.
Cisco IOS XE 3.3SE	Support for the LACP max-bundle feature and the port channel min-links features was added.

CHAPTER 62

Configuring Flex Links and the MAC Address-Table Move Update Feature

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Restrictions for Configuring Flex Links and MAC Address-Table Move Update

- Flex Links are supported only on Layer 2 ports and port channels.
- You can configure up to 16 backup links.
- You can configure only one Flex Links backup link for any active link, and it must be a different interface from the active interface.

- An interface can belong to only one Flex Links pair. An interface can be a backup link for only one active link. An active link cannot belong to another Flex Links pair.
- Neither of the links can be a port that belongs to an EtherChannel. However, you can configure two port channels (EtherChannel logical interfaces) as Flex Links, and you can configure a port channel and a physical interface as Flex Links, with either the port channel or the physical interface as the active link.
- A backup link does not have to be the same type (Gigabit Ethernet or port channel) as the active link. However, you should configure both Flex Links with similar characteristics so that there are no loops or changes in behavior if the standby link begins to forward traffic.
- STP is disabled on Flex Links ports. A Flex Links port does not participate in STP, even if the VLANs present on the port are configured for STP. When STP is not enabled, be sure that there are no loops in the configured topology.
- You cannot have a switch stack containing a mix of Catalyst 3850 and Catalyst 3650 switches.

Related Topics

Configuring a Preemption Scheme for a Pair of Flex Links , on page 1358
Configuring Flex Links , on page 1356
Configuring Flex Links: Examples, on page 1364
Configuring VLAN Load Balancing on Flex Links , on page 1360
Configuring VLAN Load Balancing on Flex Links: Examples, on page 1364
Configuring a Switch to Obtain and Process MAC Address-Table Move Update Messages , on page 1362
Configuring MAC Address-Table Move Update , on page 1361
Configuring the MAC Address-Table Move Update: Examples, on page 1366

Information About Flex Links and MAC Address-Table Move Update

Flex Links

Flex Links are a pair of a Layer 2 interfaces (switch ports or port channels) where one interface is configured to act as a backup to the other. The feature provides an alternative solution to the Spanning Tree Protocol (STP). Users can disable STP and still retain basic link redundancy. Flex Links are typically configured in service provider or enterprise networks where customers do not want to run STP on the switch. If the switch is running STP, Flex Links are not necessary because STP already provides link-level redundancy or backup.

You configure Flex Links on one Layer 2 interface (the active link) by assigning another Layer 2 interface as the Flex Links or backup link. On switches, the Flex Links can be on the same switch or on another switch in the stack. When one of the links is up and forwarding traffic, the other link is in standby mode, ready to begin forwarding traffic if the other link shuts down. At any given time, only one of the interfaces is in the linkup state and forwarding traffic. If the primary link shuts down, the standby link starts forwarding traffic. When the active link comes back up, it goes into standby mode and does not forward traffic. STP is disabled on Flex Links interfaces.

Related Topics

Configuring a Preemption Scheme for a Pair of Flex Links , on page 1358
Configuring Flex Links , on page 1356

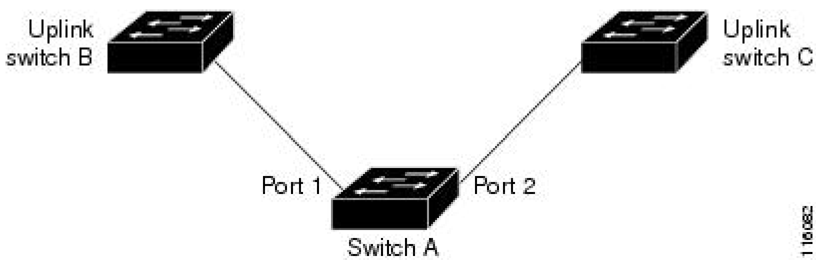
Configuring Flex Links: Examples, on page 1364

Flex Links Configuration

In the following figure, ports 1 and 2 on switch A are connected to uplink switches B and C. Because they are configured as Flex Links, only one of the interfaces is forwarding traffic; the other is in standby mode. If port 1 is the active link, it begins forwarding traffic between port 1 and switch B; the link between port 2 (the backup link) and switch C is not forwarding traffic. If port 1 goes down, port 2 comes up and starts forwarding traffic to switch C. When port 1 comes back up, it goes into standby mode and does not forward traffic; port 2 continues forwarding traffic.

You can also configure a preemption function, specifying the preferred port for forwarding traffic. For example, you can configure the Flex Links pair with preemption mode. In the scenario shown, when port 1 comes back up and has more bandwidth than port 2, port 1 begins forwarding traffic after 60 seconds. Port 2 becomes the standby port. You do this by entering the **switchport backup interface preemption mode bandwidth** and **switchport backup interface preemption delay** interface configuration commands.

Figure 64: Flex Links Configuration Example



If a primary (forwarding) link goes down, a trap notifies the network management stations. If the standby link goes down, a trap notifies the users.

Flex Links are supported only on Layer 2 ports and port channels, not on VLANs or on Layer 3 ports.

Related Topics

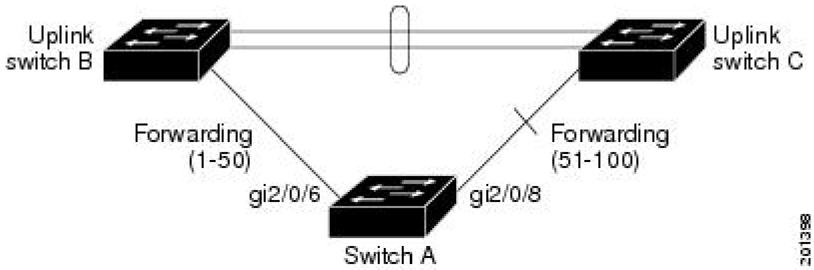
Configuring a Preemption Scheme for a Pair of Flex Links , on page 1358
Configuring Flex Links , on page 1356

VLAN Flex Links Load Balancing and Support

VLAN Flex Links load balancing allows users to configure a Flex Links pair so that both ports simultaneously forward the traffic for some mutually exclusive VLANs. For example, if Flex Links ports are configured for 1 to 100 VLANs, the traffic of the first 50 VLANs can be forwarded on one port and the rest on the other port. If one of the ports fail, the other active port forwards all the traffic. When the failed port comes back up, it resumes forwarding traffic in the preferred VLANs. In addition to providing the redundancy, this Flex Links pair can be used for load balancing. Flex Links VLAN load balancing does not impose any restrictions on uplink switches.

The following figure displays a VLAN Flex Links load-balancing configuration.

Figure 65: VLAN Flex Links Load-Balancing Configuration Example



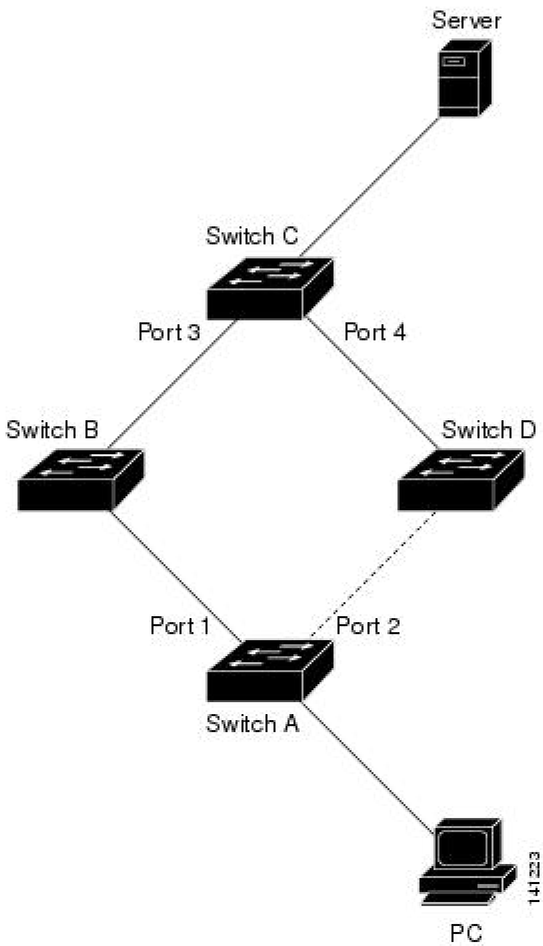
MAC Address-Table Move Update

The MAC address-table move update feature allows the switch to provide rapid bidirectional convergence when a primary (forwarding) link goes down and the standby link begins forwarding traffic.

In the following figure, switch A is an access switch, and ports 1 and 2 on switch A are connected to uplink switches B and D through a Flex Links pair. Port 1 is forwarding traffic, and port 2 is in the backup state.

Traffic from the PC to the server is forwarded from port 1 to port 3. The MAC address of the PC has been learned on port 3 of switch C. Traffic from the server to the PC is forwarded from port 3 to port 1.

Figure 66: MAC Address-Table Move Update Example



If the MAC address-table move update feature is not configured and port 1 goes down, port 2 starts forwarding traffic. However, for a short time, switch C keeps forwarding traffic from the server to the PC through port 3, and the PC does not get the traffic because port 1 is down. If switch C removes the MAC address of the PC on port 3 and relearns it on port 4, traffic can then be forwarded from the server to the PC through port 2.

If the MAC address-table move update feature is configured and enabled on the switches, and port 1 goes down, port 2 starts forwarding traffic from the PC to the server. The switch sends a MAC address-table move update packet from port 2. Switch C gets this packet on port 4 and immediately learns the MAC address of the PC on port 4, which reduces the reconvergence time.

You can configure the access switch, switch A, to *send* MAC address-table move update messages. You can also configure the uplink switches B, C, and D to *get* and process the MAC address-table move update messages. When switch C gets a MAC address-table move update message from switch A, switch C learns the MAC address of the PC on port 4. Switch C updates the MAC address table, including the forwarding table entry for the PC.

Switch A does not need to wait for the MAC address-table update. The switch detects a failure on port 1 and immediately starts forwarding server traffic from port 2, the new forwarding port. This change occurs in less than 100 milliseconds (ms). The PC is directly connected to switch A, and the connection status does not change. Switch A does not need to update the PC entry in the MAC address table.

Related Topics

Configuring a Switch to Obtain and Process MAC Address-Table Move Update Messages , on page 1362
Configuring MAC Address-Table Move Update , on page 1361
Configuring the MAC Address-Table Move Update: Examples, on page 1366

Flex Links VLAN Load Balancing Configuration Guidelines

- For Flex Links VLAN load balancing, you must choose the preferred VLANs on the backup interface.
- You cannot configure a preemption mechanism and VLAN load balancing for the same Flex Links pair.

Related Topics

Configuring VLAN Load Balancing on Flex Links , on page 1360
Configuring VLAN Load Balancing on Flex Links: Examples, on page 1364

Default Flex Links and MAC Address-Table Move Update Configuration

- Flex Links is not configured, and there are no backup interfaces defined.
- The preemption mode is off.
- The preemption delay is 35 seconds.
- The MAC address-table move update feature is not configured on the switch.

Related Topics

Configuring a Preemption Scheme for a Pair of Flex Links , on page 1358
Configuring Flex Links , on page 1356
Configuring Flex Links: Examples, on page 1364

How to Configure Flex Links and the MAC Address-Table Move Update Feature

Configuring Flex Links

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport backup interface** *interface-id*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(conf)# interface gigabitethernet1/0/1`	Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface).
Step 3	**switchport backup interface** *interface-id* **Example:** `Switch(conf-if)# switchport backup interface gigabitethernet1/0/2`	Configures a physical Layer 2 interface (or port channel) as part of a Flex Links pair with the interface. When one link is forwarding traffic, the other interface is in standby mode.
Step 4	**end** **Example:** `Switch(conf-if)# end`	Returns to privileged EXEC mode.

Related Topics

Flex Links, on page 1352
Default Flex Links and MAC Address-Table Move Update Configuration, on page 1356
Restrictions for Configuring Flex Links and MAC Address-Table Move Update, on page 1351
Configuring Flex Links: Examples, on page 1364
Flex Links Configuration, on page 1353
Monitoring Flex Links, Multicast Fast Convergence, and MAC Address-Table Move Update, on page 1363
Configuring Flex Links: Examples, on page 1364

Configuring a Preemption Scheme for a Pair of Flex Links

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport backup interface** *interface-id*
4. **switchport backup interface** *interface-id* **preemption mode** [**forced** | **bandwidth** | **off**]
5. **switchport backup interface** *interface-id* **preemption delay** *delay-time*
6. **end**
7. **show interface** [*interface-id*] **switchport backup**
8. **copy running-config startup config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode
Step 2	**interface** *interface-id* **Example:** `Switch(conf)# interface gigabitethernet1/0/1`	Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 128.
Step 3	**switchport backup interface** *interface-id* **Example:** `Switch(conf-if)# switchport backup interface gigabitethernet1/0/2`	Configures a physical Layer 2 interface (or port channel) as part of a Flex Links pair with the interface. When one link is forwarding traffic, the other interface is in standby mode.
Step 4	**switchport backup interface** *interface-id* **preemption mode** [**forced** | **bandwidth** | **off**] **Example:** `Switch(conf-if)# switchport backup interface gigabitethernet1/0/2 preemption mode forced`	Configures a preemption mechanism and delay for a Flex Links interface pair. You can configure the preemption as: • **forced—**(Optional) The active interface always preempts the backup. • **bandwidth—**(Optional) The interface with the higher bandwidth always acts as the active interface. • **off—**(Optional) No preemption occurs from active to backup.

	Command or Action	Purpose
Step 5	**switchport backup interface** *interface-id* **preemption delay** *delay-time* **Example:** `Switch(conf-if)#` **`switchport backup interface gigabitethernet1/0/2 preemption delay 50`**	Configures the time delay until a port preempts another port. **Note** Setting a delay time only works with forced and bandwidth modes.
Step 6	**end** **Example:** `Switch(conf-if)#` **`end`**	Returns to privileged EXEC mode.
Step 7	**show interface** [*interface-id*] **switchport backup** **Example:** `Switch#` **`show interface gigabitethernet1/0/2 switchport backup`**	Verifies the configuration.
Step 8	**copy running-config startup config** **Example:** `Switch#` **`copy running-config startup config`**	(Optional) Saves your entries in the switch startup configuration file.

Related Topics

Flex Links, on page 1352
Default Flex Links and MAC Address-Table Move Update Configuration, on page 1356
Restrictions for Configuring Flex Links and MAC Address-Table Move Update, on page 1351
Configuring Flex Links: Examples, on page 1364
Flex Links Configuration, on page 1353
Monitoring Flex Links, Multicast Fast Convergence, and MAC Address-Table Move Update, on page 1363
Configuring Flex Links: Examples, on page 1364

Configuring VLAN Load Balancing on Flex Links

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport backup interface** *interface-id* **prefer vlan** *vlan-range*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch (config)# interface gigabitethernet2/0/6`	Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 128.
Step 3	**switchport backup interface** *interface-id* **prefer vlan** *vlan-range* **Example:** `Switch (config-if)# switchport backup interface gigabitethernet2/0/8 prefer vlan 2`	Configures a physical Layer 2 interface (or port channel) as part of a Flex Links pair with the interface and specifies the VLANs carried on the interface. The VLAN ID range is 1 to 4094.
Step 4	**end** **Example:** `Switch (config-if)# end`	Returns to privileged EXEC mode.

Related Topics

Flex Links VLAN Load Balancing Configuration Guidelines, on page 1356
Restrictions for Configuring Flex Links and MAC Address-Table Move Update, on page 1351
Configuring VLAN Load Balancing on Flex Links: Examples, on page 1364
Configuring VLAN Load Balancing on Flex Links: Examples, on page 1364

Monitoring Flex Links, Multicast Fast Convergence, and MAC Address-Table Move Update, on page 1363

Configuring MAC Address-Table Move Update

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. Use one of the following:
   - **switchport backup interface** *interface-id*
   - **switchport backup interface** *interface-id* **mmu primary vlan** *vlan-id*
4. **end**
5. **mac address-table move update transmit**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch#interface gigabitethernet1/0/1`	Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 128.
Step 3	Use one of the following: • **switchport backup interface** *interface-id* • **switchport backup interface** *interface-id* **mmu primary vlan** *vlan-id* **Example:** `Switch(config-if)# switchport backup interface gigabitethernet0/2 mmu primary vlan 2`	Configures a physical Layer 2 interface (or port channel), as part of a Flex Links pair with the interface. The MAC address-table move update VLAN is the lowest VLAN ID on the interface. Configure a physical Layer 2 interface (or port channel) and specifies the VLAN ID on the interface, which is used for sending the MAC address-table move update. When one link is forwarding traffic, the other interface is in standby mode.

	Command or Action	Purpose
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to global configuration mode.

Step 5	**mac address-table move update transmit**
	
	Example:
	`Switch(config)#`
	`mac address-table move update transmit`
		Enables the access switch to send MAC address-table move updates to other switches in the network if the primary link goes down and the switch starts forwarding traffic through the standby link.

Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Related Topics

Configuring the MAC Address-Table Move Update: Examples, on page 1366
Monitoring Flex Links, Multicast Fast Convergence, and MAC Address-Table Move Update, on page 1363
MAC Address-Table Move Update, on page 1354
Restrictions for Configuring Flex Links and MAC Address-Table Move Update, on page 1351
Configuring the MAC Address-Table Move Update: Examples, on page 1366

Configuring a Switch to Obtain and Process MAC Address-Table Move Update Messages

SUMMARY STEPS

1. **configure terminal**
2. **mac address-table move update receive**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal**	Enters global configuration mode

	Command or Action	Purpose
	Example: `Switch# configure terminal`	
Step 2	**mac address-table move update receive** **Example:** `Switch (config)# mac address-table move update receive`	Enables the switch to obtain and processes the MAC address-table move updates.
Step 3	**end** **Example:** `Switch (config)# end`	Returns to privileged EXEC mode.

Related Topics

Monitoring Flex Links, Multicast Fast Convergence, and MAC Address-Table Move Update, on page 1363
Configuring the MAC Address-Table Move Update: Examples, on page 1366
MAC Address-Table Move Update, on page 1354
Restrictions for Configuring Flex Links and MAC Address-Table Move Update, on page 1351
Configuring the MAC Address-Table Move Update: Examples, on page 1366

Monitoring Flex Links, Multicast Fast Convergence, and MAC Address-Table Move Update

Command	Purpose
show interface [*interface-id*] **switchport backup**	Displays the Flex Links backup interface configured for an interface or all the configured Flex Links and the state of each active and backup interface (up or standby mode).
show ip igmp profile address-table move update *profile-id*	Displays the specified IGMP profile or all the IGMP profiles defined on the switch.
show mac address-table move update	Displays the MAC address-table move update information on the switch.

Related Topics

Configuring a Preemption Scheme for a Pair of Flex Links , on page 1358
Configuring Flex Links , on page 1356

Configuration Examples for Flex Links

Configuring Flex Links: Examples

This example shows how to verify the configuration after you configure an interface with a backup interface:

```
Switch# show interface switchport backup

Switch Backup Interface Pairs:
Active Interface Backup Interface State
------------------------------------------------------------------------
GigabitEthernet1/0/1 GigabitEthernet1/0/2 Active Up/Backup Standby
```

This example shows how to verify the configuration after you configure the preemption mode as forced for a backup interface pair:

```
Switch# show interface switchport backup detail

Switch Backup Interface Pairs:

Active Interface Backup Interface State
------------------------------------------------------------------------
GigabitEthernet1/0/211 GigabitEthernet1/0/2 Active Up/Backup Standby
Interface Pair : Gi1/0/1, Gi1/0/2
Preemption Mode : forced
Preemption Delay : 50 seconds
Bandwidth : 100000 Kbit (Gi1/0/1), 100000 Kbit (Gi1/0/2)
Mac Address Move Update Vlan : auto
```

Related Topics

Configuring a Preemption Scheme for a Pair of Flex Links , on page 1358
Configuring Flex Links , on page 1356
Flex Links, on page 1352
Default Flex Links and MAC Address-Table Move Update Configuration, on page 1356
Restrictions for Configuring Flex Links and MAC Address-Table Move Update, on page 1351
Configuring a Preemption Scheme for a Pair of Flex Links , on page 1358
Configuring Flex Links , on page 1356

Configuring VLAN Load Balancing on Flex Links: Examples

In the following example, VLANs 1 to 50, 60, and 100 to 120 are configured on the switch:

```
Switch(config)# interface gigabitethernet 2/0/6
Switch(config-if)# switchport backup interface gigabitethernet 2/0/8 prefer vlan 60,100-120
```

When both interfaces are up, Gi2/0/8 forwards traffic for VLANs 60 and 100 to 120 and Gi2/0/6 forwards traffic for VLANs 1 to 50.

```
Switch# show interfaces switchport backup

Switch Backup Interface Pairs:

Active Interface        Backup Interface        State
------------------------------------------------------------------------
GigabitEthernet2/0/6    GigabitEthernet2/0/8    Active Up/Backup Standby

Vlans Preferred on Active Interface: 1-50
Vlans Preferred on Backup Interface: 60, 100-120
```

When a Flex Links interface goes down (LINK_DOWN), VLANs preferred on this interface are moved to the peer interface of the Flex Links pair. In this example, if interface Gi2/0/6 goes down, Gi2/0/8 carries all VLANs of the Flex Links pair.

```
Switch# show interfaces switchport backup

Switch Backup Interface Pairs:

Active Interface        Backup Interface        State
------------------------------------------------------------------------
GigabitEthernet2/0/6    GigabitEthernet2/0/8    Active Down/Backup Up

Vlans Preferred on Active Interface: 1-50
Vlans Preferred on Backup Interface: 60, 100-120
```

When a Flex Links interface comes up, VLANs preferred on this interface are blocked on the peer interface and moved to the forwarding state on the interface that has just come up. In this example, if interface Gi2/0/6 comes up, VLANs preferred on this interface are blocked on the peer interface Gi2/0/8 and forwarded on Gi2/0/6.

```
Switch# show interfaces switchport backup

Switch Backup Interface Pairs:

Active Interface        Backup Interface        State
------------------------------------------------------------------------
GigabitEthernet2/0/6    GigabitEthernet2/0/8    Active Up/Backup Standby

Vlans Preferred on Active Interface: 1-50
Vlans Preferred on Backup Interface: 60, 100-120

Switch# show interfaces switchport backup detail

Switch Backup Interface Pairs:

Active Interface        Backup Interface        State
------------------------------------------------------------------------
FastEthernet1/0/3       FastEthernet1/0/4       Active Down/Backup Up

Vlans Preferred on Active Interface: 1-2,5-4094
Vlans Preferred on Backup Interface: 3-4
Preemption Mode  : off
Bandwidth : 10000 Kbit (Fa1/0/3), 100000 Kbit (Fa1/0/4)
Mac Address Move Update Vlan : auto
```

Related Topics

Configuring VLAN Load Balancing on Flex Links , on page 1360
Flex Links VLAN Load Balancing Configuration Guidelines, on page 1356
Restrictions for Configuring Flex Links and MAC Address-Table Move Update, on page 1351

Configuring VLAN Load Balancing on Flex Links , on page 1360

Configuring the MAC Address-Table Move Update: Examples

This example shows how to verify the configuration after you configure an access switch to send MAC address-table move updates:

```
Switch# show mac address-table move update

Switch-ID : 010b.4630.1780
Dst mac-address : 0180.c200.0010
Vlans/Macs supported : 1023/8320
Default/Current settings: Rcv Off/On, Xmt Off/On
Max packets per min : Rcv 40, Xmt 60
Rcv packet count : 5
Rcv conforming packet count : 5
Rcv invalid packet count : 0
Rcv packet count this min : 0
Rcv threshold exceed count : 0
Rcv last sequence# this min : 0
Rcv last interface : Po2
Rcv last src-mac-address : 000b.462d.c502
Rcv last switch-ID : 0403.fd6a.8700
Xmt packet count : 0
Xmt packet count this min : 0
Xmt threshold exceed count : 0
Xmt pak buf unavail cnt : 0
Xmt last interface : None
```

Related Topics

Configuring MAC Address-Table Move Update , on page 1361
Configuring a Switch to Obtain and Process MAC Address-Table Move Update Messages , on page 1362
Configuring a Switch to Obtain and Process MAC Address-Table Move Update Messages , on page 1362
Configuring MAC Address-Table Move Update , on page 1361
MAC Address-Table Move Update, on page 1354
Restrictions for Configuring Flex Links and MAC Address-Table Move Update, on page 1351

Additional References for Flex Links and MAC Address-Table Move Update

Related Documents

Related Topic	Document Title
Layer 2 command reference	*Layer 2/3 Command Reference (Catalyst 3850 Switches)*
switchport backup interface command	*Interface and Hardware Component Command Reference (Catalyst 3850 Switches)*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for Flex Links and MAC Address-Table Move Update

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.
Cisco IOS XE 3.3SE	Support for multicast fast convergence with Flex Links failover was added.

CHAPTER 63

Configuring UniDirectional Link Detection

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Restrictions for Configuring UDLD

The following are restrictions for configuring UniDirectional Link Detection (UDLD):

- A UDLD-capable port cannot detect a unidirectional link if it is connected to a UDLD-incapable port of another switch.
- When configuring the mode (normal or aggressive), make sure that the same mode is configured on both sides of the link.

Caution Loop guard works only on point-to-point links. We recommend that each end of the link has a directly connected device that is running STP.

Information About UDLD

UniDirectional Link Detection (UDLD) is a Layer 2 protocol that enables devices connected through fiber-optic or twisted-pair Ethernet cables to monitor the physical configuration of the cables and detect when a unidirectional link exists. All connected devices must support UDLD for the protocol to successfully identify and disable unidirectional links. When UDLD detects a unidirectional link, it disables the affected port and alerts you. Unidirectional links can cause a variety of problems, including spanning-tree topology loops.

Modes of Operation

UDLD supports two modes of operation: normal (the default) and aggressive. In normal mode, UDLD can detect unidirectional links due to misconnected ports on fiber-optic connections. In aggressive mode, UDLD can also detect unidirectional links due to one-way traffic on fiber-optic and twisted-pair links and to misconnected ports on fiber-optic links.

In normal and aggressive modes, UDLD works with the Layer 1 mechanisms to learn the physical status of a link. At Layer 1, autonegotiation takes care of physical signaling and fault detection. UDLD performs tasks that autonegotiation cannot perform, such as detecting the identities of neighbors and shutting down misconnected ports. When you enable both autonegotiation and UDLD, the Layer 1 and Layer 2 detections work together to prevent physical and logical unidirectional connections and the malfunctioning of other protocols.

A unidirectional link occurs whenever traffic sent by a local device is received by its neighbor but traffic from the neighbor is not received by the local device.

Normal Mode

In normal mode, UDLD detects a unidirectional link when fiber strands in a fiber-optic port are misconnected and the Layer 1 mechanisms do not detect this misconnection. If the ports are connected correctly but the traffic is one way, UDLD does not detect the unidirectional link because the Layer 1 mechanism, which is supposed to detect this condition, does not do so. In this case, the logical link is considered undetermined, and UDLD does not disable the port.

When UDLD is in normal mode, if one of the fiber strands in a pair is disconnected, as long as autonegotiation is active, the link does not stay up because the Layer 1 mechanisms detects a physical problem with the link. In this case, UDLD does not take any action and the logical link is considered undetermined.

Related Topics

Enabling UDLD Globally , on page 1373
Enabling UDLD on an Interface , on page 1374

Aggressive Mode

In aggressive mode, UDLD detects a unidirectional link by using the previous detection methods. UDLD in aggressive mode can also detect a unidirectional link on a point-to-point link on which no failure between the two devices is allowed. It can also detect a unidirectional link when one of these problems exists:

- On fiber-optic or twisted-pair links, one of the ports cannot send or receive traffic.
- On fiber-optic or twisted-pair links, one of the ports is down while the other is up.
- One of the fiber strands in the cable is disconnected.

In these cases, UDLD disables the affected port.

In a point-to-point link, UDLD hello packets can be considered as a heart beat whose presence guarantees the health of the link. Conversely, the loss of the heart beat means that the link must be shut down if it is not possible to reestablish a bidirectional link.

If both fiber strands in a cable are working normally from a Layer 1 perspective, UDLD in aggressive mode detects whether those fiber strands are connected correctly and whether traffic is flowing bidirectionally between the correct neighbors. This check cannot be performed by autonegotiation because autonegotiation operates at Layer 1.

Related Topics

Enabling UDLD Globally , on page 1373
Enabling UDLD on an Interface , on page 1374

Methods to Detect Unidirectional Links

UDLD operates by using two methods:

- Neighbor database maintenance
- Event-driven detection and echoing

Related Topics

Enabling UDLD Globally , on page 1373
Enabling UDLD on an Interface , on page 1374

Neighbor Database Maintenance

UDLD learns about other UDLD-capable neighbors by periodically sending a hello packet (also called an advertisement or probe) on every active port to keep each device informed about its neighbors.

When the switch receives a hello message, it caches the information until the age time (hold time or time-to-live) expires. If the switch receives a new hello message before an older cache entry ages, the switch replaces the older entry with the new one.

Whenever a port is disabled and UDLD is running, whenever UDLD is disabled on a port, or whenever the switch is reset, UDLD clears all existing cache entries for the ports affected by the configuration change. UDLD sends at least one message to inform the neighbors to flush the part of their caches affected by the status change. The message is intended to keep the caches synchronized.

Event-Driven Detection and Echoing

UDLD relies on echoing as its detection operation. Whenever a UDLD device learns about a new neighbor or receives a resynchronization request from an out-of-sync neighbor, it restarts the detection window on its side of the connection and sends echo messages in reply. Because this behavior is the same on all UDLD neighbors, the sender of the echoes expects to receive an echo in reply.

If the detection window ends and no valid reply message is received, the link might shut down, depending on the UDLD mode. When UDLD is in normal mode, the link might be considered undetermined and might not be shut down. When UDLD is in aggressive mode, the link is considered unidirectional, and the port is disabled.

Related Topics

Enabling UDLD Globally , on page 1373
Enabling UDLD on an Interface , on page 1374

UDLD Reset Options

If an interface becomes disabled by UDLD, you can use one of the following options to reset UDLD:

- The **udld reset** interface configuration command.
- The **shutdown** interface configuration command followed by the **no shutdown** interface configuration command restarts the disabled port.
- The **no udld** {**aggressive** | **enable**} global configuration command followed by the **udld** {**aggressive** | **enable**} global configuration command reenables the disabled ports.
- The **no udld port** interface configuration command followed by the **udld port** [**aggressive**] interface configuration command reenables the disabled fiber-optic port.
- The **errdisable recovery cause udld** global configuration command enables the timer to automatically recover from the UDLD error-disabled state, and the **errdisable recovery interval** *interval* global configuration command specifies the time to recover from the UDLD error-disabled state.

Related Topics

Enabling UDLD Globally , on page 1373
Enabling UDLD on an Interface , on page 1374

Default UDLD Configuration

Table 151: Default UDLD Configuration

Feature	Default Setting
UDLD global enable state	Globally disabled
UDLD per-port enable state for fiber-optic media	Disabled on all Ethernet fiber-optic ports
UDLD per-port enable state for twisted-pair (copper) media	Disabled on all Ethernet 10/100 and 1000BASE-TX ports

Feature	Default Setting
UDLD aggressive mode	Disabled

Related Topics

Enabling UDLD Globally , on page 1373
Enabling UDLD on an Interface , on page 1374

How to Configure UDLD

Enabling UDLD Globally

Follow these steps to enable UDLD in the aggressive or normal mode and to set the configurable message timer on all fiber-optic ports on the switch.

SUMMARY STEPS

1. **configure terminal**
2. **udld** {**aggressive** | **enable** | **message time** *message-timer-interval*}
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**udld** {**aggressive** | **enable** | **message time** *message-timer-interval*} **Example:** `Switch(config)# udld enable message time 10`	Specifies the UDLD mode of operation: • **aggressive**—Enables UDLD in aggressive mode on all fiber-optic ports. • **enable**—Enables UDLD in normal mode on all fiber-optic ports on the switch. UDLD is disabled by default. An individual interface configuration overrides the setting of the **udld enable** global configuration command. • **message time** *message-timer-interval*—Configures the period of time between UDLD probe messages on ports that are in the advertisement phase and are detected to be bidirectional. The range is from 1 to 90 seconds; the default value is 15.

	Command or Action	Purpose
		Note This command affects fiber-optic ports only. Use the **udld** interface configuration command to enable UDLD on other port types. Use the **no** form of this command, to disable UDLD.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Related Topics

Monitoring and Maintaing UDLD
Aggressive Mode, on page 1371

Normal Mode, on page 1370

Methods to Detect Unidirectional Links, on page 1371
Event-Driven Detection and Echoing, on page 1372

UDLD Reset Options, on page 1372

Default UDLD Configuration, on page 1372

Enabling UDLD on an Interface

Follow these steps either to enable UDLD in the aggressive or normal mode or to disable UDLD on a port.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **udld port** [**aggressive**]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Switch(config)#` **`interface gigabitethernet 1/0/1`**	Specifies the port to be enabled for UDLD, and enters interface configuration mode.
Step 3	**udld port** [**aggressive**] **Example:** `Switch(config-if)#` **`udld port aggressive`**	UDLD is disabled by default. • **udld port**—Enables UDLD in normal mode on the specified port. • **udld port aggressive**—(Optional) Enables UDLD in aggressive mode on the specified port. **Note** Use the **no udld port** interface configuration command to disable UDLD on a specified fiber-optic port.
Step 4	**end** **Example:** `Switch(config-if)#` **`end`**	Returns to privileged EXEC mode.

Related Topics

Monitoring and Maintaing UDLD
Aggressive Mode, on page 1371

Normal Mode, on page 1370

Methods to Detect Unidirectional Links, on page 1371
Event-Driven Detection and Echoing, on page 1372

UDLD Reset Options, on page 1372

Default UDLD Configuration, on page 1372

Monitoring and Maintaining UDLD

Command	Purpose
show udld [*interface-id* | **neighbors**]	Displays the UDLD status for the specified port or for all ports.

Additional References for UDLD

Related Documents

Related Topic	Document Title
Layer 2 command reference	*Layer 2/3 Command Reference (Catalyst 3850 Switches)*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for UDLD

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

PART IX

IPv6

CHAPTER 64

Configuring MLD Snooping

This module contains details of configuring MLD snooping

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information About Configuring IPv6 MLD Snooping

You can use Multicast Listener Discovery (MLD) snooping to enable efficient distribution of IP Version 6 (IPv6) multicast data to clients and routers in a switched network on the switch. Unless otherwise noted, the term switch refers to a standalone switch and to a switch stack.

> **Note** To use IPv6 on a Catalyst 2960-XR switch, you must configure the dual IPv4 and IPv6 Switch Database Management (SDM) template on the switch.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release or the Cisco IOS documentation referenced in the procedures.

Understanding MLD Snooping

In IP Version 4 (IPv4), Layer 2 switches can use Internet Group Management Protocol (IGMP) snooping to limit the flooding of multicast traffic by dynamically configuring Layer 2 interfaces so that multicast traffic is forwarded to only those interfaces associated with IP multicast devices. In IPv6, MLD snooping performs a similar function. With MLD snooping, IPv6 multicast data is selectively forwarded to a list of ports that want to receive the data, instead of being flooded to all ports in a VLAN. This list is constructed by snooping IPv6 multicast control packets.

MLD is a protocol used by IPv6 multicast routers to discover the presence of multicast listeners (nodes wishing to receive IPv6 multicast packets) on the links that are directly attached to the routers and to discover which multicast packets are of interest to neighboring nodes. MLD is derived from IGMP; MLD Version 1 (MLDv1) is equivalent to IGMPv2, and MLD Version 2 (MLDv2) is equivalent to IGMPv3. MLD is a subprotocol of Internet Control Message Protocol Version 6 (ICMPv6), and MLD messages are a subset of ICMPv6 messages, identified in IPv6 packets by a preceding Next Header value of 58.

The switch supports two versions of MLD snooping:

- MLDv1 snooping detects MLDv1 control packets and sets up traffic bridging based on IPv6 destination multicast addresses.
- MLDv2 basic snooping (MBSS) uses MLDv2 control packets to set up traffic forwarding based on IPv6 destination multicast addresses.

The switch can snoop on both MLDv1 and MLDv2 protocol packets and bridge IPv6 multicast data based on destination IPv6 multicast addresses.

Note The switch does not support MLDv2 enhanced snooping, which sets up IPv6 source and destination multicast address-based forwarding.

MLD snooping can be enabled or disabled globally or per VLAN. When MLD snooping is enabled, a per-VLAN IPv6 multicast address table is constructed in software and hardware. The switch then performs IPv6 multicast-address based bridging in hardware.

MLD Messages

MLDv1 supports three types of messages:

- Listener Queries are the equivalent of IGMPv2 queries and are either General Queries or Multicast-Address-Specific Queries (MASQs).
- Multicast Listener Reports are the equivalent of IGMPv2 reports
- Multicast Listener Done messages are the equivalent of IGMPv2 leave messages.

MLDv2 supports MLDv2 queries and reports, as well as MLDv1 Report and Done messages.

Message timers and state transitions resulting from messages being sent or received are the same as those of IGMPv2 messages. MLD messages that do not have valid link-local IPv6 source addresses are ignored by MLD routers and switches.

MLD Queries

The switch sends out MLD queries, constructs an IPv6 multicast address database, and generates MLD group-specific and MLD group-and-source-specific queries in response to MLD Done messages. The switch also supports report suppression, report proxying, Immediate-Leave functionality, and static IPv6 multicast group address configuration.

When MLD snooping is disabled, all MLD queries are flooded in the ingress VLAN.

When MLD snooping is enabled, received MLD queries are flooded in the ingress VLAN, and a copy of the query is sent to the CPU for processing. From the received query, MLD snooping builds the IPv6 multicast address database. It detects multicast router ports, maintains timers, sets report response time, learns the querier IP source address for the VLAN, learns the querier port in the VLAN, and maintains multicast-address aging.

Note When the IPv6 multicast router is a Catalyst 6500 switch and you are using extended VLANs (in the range 1006 to 4094), IPv6 MLD snooping must be enabled on the extended VLAN on the Catalyst 6500 switch in order for the Catalyst 2960, 2960-S, 2960-C, or 2960-X switch to receive queries on the VLAN. For normal-range VLANs (1 to 1005), it is not necessary to enable IPv6 MLD snooping on the VLAN on the Catalyst 6500 switch.

When a group exists in the MLD snooping database, the switch responds to a group-specific query by sending an MLDv1 report. When the group is unknown, the group-specific query is flooded to the ingress VLAN.

When a host wants to leave a multicast group, it can send out an MLD Done message (equivalent to IGMP Leave message). When the switch receives an MLDv1 Done message, if Immediate- Leave is not enabled, the switch sends an MASQ to the port from which the message was received to determine if other devices connected to the port should remain in the multicast group.

Multicast Client Aging Robustness

You can configure port membership removal from addresses based on the number of queries. A port is removed from membership to an address only when there are no reports to the address on the port for the configured number of queries. The default number is 2.

Multicast Router Discovery

Like IGMP snooping, MLD snooping performs multicast router discovery, with these characteristics:

- Ports configured by a user never age out.
- Dynamic port learning results from MLDv1 snooping queries and IPv6 PIMv2 packets.
- If there are multiple routers on the same Layer 2 interface, MLD snooping tracks a single multicast router on the port (the router that most recently sent a router control packet).
- Dynamic multicast router port aging is based on a default timer of 5 minutes; the multicast router is deleted from the router port list if no control packet is received on the port for 5 minutes.
- IPv6 multicast router discovery only takes place when MLD snooping is enabled on the switch.

- Received IPv6 multicast router control packets are always flooded to the ingress VLAN, whether or not MLD snooping is enabled on the switch.
- After the discovery of the first IPv6 multicast router port, unknown IPv6 multicast data is forwarded only to the discovered router ports (before that time, all IPv6 multicast data is flooded to the ingress VLAN).

MLD Reports

The processing of MLDv1 join messages is essentially the same as with IGMPv2. When no IPv6 multicast routers are detected in a VLAN, reports are not processed or forwarded from the switch. When IPv6 multicast routers are detected and an MLDv1 report is received, an IPv6 multicast group address is entered in the VLAN MLD database. Then all IPv6 multicast traffic to the group within the VLAN is forwarded using this address. When MLD snooping is disabled, reports are flooded in the ingress VLAN.

When MLD snooping is enabled, MLD report suppression, called listener message suppression, is automatically enabled. With report suppression, the switch forwards the first MLDv1 report received by a group to IPv6 multicast routers; subsequent reports for the group are not sent to the routers. When MLD snooping is disabled, report suppression is disabled, and all MLDv1 reports are flooded to the ingress VLAN.

The switch also supports MLDv1 proxy reporting. When an MLDv1 MASQ is received, the switch responds with MLDv1 reports for the address on which the query arrived if the group exists in the switch on another port and if the port on which the query arrived is not the last member port for the address.

MLD Done Messages and Immediate-Leave

When the Immediate-Leave feature is enabled and a host sends an MLDv1 Done message (equivalent to an IGMP leave message), the port on which the Done message was received is immediately deleted from the group.You enable Immediate-Leave on VLANs and (as with IGMP snooping), you should only use the feature on VLANs where a single host is connected to the port. If the port was the last member of a group, the group is also deleted, and the leave information is forwarded to the detected IPv6 multicast routers.

When Immediate Leave is not enabled in a VLAN (which would be the case when there are multiple clients for a group on the same port) and a Done message is received on a port, an MASQ is generated on that port. The user can control when a port membership is removed for an existing address in terms of the number of MASQs. A port is removed from membership to an address when there are no MLDv1 reports to the address on the port for the configured number of queries.

The number of MASQs generated is configured by using the **ipv6 mld snooping last-listener-query count** global configuration command. The default number is 2.

The MASQ is sent to the IPv6 multicast address for which the Done message was sent. If there are no reports sent to the IPv6 multicast address specified in the MASQ during the switch maximum response time, the port on which the MASQ was sent is deleted from the IPv6 multicast address database. The maximum response time is the time configured by using the **ipv6 mld snooping last-listener-query-interval** global configuration command. If the deleted port is the last member of the multicast address, the multicast address is also deleted, and the switch sends the address leave information to all detected multicast routers.

Topology Change Notification Processing

When topology change notification (TCN) solicitation is enabled by using the **ipv6 mld snooping tcn query solicit** global configuration command, MLDv1 snooping sets the VLAN to flood all IPv6 multicast traffic with a configured number of MLDv1 queries before it begins sending multicast data only to selected ports.

You set this value by using the **ipv6 mld snooping tcn flood query count** global configuration command. The default is to send two queries. The switch also generates MLDv1 global Done messages with valid link-local IPv6 source addresses when the switch becomes the STP root in the VLAN or when it is configured by the user. This is same as done in IGMP snooping.

MLD Snooping in Switch Stacks

The MLD IPv6 group address databases are maintained on all switches in the stack, regardless of which switch learns of an IPv6 multicast group. Report suppression and proxy reporting are done stack-wide. During the maximum response time, only one received report for a group is forwarded to the multicast routers, regardless of which switch the report arrives on.

The election of a new stack master does not affect the learning or bridging of IPv6 multicast data; bridging of IPv6 multicast data does not stop during a stack master re-election. When a new switch is added to the stack, it synchronizes the learned IPv6 multicast information from the stack master. Until the synchronization is complete, data ingress on the newly added switch is treated as unknown multicast data.

How to Configure IPv6 MLD Snooping

Default MLD Snooping Configuration

Table 152: Default MLD Snooping Configuration

Feature	Default Setting
MLD snooping (Global)	Disabled.
MLD snooping (per VLAN)	Enabled. MLD snooping must be globally enabled for VLAN MLD snooping to take place.
IPv6 Multicast addresses	None configured.
IPv6 Multicast router ports	None configured.
MLD snooping Immediate Leave	Disabled.
MLD snooping robustness variable	Global: 2; Per VLAN: 0. **Note** The VLAN value overrides the global setting. When the VLAN value is 0, the VLAN uses the global count.
Last listener query count	Global: 2; Per VLAN: 0. **Note** The VLAN value overrides the global setting. When the VLAN value is 0, the VLAN uses the global count.
Last listener query interval	Global: 1000 (1 second); VLAN: 0. **Note** The VLAN value overrides the global setting. When the VLAN value is 0, the VLAN uses the global interval.
TCN query solicit	Disabled.

Feature	Default Setting
TCN query count	2.
MLD listener suppression	

MLD Snooping Configuration Guidelines

When configuring MLD snooping, consider these guidelines:

- You can configure MLD snooping characteristics at any time, but you must globally enable MLD snooping by using the **ipv6 mld snooping** global configuration command for the configuration to take effect.
- When the IPv6 multicast router is a Catalyst 6500 switch and you are using extended VLANs (in the range 1006 to 4094), IPv6 MLD snooping must be enabled on the extended VLAN on the Catalyst 6500 switch in order for the switch to receive queries on the VLAN. For normal-range VLANs (1 to 1005), it is not necessary to enable IPv6 MLD snooping on the VLAN on the Catalyst 6500 switch.
- MLD snooping and IGMP snooping act independently of each other. You can enable both features at the same time on the switch.
-

Enabling or Disabling MLD Snooping on the Switch

By default, IPv6 MLD snooping is globally disabled on the switch and enabled on all VLANs. When MLD snooping is globally disabled, it is also disabled on all VLANs. When you globally enable MLD snooping, the VLAN configuration overrides the global configuration. That is, MLD snooping is enabled only on VLAN interfaces in the default state (enabled).

You can enable and disable MLD snooping on a per-VLAN basis or for a range of VLANs, but if you globally disable MLD snooping, it is disabled in all VLANs. If global snooping is enabled, you can enable or disable VLAN snooping.

Beginning in privileged EXEC mode, follow these steps to globally enable MLD snooping on the switch:

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**ipv6 mld snooping** **Example:** `Switch(config)# ipv6 mld snooping`	Enables MLD snooping on the switch.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**copy running-config startup-config** **Example:** `Switch(config)# copy running-config startup-config`	(Optional) Save your entries in the configuration file.
Step 5	**reload** **Example:** `Switch(config)# reload`	Reload the operating system.

Enabling or Disabling MLD Snooping on a VLAN

Beginning in privileged EXEC mode, follow these steps to enable MLD snooping on a VLAN.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ipv6 mld snooping** **Example:** `Switch(config)# ipv6 mld snooping`	Enables MLD snooping on the switch.

	Command or Action	Purpose
Step 3	**ipv6 mld snooping vlan** *vlan-id* **Example:** `Switch(config)# ipv6 mld snooping vlan 1`	Enables MLD snooping on the VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094. **Note** MLD snooping must be globally enabled for VLAN snooping to be enabled.
Step 4	**end** **Example:** `Switch(config)# ipv6 mld snooping vlan 1`	Returns to privileged EXEC mode.

Configuring a Static Multicast Group

Hosts or Layer 2 ports normally join multicast groups dynamically, but you can also statically configure an IPv6 multicast address and member ports for a VLAN.

Beginning in privileged EXEC mode, follow these steps to add a Layer 2 port as a member of a multicast group:

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode
Step 2	**ipv6 mld snooping vlan** *vlan-id* **static** *ipv6_multicast_address* **interface** *interface-id* **Example:** `Switch(config)# ipv6 mld snooping vlan 1 static FF12::3 interface gigabitethernet 0/1`	Configures a multicast group with a Layer 2 port as a member of a multicast group: • *vlan-id* is the multicast group VLAN ID. The VLAN ID range is 1 to 1001 and 1006 to 4094. • *ipv6_multicast_address* is the 128-bit group IPv6 address. The address must be in the form specified in RFC 2373. • *interface-id* is the member port. It can be a physical interface or a port channel (1 to 48).

	Command or Action	Purpose
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Step 4	Use one of the following:
	• **show ipv6 mld snooping address**
	• **show ipv6 mld snooping address vlan** *vlan-id*
	
	Example:
	`Switch# show ipv6 mld snooping address`
	or
	`Switch# show ipv6 mld snooping vlan 1`
		Verifies the static member port and the IPv6 address.

Configuring a Multicast Router Port

Note Static connections to multicast routers are supported only on switch ports.

Beginning in privileged EXEC mode, follow these steps to add a multicast router port to a VLAN:

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ipv6 mld snooping vlan** *vlan-id* **mrouter interface** *interface-id* **Example:** `Switch(config)# ipv6 mld snooping vlan 1 mrouter interface gigabitethernet 0/2`	Specifies the multicast router VLAN ID, and specify the interface to the multicast router. • The VLAN ID range is 1 to 1001 and 1006 to 4094. • The interface can be a physical interface or a port channel. The port-channel range is 1 to 48.

	Command or Action	Purpose
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show ipv6 mld snooping mrouter** [**vlan** *vlan-id*] **Example:** `Switch# show ipv6 mld snooping mrouter vlan 1`	Verifies that IPv6 MLD snooping is enabled on the VLAN interface.

Enabling MLD Immediate Leave

Beginning in privileged EXEC mode, follow these steps to enable MLDv1 Immediate Leave:

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ipv6 mld snooping vlan** *vlan-id* **immediate-leave** **Example:** `Switch(config)# ipv6 mld snooping vlan 1 immediate-leave`	Enables MLD Immediate Leave on the VLAN interface.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show ipv6 mld snooping vlan** *vlan-id* **Example:** `Switch# show ipv6 mld snooping vlan 1`	Verifies that Immediate Leave is enabled on the VLAN interface.

Configuring MLD Snooping Queries

Beginning in privileged EXEC mode, follow these steps to configure MLD snooping query characteristics for the switch or for a VLAN:

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ipv6 mld snooping robustness-variable** *value* **Example:** `Switch(config)# ipv6 mld snooping robustness-variable 3`	(Optional) Sets the number of queries that are sent before switch will deletes a listener (port) that does not respond to a general query. The range is 1 to 3; the default is 2.
Step 3	**ipv6 mld snooping vlan** *vlan-id* **robustness-variable** *value* **Example:** `Switch(config)# ipv6 mld snooping vlan 1 robustness-variable 3`	(Optional) Sets the robustness variable on a VLAN basis, which determines the number of general queries that MLD snooping sends before aging out a multicast address when there is no MLD report response. The range is 1 to 3; the default is 0. When set to 0, the number used is the global robustness variable value.
Step 4	**ipv6 mld snooping last-listener-query-count** *count* **Example:** `Switch(config)# ipv6 mld snooping last-listener-query-count 7`	(Optional) Sets the number of MASQs that the switch sends before aging out an MLD client. The range is 1 to 7; the default is 2. The queries are sent 1 second apart.
Step 5	**ipv6 mld snooping vlan** *vlan-id* **last-listener-query-count** *count* **Example:** `Switch(config)# ipv6 mld snooping vlan 1 last-listener-query-count 7`	(Optional) Sets the last-listener query count on a VLAN basis. This value overrides the value configured globally. The range is 1 to 7; the default is 0. When set to 0, the global count value is used. Queries are sent 1 second apart.
Step 6	**ipv6 mld snooping last-listener-query-interval** *interval* **Example:** `Switch(config)# ipv6 mld snooping last-listener-query-interval 2000`	(Optional) Sets the maximum response time that the switch waits after sending out a MASQ before deleting a port from the multicast group. The range is 100 to 32,768 thousands of a second. The default is 1000 (1 second).
Step 7	**ipv6 mld snooping vlan** *vlan-id* **last-listener-query-interval** *interval*	(Optional) Sets the last-listener query interval on a VLAN basis. This value overrides the value configured globally. The range is 0

	Command or Action	Purpose
	Example: `Switch(config)# ipv6 mld snooping vlan 1 last-listener-query-interval 2000`	to 32,768 thousands of a second. The default is 0. When set to 0, the global last-listener query interval is used.
Step 8	**ipv6 mld snooping tcn query solicit** **Example:** `Switch(config)# ipv6 mld snooping tcn query solicit`	(Optional) Enables topology change notification (TCN) solicitation, which means that VLANs flood all IPv6 multicast traffic for the configured number of queries before sending multicast data to only those ports requesting to receive it. The default is for TCN to be disabled.
Step 9	**ipv6 mld snooping tcn flood query count** *count* **Example:** `Switch(config)# ipv6 mld snooping tcn flood query count 5`	(Optional) When TCN is enabled, specifies the number of TCN queries to be sent. The range is from 1 to 10; the default is 2.
Step 10	**end**	Returns to privileged EXEC mode.
Step 11	**show ipv6 mld snooping querier** [**vlan** *vlan-id*] **Example:** `Switch(config)# show ipv6 mld snooping querier vlan 1`	(Optional) Verifies that the MLD snooping querier information for the switch or for the VLAN.

Disabling MLD Listener Message Suppression

MLD snooping listener message suppression is enabled by default. When it is enabled, the switch forwards only one MLD report per multicast router query. When message suppression is disabled, multiple MLD reports could be forwarded to the multicast routers.

Beginning in privileged EXEC mode, follow these steps to disable MLD listener message suppression:

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enter global configuration mode.

	Command or Action	Purpose
Step 2	**no ipv6 mld snooping listener-message-suppression** **Example:** `Switch(config)# no ipv6 mld snooping listener-message-suppression`	Disable MLD message suppression.
Step 3	**end** **Example:** `Switch(config)# end`	Return to privileged EXEC mode.
Step 4	**show ipv6 mld snooping** **Example:** `Switch# show ipv6 mld snooping`	Verify that IPv6 MLD snooping report suppression is disabled.

Displaying MLD Snooping Information

You can display MLD snooping information for dynamically learned and statically configured router ports and VLAN interfaces. You can also display IPv6 group address multicast entries for a VLAN configured for MLD snooping.

Table 153: Commands for Displaying MLD Snooping Information

Command	Purpose
show ipv6 mld snooping [**vlan** *vlan-id*]	Displays the MLD snooping configuration information for all VLANs on the switch or for a specified VLAN. (Optional) Enter **vlan** *vlan-id* to display information for a single VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094.
show ipv6 mld snooping mrouter [**vlan** *vlan-id*]	Displays information on dynamically learned and manually configured multicast router interfaces. When you enable MLD snooping, the switch automatically learns the interface to which a multicast router is connected. These are dynamically learned interfaces. (Optional) Enters **vlan** *vlan-id* to display information for a single VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094.

Command	Purpose
show ipv6 mld snooping querier [**vlan** *vlan-id*]	Displays information about the IPv6 address and incoming port for the most-recently received MLD query messages in the VLAN. (Optional) Enters **vlan** *vlan-id* to display information for a single VLAN.The VLAN ID range is 1 to 1001 and 1006 to 4094.
show ipv6 mld snooping address [**vlan** *vlan-id*] [**count** | **dynamic** | **user**]	Displays all IPv6 multicast address information or specific IPv6 multicast address information for the switch or a VLAN. • Enters **count** to show the group count on the switch or in a VLAN. • Enters **dynamic** to display MLD snooping learned group information for the switch or for a VLAN. • Entesr **user** to display MLD snooping user-configured group information for the switch or for a VLAN.
show ipv6 mld snooping address vlan *vlan-id* [*ipv6-multicast-address*]	Displays MLD snooping for the specified VLAN and IPv6 multicast address.

Configuration Examples for Configuring MLD Snooping

Configuring a Static Multicast Group: Example

This example shows how to statically configure an IPv6 multicast group:

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping vlan 2 static FF12::3 interface gigabitethernet

          1/0/1
Switch(config)# end
```

Configuring a Multicast Router Port: Example

This example shows how to add a multicast router port to VLAN 200:

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping vlan 200 mrouter interface gigabitethernet

          0/2
Switch(config)# exit
```

Enabling MLD Immediate Leave: Example

This example shows how to enable MLD Immediate Leave on VLAN 130:

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping vlan 130 immediate-leave
Switch(config)# exit
```

Configuring MLD Snooping Queries: Example

This example shows how to set the MLD snooping global robustness variable to 3:

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping robustness-variable 3
Switch(config)# exit
```

This example shows how to set the MLD snooping last-listener query count for a VLAN to 3:

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping vlan 200 last-listener-query-count 3
Switch(config)# exit
```

This example shows how to set the MLD snooping last-listener query interval (maximum response time) to 2000 (2 seconds):

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping last-listener-query-interval 2000
Switch(config)# exit
```

CHAPTER 65

Configuring IPv6 Unicast Routing

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information About Configuring IPv6 Unicast Routing

This chapter describes how to configure IPv6 unicast routing on the switch.

Note To use all IPv6 features in this chapter, the switch or stack master must be running the IP services feature set. Switches running the IP base feature set support IPv6 static routing, RIP for IPv6, and OSPF. Switches running the LAN base feature set support only IPv6 host functionality.

Understanding IPv6

IPv4 users can move to IPv6 and receive services such as end-to-end security, quality of service (QoS), and globally unique addresses. The IPv6 address space reduces the need for private addresses and Network Address Translation (NAT) processing by border routers at network edges.

For information about how Cisco Systems implements IPv6, go to:

http://www.cisco.com/en/US/products/ps6553/products_ios_technology_home.html

For information about IPv6 and other features in this chapter

- See the *Cisco IOS IPv6 Configuration Library*.
- Use the Search field on Cisco.com to locate the Cisco IOS software documentation. For example, if you want information about static routes, you can enter *Implementing Static Routes for IPv6* in the search field to learn about static routes.

IPv6 Addresses

The switch supports only IPv6 unicast addresses. It does not support site-local unicast addresses, or anycast addresses.

The IPv6 128-bit addresses are represented as a series of eight 16-bit hexadecimal fields separated by colons in the format: n:n:n:n:n:n:n:n. This is an example of an IPv6 address:

2031:0000:130F:0000:0000:09C0:080F:130B

For easier implementation, leading zeros in each field are optional. This is the same address without leading zeros:

2031:0:130F:0:0:9C0:80F:130B

You can also use two colons (::) to represent successive hexadecimal fields of zeros, but you can use this short version only once in each address:

2031:0:130F::09C0:080F:130B

For more information about IPv6 address formats, address types, and the IPv6 packet header, see the “Implementing IPv6 Addressing and Basic Connectivity” chapter of *Cisco IOS IPv6 Configuration Library* on Cisco.com.

- IPv6 Address Formats
- IPv6 Address Type: Multicast
- IPv6 Address Output Display
- Simplified IPv6 Packet Header

Supported IPv6 Unicast Routing Features

The switch supports hop-by-hop extension header packets, which are routed in software.

128-Bit Wide Unicast Addresses

The switch supports aggregatable global unicast addresses and link-local unicast addresses. It does not support site-local unicast addresses.

- Aggregatable global unicast addresses are IPv6 addresses from the aggregatable global unicast prefix. The address structure enables strict aggregation of routing prefixes and limits the number of routing table entries in the global routing table. These addresses are used on links that are aggregated through organizations and eventually to the Internet service provider.

  These addresses are defined by a global routing prefix, a subnet ID, and an interface ID. Current global unicast address allocation uses the range of addresses that start with binary value 001 (2000::/3). Addresses with a prefix of 2000::/3(001) through E000::/3(111) must have 64-bit interface identifiers in the extended unique identifier (EUI)-64 format.

- Link local unicast addresses can be automatically configured on any interface by using the link-local prefix FE80::/10(1111 1110 10) and the interface identifier in the modified EUI format. Link-local addresses are used in the neighbor discovery protocol (NDP) and the stateless autoconfiguration process. Nodes on a local link use link-local addresses and do not require globally unique addresses to communicate. IPv6 routers do not forward packets with link-local source or destination addresses to other links.

For more information, see the section about IPv6 unicast addresses in the “Implementing IPv6 Addressing and Basic Connectivity” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

DNS for IPv6

IPv6 supports Domain Name System (DNS) record types in the DNS name-to-address and address-to-name lookup processes. The DNS AAAA resource record types support IPv6 addresses and are equivalent to an A address record in IPv4. The switch supports DNS resolution for IPv4 and IPv6.

Path MTU Discovery for IPv6 Unicast

The switch supports advertising the system maximum transmission unit (MTU) to IPv6 nodes and path MTU discovery. Path MTU discovery allows a host to dynamically discover and adjust to differences in the MTU size of every link along a given data path. In IPv6, if a link along the path is not large enough to accommodate the packet size, the source of the packet handles the fragmentation.

ICMPv6

The Internet Control Message Protocol (ICMP) in IPv6 generates error messages, such as ICMP destination unreachable messages, to report errors during processing and other diagnostic functions. In IPv6, ICMP packets are also used in the neighbor discovery protocol and path MTU discovery.

Neighbor Discovery

The switch supports NDP for IPv6, a protocol running on top of ICMPv6, and static neighbor entries for IPv6 stations that do not support NDP. The IPv6 neighbor discovery process uses ICMP messages and solicited-node multicast addresses to determine the link-layer address of a neighbor on the same network (local link), to verify the reachability of the neighbor, and to keep track of neighboring routers.

The switch supports ICMPv6 redirect for routes with mask lengths less than 64 bits. ICMP redirect is not supported for host routes or for summarized routes with mask lengths greater than 64 bits.

Neighbor discovery throttling ensures that the switch CPU is not unnecessarily burdened while it is in the process of obtaining the next hop forwarding information to route an IPv6 packet. The switch drops any additional IPv6 packets whose next hop is the same neighbor that the switch is actively trying to resolve. This drop avoids further load on the CPU.

Default Router Preference

The switch supports IPv6 default router preference (DRP), an extension in router advertisement messages. DRP improves the ability of a host to select an appropriate router, especially when the host is multihomed and the routers are on different links. The switch does not support the Route Information Option in RFC 4191.

An IPv6 host maintains a default router list from which it selects a router for traffic to offlink destinations. The selected router for a destination is then cached in the destination cache. NDP for IPv6 specifies that routers that are reachable or probably reachable are preferred over routers whose reachability is unknown or suspect. For reachable or probably reachable routers, NDP can either select the same router every time or cycle through the router list. By using DRP, you can configure an IPv6 host to prefer one router over another, provided both are reachable or probably reachable.

For more information about DRP for IPv6, see the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

IPv6 Stateless Autoconfiguration and Duplicate Address Detection

The switch uses stateless autoconfiguration to manage link, subnet, and site addressing changes, such as management of host and mobile IP addresses. A host autonomously configures its own link-local address, and booting nodes send router solicitations to request router advertisements for configuring interfaces.

For more information about autoconfiguration and duplicate address detection, see the "Implementing IPv6 Addressing and Basic Connectivity" chapter of *Cisco IOS IPv6 Configuration Library* on Cisco.com.

IPv6 Applications

The switch has IPv6 support for these applications:

- Ping, traceroute, Telnet
- Secure Shell (SSH) over an IPv6 transport
- HTTP server access over IPv6 transport
- DNS resolver for AAAA over IPv4 transport
- Cisco Discovery Protocol (CDP) support for IPv6 addresses

For more information about managing these applications, see the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

DHCP for IPv6 Address Assignment

DHCPv6 enables DHCP servers to pass configuration parameters, such as IPv6 network addresses, to IPv6 clients. The address assignment feature manages non-duplicate address assignment in the correct prefix based on the network where the host is connected. Assigned addresses can be from one or multiple prefix pools. Additional options, such as default domain and DNS name-server address, can be passed back to the client. Address pools can be assigned for use on a specific interface, on multiple interfaces, or the server can automatically find the appropriate pool.

For more information and to configure these features, see the *Cisco IOS IPv6 Configuration Guide*.

This document describes only the DHCPv6 address assignment. For more information about configuring the DHCPv6 client, server, or relay agent functions, see the "Implementing DHCP for IPv6" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Static Routes for IPv6

Static routes are manually configured and define an explicit route between two networking devices. Static routes are useful for smaller networks with only one path to an outside network or to provide security for certain types of traffic in a larger network.

For more information about static routes, see the "Implementing Static Routes for IPv6" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

RIP for IPv6

Routing Information Protocol (RIP) for IPv6 is a distance-vector protocol that uses hop count as a routing metric. It includes support for IPv6 addresses and prefixes and the all-RIP-routers multicast group address FF02::9 as the destination address for RIP update messages.

For more information about RIP for IPv6, see the "Implementing RIP for IPv6" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

OSPF for IPv6

The switch running the IP Base feature set supports Open Shortest Path First (OSPF) for IPv6, a link-state protocol for IP. For more information, see*Cisco IOS IPv6 Configuration Library* on Cisco.com.

HSRP for IPv6

Switches running the IP-services feature set support the Hot Standby Router Protocol (HSRP) for IPv6. HSRP provides routing redundancy for routing IPv6 traffic not dependent on the availability of any single router. IPv6 hosts learn of available routers through IPv6 neighbor discovery router advertisement messages. These messages are multicast periodically or are solicited by hosts.

An HSRP IPv6 group has a virtual MAC address that is derived from the HSRP group number and a virtual IPv6 link-local address that is, by default, derived from the HSRP virtual MAC address. Periodic messages are sent for the HSRP virtual IPv6 link-local address when the HSRP group is active. These messages stop after a final one is sent when the group leaves the active state.

For more information about configuring HSRP for IPv6, see the "Configuring First Hop Redundancy Protocols in IPv6" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

EIGRP IPv6

Switches running the IP services feature set support the Enhanced Interior Gateway Routing Protocol (EIGRP) for IPv6. It is configured on the interfaces on which it runs and does not require a global IPv6 address.

Note Switches running the IP base feature set do not support any IPv6 EIGRP features, including IPv6 EIGRP stub routing.

Before running, an instance of EIGRP IPv6 requires an implicit or explicit router ID. An implicit router ID is derived from a local IPv4 address, so any IPv4 node always has an available router ID. However, EIGRP IPv6 might be running in a network with only IPv6 nodes and therefore might not have an available IPv4 router ID.

For more information about EIGRP for IPv6, see the "Implementing EIGRP for IPv6" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

SNMP and Syslog Over IPv6

To support both IPv4 and IPv6, IPv6 network management requires both IPv6 and IPv4 transports. Syslog over IPv6 supports address data types for these transports.

SNMP and syslog over IPv6 provide these features:

- Support for both IPv4 and IPv6
- IPv6 transport for SNMP and to modify the SNMP agent to support traps for an IPv6 host
- SNMP- and syslog-related MIBs to support IPv6 addressing
- Configuration of IPv6 hosts as trap receivers

For support over IPv6, SNMP modifies the existing IP transport mapping to simultaneously support IPv4 and IPv6. These SNMP actions support IPv6 transport management:

- Opens User Datagram Protocol (UDP) SNMP socket with default settings
- Provides a new transport mechanism called *SR_IPV6_TRANSPORT*
- Sends SNMP notifications over IPv6 transport

- Supports SNMP-named access lists for IPv6 transport
- Supports SNMP proxy forwarding using IPv6 transport
- Verifies SNMP Manager feature works with IPv6 transport

For information on SNMP over IPv6, including configuration procedures, see the “Managing Cisco IOS Applications over IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

For information about syslog over IPv6, including configuration procedures, see the “Implementing IPv6 Addressing and Basic Connectivity” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

HTTP(S) Over IPv6

The HTTP client sends requests to both IPv4 and IPv6 HTTP servers, which respond to requests from both IPv4 and IPv6 HTTP clients. URLs with literal IPv6 addresses must be specified in hexadecimal using 16-bit values between colons.

The accept socket call chooses an IPv4 or IPv6 address family. The accept socket is either an IPv4 or IPv6 socket. The listening socket continues to listen for both IPv4 and IPv6 signals that indicate a connection. The IPv6 listening socket is bound to an IPv6 wildcard address.

The underlying TCP/IP stack supports a dual-stack environment. HTTP relies on the TCP/IP stack and the sockets for processing network-layer interactions.

Basic network connectivity (**ping**) must exist between the client and the server hosts before HTTP connections can be made.

For more information, see the “Managing Cisco IOS Applications over IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Unsupported IPv6 Unicast Routing Features

The switch does not support these IPv6 features:

- IPv6 policy-based routing
- IPv6 virtual private network (VPN) routing and forwarding (VRF) table support
- Support for IPv6 routing protocols: multiprotocol Border Gateway Protocol (BGP) and Intermediate System-to-Intermediate System (IS-IS) routing
- IPv6 packets destined to site-local addresses
- Tunneling protocols, such as IPv4-to-IPv6 or IPv6-to-IPv4
- The switch as a tunnel endpoint supporting IPv4-to-IPv6 or IPv6-to-IPv4 tunneling protocols
- IPv6 unicast reverse-path forwarding
- IPv6 Web Cache Communication Protocol (WCCP)

IPv6 Feature Limitations

Because IPv6 is implemented in switch hardware, some limitations occur due to the IPv6 compressed addresses in the hardware memory. These hardware limitations result in some loss of functionality and limits some features.

These are feature limitations.

- The switch cannot forward SNAP-encapsulated IPv6 packets in hardware. They are forwarded in software.
- The switch cannot apply QoS classification on source-routed IPv6 packets in hardware.

IPv6 and Switch Stacks

The switch supports IPv6 forwarding across the stack and IPv6 host functionality on the stack master. The stack master runs the IPv6 unicast routing protocols and computes the routing tables. They receive the tables and create hardware IPv6 routes for forwarding. The stack master also runs all IPv6 applications.

Note To route IPv6 packets in a stack, all switches in the stack should be running the IP Base feature set.

If a new switch becomes the stack master, it recomputes the IPv6 routing tables and distributes them to the member switches. While the new stack master is being elected and is resetting, the switch stack does not forward IPv6 packets. The stack MAC address changes, which also changes the IPv6 address. When you specify the stack IPv6 address with an extended unique identifier (EUI) by using the **ipv6 address** *ipv6-prefix/prefix length* **eui-64** interface configuration command, the address is based on the interface MAC address. See the Configuring IPv6 Addressing and Enabling IPv6 Routing, on page 1404.

If you configure the persistent MAC address feature on the stack and the stack master changes, the stack MAC address does not change for approximately 4 minutes.

These are the functions of IPv6 stack master and members:

- Stack master:
  - runs IPv6 routing protocols
  - generates routing tables
  - distributes routing tables to stack members that use dCEFv6
  - runs IPv6 host functionality and IPv6 applications
- Stack member (must be running the IP services feature set):
  - receives CEFv6 routing tables from the stack master
  - programs the routes into hardware

    Note IPv6 packets are routed in hardware across the stack if the packet does not have exceptions (IPv6Options) and the switches in the stack have not run out of hardware resources.

  - flushes the CEFv6 tables on master re-election

Default IPv6 Configuration

Table 154: Default IPv6 Configuration

Feature	Default Setting
SDM template	Advance desktop. Default is advanced template
IPv6 addresses	None configured

Configuring IPv6 Addressing and Enabling IPv6 Routing

This section describes how to assign IPv6 addresses to individual Layer 3 interfaces and to globally forward IPv6 traffic on the switch.

Before configuring IPv6 on the switch, consider these guidelines:

-
-
- In the **ipv6 address** interface configuration command, you must enter the *ipv6-address* and *ipv6-prefix* variables with the address specified in hexadecimal using 16-bit values between colons. The *prefix-length* variable (preceded by a slash [/]) is a decimal value that shows how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the address).

To forward IPv6 traffic on an interface, you must configure a global IPv6 address on that interface. Configuring an IPv6 address on an interface automatically configures a link-local address and activates IPv6 for the interface. The configured interface automatically joins these required multicast groups for that link:

- solicited-node multicast group FF02:0:0:0:0:1:ff00::/104 for each unicast address assigned to the interface (this address is used in the neighbor discovery process.)
- all-nodes link-local multicast group FF02::1
- all-routers link-local multicast group FF02::2

For more information about configuring IPv6 routing, see the "Implementing Addressing and Basic Connectivity for IPv6" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Beginning in privileged EXEC mode, follow these steps to assign an IPv6 address to a Layer 3 interface and enable IPv6 routing:

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**sdm prefer dual-ipv4-and-ipv6** {**advanced** | **vlan**} **Example:** `Switch(config)# sdm prefer dual-ipv4-and-ipv6 default`	Selects an SDM template that supports IPv4 and IPv6. • **advanced—**Sets the switch to the default template to balance system resources. • **vlan—**Maximizes VLAN configuration on the switch with no routing supported in hardware. **Note** Advanced is available at all license levels. VLAN template is available only in lanbase.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**reload** **Example:** `Switch# reload`	Reloads the operating system.
Step 5	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode after the switch reloads.
Step 6	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the Layer 3 interface to configure. The interface can be a physical interface, a switch virtual interface (SVI), or a Layer 3 EtherChannel.
Step 7	**no switchport** **Example:** `Switch(config-if)# no switchport`	Removes the interface from Layer 2 configuration mode (if it is a physical interface).
Step 8	Use one of the following:	• Specifies a global IPv6 address with an extended unique identifier (EUI) in the low-order 64 bits of the IPv6 address.

	Command or Action	Purpose

	• **ipv6 address** *ipv6-prefix/prefix length* **eui-64**
	• **ipv6 address** *ipv6-address/prefix length*
	• **ipv6 address** *ipv6-address* **link-local**
	• **ipv6 enable**
	• **ipv6 address***WORD*
	• **ipv6 address***autoconfig*
	• **ipv6 address***dhcp*
	
	Example:
	`Switch(config-if)# ipv6 address 2001:0DB8:c18:1::/64 eui 64`
	
	`Switch(config-if)# ipv6 address 2001:0DB8:c18:1::/64`
	
	`Switch(config-if)# ipv6 address 2001:0DB8:c18:1:: link-local`
	
	`Switch(config-if)# ipv6 enable`
		Specify only the network prefix; the last 64 bits are automatically computed from the switch MAC address. This enables IPv6 processing on the interface.
		• Manually configures an IPv6 address on the interface.
		• Specifies a link-local address on the interface to be used instead of the link-local address that is automatically configured when IPv6 is enabled on the interface. This command enables IPv6 processing on the interface.
		• Automatically configures an IPv6 link-local address on the interface, and enables the interface for IPv6 processing. The link-local address can only be used to communicate with nodes on the same link.

Step 9	**exit** **Example:** `Switch(config-if)# exit`	Returns to global configuration mode.
Step 10	**ip routing** **Example:** `Switch(config)# ip routing`	Enables IP routing on the switch.
Step 11	**ipv6 unicast-routing** **Example:** `Switch(config)# ipv6 unicast-routing`	Enables forwarding of IPv6 unicast data packets.
Step 12	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 13	**show ipv6 interface** *interface-id* **Example:** `Switch# show ipv6 interface gigabitethernet 1/0/1`	Verifies your entries.
Step 14	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring IPv4 and IPv6 Protocol Stacks

Beginning in privileged EXEC mode, follow these steps to configure a Layer 3 interface to support both IPv4 and IPv6 and to enable IPv6 routing.

Note To disable IPv6 processing on an interface that has not been configured with an IPv6 address, use the **no ipv6 enable** interface configuration command.

SUMMARY STEPS

1. **configure terminal**
2. **ip routing**
3. **ipv6 unicast-routing**
4. **interface** *interface-id*
5. **no switchport**
6. **ip address** *ip-address mask* [**secondary**]
7. Use one of the following:
   - **ipv6 address** *ipv6-prefix/prefix length* **eui-64**
   - **ipv6 address** *ipv6-address/prefix length*
   - **ipv6 address** *ipv6-address* **link-local**
   - **ipv6 enable**
   - **ipv6 address***WORD*
   - **ipv6 address***autoconfig*
   - **ipv6 address***dhcp*
8. **end**
9. Use one of the following:
   - **show interface** *interface-id*
   - **show ip interface** *interface-id*
   - **show ipv6 interface** *interface-id*
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip routing** **Example:** `Switch(config)# ip routing`	Enables routing on the switch.

	Command or Action	Purpose
Step 3	**ipv6 unicast-routing** **Example:** `Switch(config)# ipv6 unicast-routing`	Enables forwarding of IPv6 data packets on the switch.
Step 4	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 5	**no switchport** **Example:** `Switch(config-if)# no switchport`	Removes the interface from Layer 2 configuration mode (if it is a physical interface).
Step 6	**ip address** *ip-address mask* [**secondary**] **Example:** `Switch(config-if)# ip address 10.1.2.3 255.255.255`	Specifies a primary or secondary IPv4 address for the interface.
Step 7	Use one of the following: • **ipv6 address** *ipv6-prefix/prefix length* **eui-64** • **ipv6 address** *ipv6-address/prefix length* • **ipv6 address** *ipv6-address* **link-local** • **ipv6 enable** • **ipv6 address***WORD* • **ipv6 address***autoconfig* • **ipv6 address***dhcp*	• Specifies a global IPv6 address. Specify only the network prefix; the last 64 bits are automatically computed from the switch MAC address. • Specifies a link-local address on the interface to be used instead of the automatically configured link-local address when IPv6 is enabled on the interface. • Automatically configures an IPv6 link-local address on the interface, and enables the interface for IPv6 processing. The link-local address can only be used to communicate with nodes on the same link. **Note** To remove all manually configured IPv6 addresses from an interface, use the **no ipv6 address** interface configuration command without arguments.
Step 8	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 9	Use one of the following: • **show interface** *interface-id* • **show ip interface** *interface-id* • **show ipv6 interface** *interface-id*	Verifies your entries.
Step 10	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Default Router Preference

Router advertisement messages are sent with the default router preference (DRP) configured by the **ipv6 nd router-preference** interface configuration command. If no DRP is configured, RAs are sent with a medium preference.

A DRP is useful when two routers on a link might provide equivalent, but not equal-cost routing, and policy might dictate that hosts should prefer one of the routers.

For more information about configuring DRP for IPv6, see the "Implementing IPv6 Addresses and Basic Connectivity" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Beginning in privileged EXEC mode, follow these steps to configure a DRP for a router on an interface.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode and identifies the Layer 3 interface on which you want to specify the DRP.
Step 3	**ipv6 nd router-preference** {**high** | **medium** | **low**} **Example:** `Switch(config-if)# ipv6 nd router-preference medium`	Specifies a DRP for the router on the switch interface.

	Command or Action	Purpose
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show ipv6 interface** **Example:** `Switch# show ipv6 interface`	Verifies the configuration.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring IPv6 ICMP Rate Limiting

ICMP rate limiting is enabled by default with a default interval between error messages of 100 milliseconds and a bucket size (maximum number of tokens to be stored in a bucket) of 10.

Beginning in privileged EXEC mode, follow these steps to change the ICMP rate-limiting parameters:

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ipv6 icmp error-interval** *interval* [*bucketsize*] **Example:** `Switch(config)# ipv6 icmp error-interval 50 20`	Configures the interval and bucket size for IPv6 ICMP error messages: • *interval*—The interval (in milliseconds) between tokens being added to the bucket. The range is from 0 to 2147483647 milliseconds. • *bucketsize*—(Optional) The maximum number of tokens stored in the bucket. The range is from 1 to 200.

	Command or Action	Purpose
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show ipv6 interface** [*interface-id*] **Example:** `Switch# show ipv6 interface gigabitethernet 1/0/1`	Verifies your entries.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring CEF and dCEF for IPv6

Cisco Express Forwarding (CEF) is a Layer 3 IP switching technology to improve network performance. CEF implements an advanced IP look-up and forwarding algorithm to deliver maximum Layer 3 switching performance. It is less CPU-intensive than fast-switching route-caching, allowing more CPU processing power to be dedicated to packet forwarding. In a switch stack, the hardware uses distributed CEF (dCEF) in the stack. IPv4 CEF and dCEF are enabled by default. IPv6 CEF and dCEF are disabled by default, but automatically enabled when you configure IPv6 routing.

IPv6 CEF and dCEF are automatically disabled when IPv6 routing is unconfigured. IPv6 CEF and dCEF cannot disabled through configuration. You can verify the IPv6 state by entering the **show ipv6 cef** privileged EXEC command.

To route IPv6 unicast packets, you must first globally configure forwarding of IPv6 unicast packets by using the **ipv6 unicast-routing** global configuration command, and you must configure an IPv6 address and IPv6 processing on an interface by using the **ipv6 address** interface configuration command.

For more information about configuring CEF and dCEF, see *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Configuring Static Routing for IPv6

Before configuring a static IPv6 route, you must enable routing by using the **ip routing** global configuration command, enable the forwarding of IPv6 packets by using the **ipv6 unicast-routing** global configuration command, and enable IPv6 on at least one Layer 3 interface by configuring an IPv6 address on the interface.

For more information about configuring static IPv6 routing, see the “Implementing Static Routes for IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ipv6 route** *ipv6-prefix/prefix length* {*ipv6-address* | *interface-id* [*ipv6-address*]} [*administrative distance*] **Example:** `Switch(config)# ipv6 route 2001:0DB8::/32 gigabitethernet2/0/1 130`	Configures a static IPv6 route. • *ipv6-prefix*—The IPv6 network that is the destination of the static route. It can also be a hostname when static host routes are configured. • */prefix length*—The length of the IPv6 prefix. A decimal value that shows how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the address). A slash mark must precede the decimal value. • *ipv6-address*—The IPv6 address of the next hop that can be used to reach the specified network. The IPv6 address of the next hop need not be directly connected; recursion is done to find the IPv6 address of the directly connected next hop. The address must be in the form documented in RFC 2373, specified in hexadecimal using 16-bit values between colons. • *interface-id*—Specifies direct static routes from point-to-point and broadcast interfaces. With point-to-point interfaces, there is no need to specify the IPv6 address of the next hop. With broadcast interfaces, you should always specify the IPv6 address of the next hop, or ensure that the specified prefix is assigned to the link, specifying a link-local address as the next hop. You can optionally specify the IPv6 address of the next hop to which packets are sent. **Note** You must specify an *interface-id* when using a link-local address as the next hop (the link-local next hop must also be an adjacent router). • *administrative distance*—(Optional) An administrative distance. The range is 1 to 254; the default value is 1, which gives static routes precedence over any other type of route except connected routes. To configure a floating static route, use an administrative distance greater than that of the dynamic routing protocol.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	Use one of the following: • **show ipv6 static** [*ipv6-address* | *ipv6-prefix/prefix length*]	Verifies your entries by displaying the contents of the IPv6 routing table. • **interface** *interface-id*—(Optional) Displays only those static routes with the specified interface as an egress interface.

	Command or Action	Purpose

	[**interface** *interface-id*] [**detail**]][**recursive**] [**detail**]
	• **show ipv6 route static** [*updated*]
	
	Example:
	`Switch# show ipv6 static 2001:0DB8::/32 interface gigabitethernet2/0/1`
	or
	`Switch# show ipv6 route static`
		• **recursive—**(Optional) Displays only recursive static routes. The **recursive** keyword is mutually exclusive with the **interface** keyword, but it can be used with or without the IPv6 prefix included in the command syntax.
		• **detail—**(Optional) Displays this additional information:
		◦ For valid recursive routes, the output path set, and maximum resolution depth.
		◦ For invalid routes, the reason why the route is not valid.

Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring RIP for IPv6

Before configuring the switch to run IPv6 RIP, you must enable routing by using the **ip routing** global configuration command, enable the forwarding of IPv6 packets by using the **ipv6 unicast-routing** global configuration command, and enable IPv6 on any Layer 3 interfaces on which IPv6 RIP is to be enabled.

For more information about configuring RIP routing for IPv6, see the “Implementing RIP for IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com,

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ipv6 router rip** *name* **Example:** `Switch(config)# ipv6 router rip cisco`	Configures an IPv6 RIP routing process, and enters router configuration mode for the process.

	Command or Action	Purpose
Step 3	**maximum-paths** *number-paths* **Example:** `Switch(config-router)# maximum-paths 6`	(Optional) Define the maximum number of equal-cost routes that IPv6 RIP can support. The range is from 1 to 32, and the default is 16 routes.
Step 4	**exit** **Example:** `Switch(config-router)# exit`	Returns to global configuration mode.
Step 5	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 6	**ipv6 rip** *name* **enable** **Example:** `Switch(config-if)# ipv6 rip cisco enable`	Enables the specified IPv6 RIP routing process on the interface.
Step 7	**ipv6 rip** *name* **default-information** {**only** | **originate**} **Example:** `Switch(config-if)# ipv6 rip cisco default-information only`	(Optional) Originates the IPv6 default route (::/0) into the RIP routing process updates sent from the specified interface. **Note** To avoid routing loops after the IPv6 default route (::/0) is originated from any interface, the routing process ignores all default routes received on any interface. • **only**—Select to originate the default route, but suppress all other routes in the updates sent on this interface. • **originate**—Select to originate the default route in addition to all other routes in the updates sent on this interface.
Step 8	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 9	Use one of the following: • **show ipv6 rip** [*name*] [**interface***interface-id*] [**database**] [**next-hops**] • **show ipv6 rip**	• Displays information about current IPv6 RIP processes. • Displays the current contents of the IPv6 routing table.

	Command or Action	Purpose

	Example:
	`Switch# show ipv6 rip cisco interface gigabitethernet2/0/1`
	or
	`Switch# show ipv6 rip`	

Step 10	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring OSPF for IPv6

You can customize OSPF for IPv6 for your network. However, the defaults for OSPF in IPv6 are set to meet the requirements of most customers and features.

Follow these guidelines:

- Be careful when changing the defaults for IPv6 commands. Changing the defaults might adversely affect OSPF for the IPv6 network.
- Before you enable IPv6 OSPF on an interface, you must enable routing by using the **ip routing** global configuration command, enable the forwarding of IPv6 packets by using the **ipv6 unicast-routing** global configuration command, and enable IPv6 on Layer 3 interfaces on which you are enabling IPv6 OSPF.

For more information about configuring OSPF routing for IPv6, see the “Implementing OSPF for IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ipv6 router ospf** *process-id* **Example:** `Switch(config)# ipv6 router ospf 21`	Enables OSPF router configuration mode for the process. The process ID is the number assigned administratively when enabling the OSPF for IPv6 routing process. It is locally assigned and can be a positive integer from 1 to 65535.

	Command or Action	Purpose
Step 3	**area** *area-id* **range** {*ipv6-prefix/prefix length*} [**advertise** | **not-advertise**] [**cost** *cost*] **Example:** `Switch(config)# area .3 range 2001:0DB8::/32 not-advertise`	(Optional) Consolidates and summarizes routes at an area boundary. • *area-id*—Identifier of the area about which routes are to be summarized. It can be specified as either a decimal value or as an IPv6 prefix. • *ipv6-prefix/prefix length*—The destination IPv6 network and a decimal value that shows how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the address). A slash mark (/) must precede the decimal value. • **advertise**—(Optional) Sets the address range status to advertise and generate a Type 3 summary link-state advertisement (LSA). • **not-advertise**—(Optional) Sets the address range status to DoNotAdvertise. The Type 3 summary LSA is suppressed, and component networks remain hidden from other networks. • **cost** *cost*—(Optional) Sets the metric or cost for this summary route, which is used during OSPF SPF calculation to determine the shortest paths to the destination. The value can be 0 to 16777215.
Step 4	**maximum paths** *number-paths* **Example:** `Switch(config)# maximum paths 16`	(Optional) Defines the maximum number of equal-cost routes to the same destination that IPv6 OSPF should enter in the routing table. The range is from 1 to 32, and the default is 16 paths.
Step 5	**exit** **Example:** `Switch(config-if)# exit`	Returns to global configuration mode.
Step 6	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 7	**ipv6 ospf** *process-id* **area** *area-id* [**instance** *instance-id*] **Example:** `Switch(config-if)# ipv6 ospf 21 area .3`	Enables OSPF for IPv6 on the interface. • **instance** *instance-id*—(Optional) Instance identifier.

	Command or Action	Purpose
Step 8	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Step 9	Use one of the following:
	• **show ipv6 ospf** [*process-id*] [*area-id*] **interface** [*interface-id*]
	• **show ipv6 ospf** [*process-id*] [*area-id*]
	
	Example:
	`Switch# show ipv6 ospf 21 interface gigabitethernet2/0/1`
	
	or
	
	`Switch# show ipv6 ospf 21`
		• Displays information about OSPF interfaces.
		• Displays general information about OSPF routing processes.

Step 10	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring EIGRP for IPv6

Before configuring the switch to run IPv6 EIGRP, enable routing by entering the **ip routing global configuration** command, enable the forwarding of IPv6 packets by entering the **ipv6 unicast-routing global** configuration command, and enable IPv6 on any Layer 3 interfaces on which you want to enable IPv6 EIGRP.

To set an explicit router ID, use the **show ipv6 eigrp** command to see the configured router IDs, and then use the **router-id** command.

As with EIGRP IPv4, you can use EIGRPv6 to specify your EIGRP IPv6 interfaces and to select a subset of those as passive interfaces. Use the **passive-interface** command to make an interface passive, and then use the **no passive-interface** command on selected interfaces to make them active. EIGRP IPv6 does not need to be configured on a passive interface.

For more configuration procedures, see the “Implementing EIGRP for IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Displaying IPv6

For complete syntax and usage information on these commands, see the Cisco IOS command reference publications.

Table 155: Commands for Monitoring IPv6

Command	Purpose
show ipv6 access-list	Displays a summary of access lists.
show ipv6 cef	Displays Cisco Express Forwarding for IPv6.
show ipv6 interface *interface-id*	Displays IPv6 interface status and configuration.
show ipv6 mtu	Displays IPv6 MTU per destination cache.
show ipv6 neighbors	Displays IPv6 neighbor cache entries.
show ipv6 prefix-list	Displays a list of IPv6 prefix lists.
show ipv6 protocols	Displays IPv6 routing protocols on the switch.
show ipv6 route	Displays the IPv6 route table entries.
show ipv6 static	Displays IPv6 static routes.
show ipv6 traffic	Displays IPv6 traffic statistics.

Configuring DHCP for IPv6 Address Assignment

Default DHCPv6 Address Assignment Configuration

By default, no DHCPv6 features are configured on the switch.

DHCPv6 Address Assignment Configuration Guidelines

When configuring DHCPv6 address assignment, consider these guidelines:

- In the procedures, the specified interface must be one of these Layer 3 interfaces:
    - DHCPv6 IPv6 routing must be enabled on a Layer 3 interface.
    - SVI: a VLAN interface created by using the **interface vlan** *vlan_id* command.

  - EtherChannel port channel in Layer 3 mode: a port-channel logical interface created by using the **interface port-channel port-channel-number** command.

- The switch can act as a DHCPv6 client, server, or relay agent. The DHCPv6 client, server, and relay function are mutually exclusive on an interface.
- The DHCPv6 client, server, or relay agent runs only on the master switch. When there is a stack master re-election, the new master switch retains the DHCPv6 configuration. However, the local RAM copy of the DHCP server database lease information is not retained.

Enabling DHCPv6 Server Function

Use the **no** form of the DHCP pool configuration mode commands to change the DHCPv6 pool characteristics. To disable the DHCPv6 server function on an interface, use the **no ipv6 dhcp server** interface configuration command.

Beginning in privileged EXEC mode, follow these steps to enable the DHCPv6 server function on an interface.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ipv6 dhcp pool** *poolname* **Example:** `Switch(config)# ipv6 dhcp pool 7`	Enters DHCP pool configuration mode, and define the name for the IPv6 DHCP pool. The pool name can be a symbolic string (such as Engineering) or an integer (such as 0).
Step 3	**address prefix** *IPv6-prefix* {**lifetime**} {*t1 t1* | **infinite**} **Example:** `Switch(config-dhcpv6)# address prefix 2001:1000::0/64 lifetime 3600`	(Optional) Specifies an address prefix for address assignment. This address must be in hexadecimal, using 16-bit values between colons. **lifetime** *t1 t1*—Specifies a time interval (in seconds) that an IPv6 address prefix remains in the valid state. The range is 5 to 4294967295 seconds. Specify **infinite** for no time interval.
Step 4	**link-address** *IPv6-prefix* **Example:** `Switch(config-dhcpv6)# link-address 2001:1002::0/64`	(Optional) Specifies a link-address IPv6 prefix. When an address on the incoming interface or a link-address in the packet matches the specified IPv6 prefix, the server uses the configuration information pool. This address must be in hexadecimal, using 16-bit values between colons.

	Command or Action	Purpose
Step 5	**vendor-specific** *vendor-id* **Example:** `Switch(config-dhcpv6)# vendor-specific 9`	(Optional) Enters vendor-specific configuration mode and specifies a vendor-specific identification number. This number is the vendor IANA Private Enterprise Number. The range is 1 to 4294967295.
Step 6	**suboption** *number* {**address** *IPv6-address* | **ascii** *ASCII-string* | **hex** *hex-string*} **Example:** `Switch(config-dhcpv6-vs)# suboption 1 address 1000:235D::`	(Optional) Enters a vendor-specific suboption number. The range is 1 to 65535. Enter an IPv6 address, ASCII text, or a hex string as defined by the suboption parameters.
Step 7	**exit** **Example:** `Switch(config-dhcpv6-vs)# exit`	Returns to DHCP pool configuration mode.
Step 8	**exit** **Example:** `Switch(config-dhcpv6)# exit`	Returns to global configuration mode.
Step 9	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the interface to configure.
Step 10	**ipv6 dhcp server** [*poolname* | **automatic**] [**rapid-commit**] [**preference** *value*] [**allow-hint**] **Example:** `Switch(config-if)# ipv6 dhcp server automatic`	Enables DHCPv6 server function on an interface. • *poolname*—(Optional) User-defined name for the IPv6 DHCP pool. The pool name can be a symbolic string (such as Engineering) or an integer (such as 0). • **automatic**—(Optional) Enables the system to automatically determine which pool to use when allocating addresses for a client. • **rapid-commit**—(Optional) Allows two-message exchange method. • **preference** *value*—(Optional) Configures the preference value carried in the preference option in the advertise message sent by the server. The range is from 0 to 255. The preference value default is 0.

	Command or Action	Purpose
		• **allow-hint—**(Optional) Specifies whether the server should consider client suggestions in the SOLICIT message. By default, the server ignores client hints.
Step 11	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Step 12	Do one of the following:
	• **show ipv6 dhcp pool**
	• **show ipv6 dhcp interface**
	
	Example:
	`Switch# show ipv6 dhcp pool`
	or
	`Switch# show ipv6 dhcp interface`
		• Verifies DHCPv6 pool configuration.
		• Verifies that the DHCPv6 server function is enabled on an interface.

Step 13	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Enabling DHCPv6 Client Function

This task explains how to enable the DHCPv6 client on an interface.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the interface to configure.
Step 3	**ipv6 address dhcp** [**rapid-commit**] **Example:** `Switch(config-if)# ipv6 address dhcp rapid-commit`	Enables the interface to acquire an IPv6 address from the DHCPv6 server. **rapid-commit**—(Optional) Allow two-message exchange method for address assignment.
Step 4	**ipv6 dhcp client request** [**vendor-specific**] **Example:** `Switch(config-if)# ipv6 dhcp client request vendor-specific`	(Optional) Enables the interface to request the vendor-specific option.
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 6	**show ipv6 dhcp interface** **Example:** `Switch# show ipv6 dhcp interface`	Verifies that the DHCPv6 client is enabled on an interface.

Configuration Examples for IPv6 Unicast Routing

Configuring IPv6 Addressing and Enabling IPv6 Routing: Example

This example shows how to enable IPv6 with both a link-local address and a global address based on the IPv6 prefix 2001:0DB8:c18:1::/64. The EUI-64 interface ID is used in the low-order 64 bits of both addresses. Output from the **show ipv6 interface** EXEC command is included to show how the interface ID (20B:46FF:FE2F:D940) is appended to the link-local prefix FE80::/64 of the interface.

```
Switch(config)# ipv6 unicast-routing
Switch(config)# interface gigabitethernet1/0/11

Switch(config-if)# ipv6 address 2001:0DB8:c18:1::/64 eui 64
```

```
Switch(config-if)# end
Switch# show ipv6 interface gigabitethernet1/0/11
GigabitEthernet1/0/11 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::20B:46FF:FE2F:D940
  Global unicast address(es):
  2001:0DB8:c18:1:20B:46FF:FE2F:D940, subnet is 2001:0DB8:c18:1::/64 [EUI]
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF2F:D940
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  ND advertised reachable time is 0 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
  Hosts use stateless autoconfig for addresses.
```

Configuring Default Router Preference: Example

This example shows how to configure a DRP of *high* for the router on an interface.

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ipv6 nd router-preference high
Switch(config-if)# end
```

Configuring IPv4 and IPv6 Protocol Stacks: Example

This example shows how to enable IPv4 and IPv6 routing on an interface.

```
Switch(config)# ip routing
Switch(config)# ipv6 unicast-routing
Switch(config)# interface fastethernet1/0/11
Switch(config-if)# no switchport
Switch(config-if)# ip address 192.168.99.1 255.255.255.0
Switch(config-if)# ipv6 address 2001:0DB8:c18:1::/64 eui 64
Switch(config-if)# end
```

Enabling DHCPv6 Server Function: Example

This example shows how to configure a pool called *engineering* with an IPv6 address prefix:

```
Switch# configure terminal
Switch(config)# ipv6 dhcp pool engineering
Switch(config-dhcpv6)#address prefix 2001:1000::0/64
Switch(config-dhcpv6)# end
```

This example shows how to configure a pool called *testgroup* with three link-addresses and an IPv6 address prefix:

```
Switch# configure terminal
Switch(config)# ipv6 dhcp pool testgroup
Switch(config-dhcpv6)# link-address 2001:1001::0/64
```

```
Switch(config-dhcpv6)# link-address 2001:1002::0/64
Switch(config-dhcpv6)# link-address 2001:2000::0/48
Switch(config-dhcpv6)# address prefix 2001:1003::0/64
Switch(config-dhcpv6)# end
```

This example shows how to configure a pool called *350* with vendor-specific options:

```
Switch# configure terminal
Switch(config)# ipv6 dhcp pool 350
Switch(config-dhcpv6)# address prefix 2001:1005::0/48
Switch(config-dhcpv6)# vendor-specific 9
Switch(config-dhcpv6-vs)# suboption 1 address 1000:235D::1
Switch(config-dhcpv6-vs)# suboption 2 ascii "IP-Phone"
Switch(config-dhcpv6-vs)# end
```

Enabling DHCPv6 Client Function: Example

This example shows how to acquire an IPv6 address and to enable the rapid-commit option:

```
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# ipv6 address dhcp rapid-commit
```

Configuring IPv6 ICMP Rate Limiting: Example

This example shows how to configure an IPv6 ICMP error message interval of 50 milliseconds and a bucket size of 20 tokens.

```
Switch(config)#ipv6 icmp error-interval 50 20
```

Configuring Static Routing for IPv6: Example

This example shows how to configure a floating static route to an interface with an administrative distance of 130:

```
Switch(config)# ipv6 route 2001:0DB8::/32 gigabitethernet2/0/1 130
```

Configuring RIP for IPv6: Example

This example shows how to enable the RIP routing process *cisco* with a maximum of eight equal-cost routes and to enable it on an interface:

```
Switch(config)# ipv6 router rip cisco
Switch(config-router)# maximum-paths 8
Switch(config)# exit
Switch(config)# interface gigabitethernet2/0/11
Switch(config-if)# ipv6 rip cisco enable
```

Displaying IPv6: Example

This is an example of the output from the **show ipv6 interface** privileged EXEC command:

```
Switch# show ipv6 interface
Vlan1 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::20B:46FF:FE2F:D940
  Global unicast address(es):
    3FFE:C000:0:1:20B:46FF:FE2F:D940, subnet is 3FFE:C000:0:1::/64 [EUI]
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF2F:D940
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  ND advertised reachable time is 0 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
<output truncated>
```

CHAPTER 66

Configuring IPv6 Client IP Address Learning

Prerequisites for IPv6 Client Address Learning

Before configuring IPv6 client address learning, configure the wireless clients to support IPv6.

Related Topics

Configuring RA Guard Policy (CLI), on page 1434

Information About IPv6 Client Address Learning

Client Address Learning is configured on switch to learn the wireless client's IPv4 and IPv6 address and clients transition state maintained by the switch on an association, re-association, de-authentication and timeout.

There are three ways for IPv6 client to acquire IPv6 addresses:

- Stateless Address Auto-Configuration (SLACC)
- Stateful DHCPv6
- Static Configuration

For all of these methods, the IPv6 client always sends neighbor solicitation DAD (Duplicate Address Detection) request to ensure there is no duplicate IP address on the network. The switch snoops the client's NDP and DHCPv6 packets to learn about its client IP addresses.

SLAAC Address Assignment

The most common method for IPv6 client address assignment is Stateless Address Auto-Configuration (SLAAC). SLAAC provides simple plug-and-play connectivity where clients self-assign an address based on the IPv6 prefix. This process is achieved

Stateless Address Auto-Configuration (SLAAC) is configured as follows:

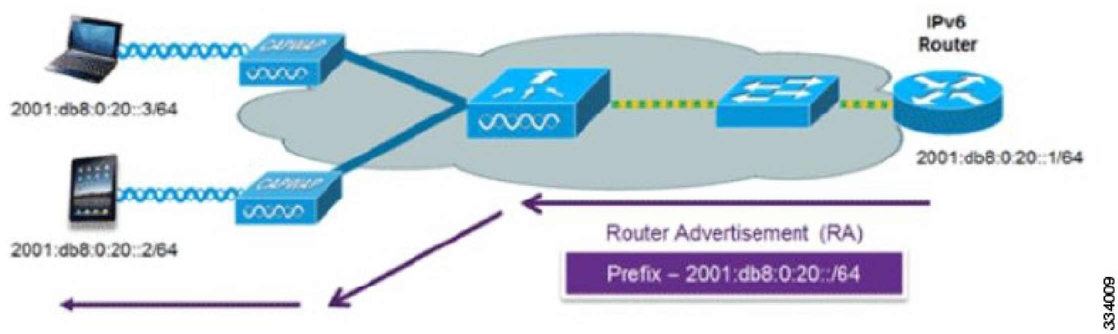
- Host sends a router solicitation message.
- Hosts waits for a Router Advertisement message.
- Hosts take the first 64 bits of the IPv6 prefix from the Router Advertisement message and combines it with the 64 bit EUI-64 address (in the case of ethernet, this is created from the MAC Address) to create a global unicast message. The host also uses the source IP address, in the IP header, of the Router Advertisement message, as its default gateway.
- Duplicate Address Detection is performed by IPv6 clients in order to ensure that random addresses that are picked do not collide with other clients.
- The choice of algorithm is up to the client and is often configurable.

The last 64 bits of the IP v6 address can be learned based on the following 2 algorithms:

- EUI-64 which is based on the MAC address of the interface, or

- Private addresses that are randomly generated.

Figure 67: SLAAC Address Assignment



The following Cisco IOS configuration commands from a Cisco-capable IPv6 router are used to enable SLAAC addressing and router advertisements:

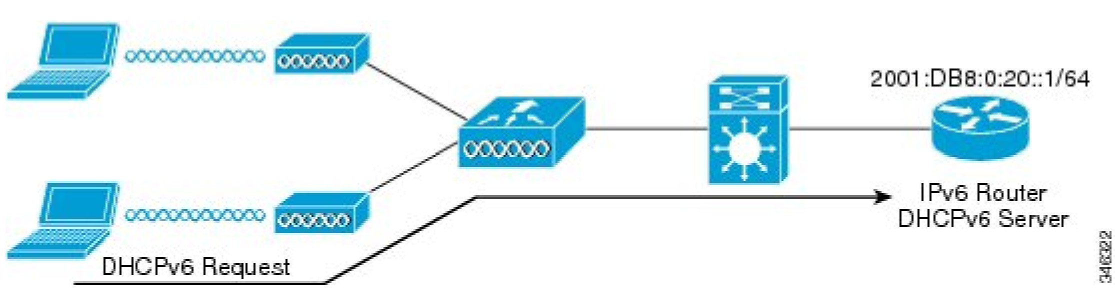
```
ipv6 unicast-routing
interface Vlan20
description IPv6-SLAAC
ip address 192.168.20.1 255.255.255.0
ipv6 address FE80:DB8:0:20::1 linklocal
ipv6 address 2001:DB8:0:20::1/64
ipv6 enable
end
```

Related Topics

Configuring IPv6 Snooping (CLI), on page 1438
Configuring DHCP Pool (CLI), on page 1442
Configuring Stateless Auto Address Configuration Without DHCP (CLI), on page 1443
Configuring Stateless Auto Address Configuration With DHCP (CLI), on page 1444
Configuring Stateful DHCP Locally (CLI), on page 1445
Configuring Stateful DHCP Externally (CLI), on page 1447

Stateful DHCPv6 Address Assignment

Figure 68: Stateful DHCPv6 Address Assignment



The use of DHCPv6 is not required for IPv6 client connectivity if SLAAC is already deployed. There are two modes of operation for DHCPv6 called Stateless and Stateful.

The DHCPv6 Stateless mode is used to provide clients with additional network information that is not available in the router advertisement, but not an IPv6 address as this is already provided by SLAAC. This information can include the DNS domain name, DNS server(s), and other DHCP vendor-specific options. This interface configuration is for a Cisco IOS IPv6 router implementing stateless DHCPv6 with SLAAC enabled:

```
ipv6 unicast-routing
ipv6 dhcp pool IPV6_DHCPPOOL
address prefix 2001:db8:5:10::/64
domain-name cisco.com
dns-server 2001:db8:6:6::1
interface Vlan20
description IPv6-DHCP-Stateless
ip address 192.168.20.1 255.255.255.0
ipv6 nd other-config-flag
ipv6 dhcp server IPV6_DHCPPOOL
ipv6 address 2001:DB8:0:20::1/64
end
```

The DHCPv6 Stateful option, also known as managed mode, operates similarly to DHCPv4 in that it assigns unique addresses to each client instead of the client generating the last 64 bits of the address as in SLAAC. This interface configuration is for a Cisco IOS IPv6 router implementing stateful DHCPv6 on a local Switch:

```
ipv6 unicast-routing
ipv6 dhcp pool IPV6_DHCPPOOL
address prefix 2001:db8:5:10::/64
domain-name cisco.com
dns-server 2001:db8:6:6::1
interface Vlan20
description IPv6-DHCP-Stateful
ip address 192.168.20.1 255.255.255.0
ipv6 address 2001:DB8:0:20::1/64
ipv6 nd prefix 2001:DB8:0:20::/64 no-advertise
ipv6 nd managed-config-flag
ipv6 nd other-config-flag
ipv6 dhcp server IPV6_DHCPPOOL
end
```

This interface configuration is for a Cisco IOS IPv6 router implementing stateful DHCPv6 on an external DHCP server:

```
ipv6 unicast-routing
domain-name cisco.com
dns-server 2001:db8:6:6::1
interface Vlan20
description IPv6-DHCP-Stateful
ip address 192.168.20.1 255.255.255.0
ipv6 address 2001:DB8:0:20::1/64
ipv6 nd prefix 2001:DB8:0:20::/64 no-advertise
ipv6 nd managed-config-flag
ipv6 nd other-config-flag
ipv6 dhcp_relay destination 2001:DB8:0:20::2
end
```

Related Topics

Configuring IPv6 Snooping (CLI), on page 1438
Configuring DHCP Pool (CLI), on page 1442
Configuring Stateless Auto Address Configuration Without DHCP (CLI), on page 1443
Configuring Stateless Auto Address Configuration With DHCP (CLI), on page 1444
Configuring Stateful DHCP Locally (CLI), on page 1445
Configuring Stateful DHCP Externally (CLI), on page 1447

Static IP Address Assignment

Statically configured address on a client.

Router Solicitation

A Router Solicitation message is issued by a host controller to facilitate local routers to transmit Router Advertisement from which it can obtain information about local routing or perform Stateless Auto-configuration. Router Advertisements are transmitted periodically and the host prompts with an immediate Router Advertisement using a Router Solicitation such as - when it boots or following a restart operation.

Related Topics

Configuring IPv6 ND Suppress Policy (CLI), on page 1439

Router Advertisement

A Router Advertisement message is issued periodically by a router or in response to a Router Solicitation message from a host. The information contained in these messages is used by hosts to perform Stateless Auto-configuration and to modify its routing table.

Related Topics

Configuring IPv6 ND Suppress Policy (CLI), on page 1439

Neighbor Discovery

IPv6 Neighbor Discovery is a set of messages and processes that determine relationships between neighboring nodes. Neighbor Discovery replaces ARP, ICMP Router Discovery, and ICMP Redirect used in IPv4.

IPv6 Neighbor Discovery inspection analyzes neighbor discovery messages in order to build a trusted binding table database, and IPv6 neighbor discovery packets that do not comply are dropped. The neighbor binding table in the switch tracks each IPv6 address and its associated MAC address. Clients are expired from the table according to Neighbor Binding timers.

Related Topics

Configuring IPv6 ND Suppress Policy (CLI), on page 1439

Neighbor Discovery Suppression

The IPv6 addresses of wireless clients are cached by the switch. When the switch receives an NS multicast looking for an IPv6 address, and if the target address is known to the switch and belongs to one of its clients, the switch will reply with an NA message on behalf of the client. The result of this process generates the equivalent of the Address Resolution Protocol (ARP) table of IPv4 but is more efficient - uses generally fewer messages.

Note The switch acts like proxy and respond with NA, only when the **ipv6 nd suppress** command is configured

If the switch does not have the IPv6 address of a wireless client, the switch will not respond with NA and forward the NS packet to the wireless side. To resolve this, an NS Multicast Forwarding knob is provided. If this knob is enabled, the switch gets the NS packet for the IPv6 address that it does not have (cache miss) and forwards it to the wireless side. This packet reaches the intended wireless client and the client replies with NA.

This cache miss scenario occurs rarely, and only very few clients which do not implement complete IPv6 stack may not advertise their IPv6 address during NDP.

Related Topics

Configuring IPv6 ND Suppress Policy (CLI), on page 1439

RA Guard

IPv6 clients configure IPv6 addresses and populate their router tables based on IPv6 router advertisement (RA) packets. The RA guard feature is similar to the RA guard feature of wired networks. RA guard increases the security of the IPv6 network by dropping the unwanted or rogue RA packets that come from wireless clients. If this feature is not configured, malicious IPv6 wireless clients announce themselves as the router for the network often with high priority, which would take higher precedence over legitimate IPv6 routers.

RA-Guard also examines the incoming RA's and decides whether to switch or block them based solely on information found in the message or in the switch configuration. The information available in the frames received is useful for RA validation:

- Port on which the frame is received
- IPv6 source address
- Prefix list

The following configuration information created on the switch is available to RA-Guard to validate against the information found in the received RA frame:

- Trusted/Untrusted ports for receiving RA-guard messages
- Trusted/Untrusted IPv6 source addresses of RA-sender
- Trusted/Untrusted Prefix list and Prefix ranges
- Router Preference

RA guard occurs at the switch. You can configure the switch to drop RA messages at the switch. All IPv6 RA messages are dropped, which protects other wireless clients and upstream wired network from malicious IPv6 clients.

```
//Create a policy for RA Guard//
ipv6 nd raguard policy raguard-router
trusted-port
device-role router
//Applying the RA Guard Policy on port/interface//
interface tengigabitethernet1/0/1 (Katana)
interface gigabitethernet1/0/1 (Edison)

ipv6 nd raguard attach-policy raguard-router
```

Related Topics

Configuring RA Guard Policy (CLI), on page 1434

Applying RA Guard Policy (CLI), on page 1435

Configuring RA Throttle Policy (CLI), on page 1436

Applying RA Throttle Policy on VLAN (CLI), on page 1437

RA Throttling

RA throttling allows the controller to enforce limits to RA packets headed toward the wireless network. By enabling RA throttling, routers that send many RA packets can be trimmed to a minimum frequency that will still maintain an IPv6 client connectivity. If a client sends an RS packet, an RA is sent back to the client. This RA is allowed through the controller and unicasted to the client. This process ensures that the new clients or roaming clients are not affected by the RA throttling.

Related Topics

Configuring RA Guard Policy (CLI), on page 1434

Applying RA Guard Policy (CLI), on page 1435

Configuring RA Throttle Policy (CLI), on page 1436

Applying RA Throttle Policy on VLAN (CLI), on page 1437

Configuring IPv6 Unicast (CLI)

IPv6 unicasting must always be enabled on the switch and the controller. IPv6 unicast routing is disabled.

Before You Begin

To enable the forwarding of IPv6 unicast datagrams, use the **ipv6 unicast-routing** command in global configuration mode. To disable the forwarding of IPv6 unicast datagrams, use the **no** form of this command.

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 unicast routing**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ipv6 unicast routing** **Example:** `Switch (config)# ipv6 unicast routing`	enable the forwarding of IPv6 unicast datagrams

Configuring RA Guard Policy (CLI)

Configure RA Guard policy on the switch to add IPv6 client addresses and populate the router table based on IPv6 router advertisement packets.

Before You Begin

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 nd raguard policy** raguard-router
3. **trustedport**
4. **device-role router**
5. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ipv6 nd raguard policy** raguard-router **Example:** `Switch(config)# ipv6 nd raguard policy raguard-router`	Defines the RA guard policy name and enters RA guard policy configuration mode.
Step 3	**trustedport** **Example:** `Switch(config-ra-guard)# trustedport`	(Optional) Specifies that this policy is being applied to trusted ports.
Step 4	**device-role router** **Example:** `Switch(config-ra-guard)# device-role router`	Specifies the role of the device attached to the port.
Step 5	**exit** **Example:** `Switch(config-ra-guard)# exit`	Exits RA guard policy configuration mode and returns to global configuration mode.

Related Topics

Prerequisites for IPv6 Client Address Learning, on page 1427

Applying RA Guard Policy (CLI)

Applying the RA Guard policy on the switch will block all the untrusted RA's.

Before You Begin

SUMMARY STEPS

1. **configure terminal**
2. **interface tengigabitethernet** 1/0/1
3. **ipv6 nd raguard attach-policy** raguard-router
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface tengigabitethernet** 1/0/1 **Example:** `Switch (config)# interface tengigabitethernet 1/0/1`	Specifies an interface type and number, and places the device in interface configuration mode.
Step 3	**ipv6 nd raguard attach-policy** raguard-router **Example:** `Switch(config-if)# ipv6 nd raguard attach-policy raguard-router`	Applies the IPv6 RA Guard feature to a specified interface.
Step 4	**exit** **Example:** `Switch(config-if)# exit`	Exits interface configuration mode.

Related Topics

Configuring RA Guard Policy (CLI), on page 1434

RA Guard, on page 1432
RA Throttling, on page 1433
Configuring RA Throttle Policy (CLI), on page 1436
Applying RA Throttle Policy on VLAN (CLI), on page 1437

Configuring RA Throttle Policy (CLI)

Configure RA Throttle policy to allow the enforce the limits

Before You Begin

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 nd ra-throttler policy** ra-throttler1
3. **throttleperiod**500
4. **max-through**10
5. **allow-atleast** 5 **at-most** 10

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ipv6 nd ra-throttler policy** ra-throttler1 **Example:** `Switch(config)# ipv6 nd ra-throttler policy ra-throttler1`	Define the router advertisement (RA) throttler policy name and enter IPv6 RA throttle policy configuration mode.
Step 3	**throttleperiod**500 **Example:** `Switch(config-nd-ra-throttle)# throttleperiod 500`	Configures the throttle period in an IPv6 RA throttler policy.
Step 4	**max-through**10 **Example:** `Switch(config-nd-ra-throttle)# max-through 500`	Limits multicast RAs per VLAN per throttle period.
Step 5	**allow-atleast** 5 **at-most** 10 **Example:** `Switch(config-nd-ra-throttle)# allow-atleast 5 at-most 10`	Limits the number of multicast RAs per device per throttle period in an RA throttler policy.

Related Topics

Configuring RA Guard Policy (CLI), on page 1434

Applying RA Guard Policy (CLI), on page 1435

RA Guard, on page 1432

RA Throttling, on page 1433

Applying RA Throttle Policy on VLAN (CLI), on page 1437

Applying RA Throttle Policy on VLAN (CLI)

Applying the RA Throttle policy on a VLAN. By enabling RA throttling, routers that send many RA packets can be trimmed to a minimum frequency that will still maintain an IPv6 client connectivity.

Before You Begin

SUMMARY STEPS

1. **configure terminal**
2. **vlan configuration** 1
3. **ipv6 nd ra throttler attach-policy** ra-throttler1

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**vlan configuration** 1 **Example:** `Switch(config)# vlan configuration 1`	Configures a VLAN or a collection of VLANs and enters VLAN configuration mode.
Step 3	**ipv6 nd ra throttler attach-policy** ra-throttler1 **Example:** `Switch(config-vlan)# ipv6 nd ra throttler attach-policy ra-throttler1`	Attaches an IPv6 RA throttler policy to a VLAN or a collection of VLANs.

Related Topics

Configuring RA Guard Policy (CLI), on page 1434
Applying RA Guard Policy (CLI), on page 1435
Configuring RA Throttle Policy (CLI), on page 1436
RA Guard, on page 1432
RA Throttling, on page 1433

Configuring IPv6 Snooping (CLI)

IPv6 snooping must always be enabled on the switch and the controller.

Before You Begin

Enable IPv6 on the client machine.

SUMMARY STEPS

1. **vlan configuration 1**
2. **ipv6 snooping**
3. **ipv6 nd suppress**
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**vlan configuration 1** **Example:** `Switch(config)# vlan configuration 1`	Enters Vlan configuration mode.
Step 2	**ipv6 snooping** **Example:** `Switch(config-vlan)# ipv6 snooping`	Enables IPv6 snooping on the Vlan.
Step 3	**ipv6 nd suppress** **Example:** `Switch(config-vlan-config)# ipv6 nd suppress`	Enables the IPv6 ND suppress on the Vlan.
Step 4	**exit** **Example:** `Switch(config-vlan-config)# exit`	Saves the configuration and comes out of the Vlan configuration mode.

Related Topics

SLAAC Address Assignment, on page 1428
Stateful DHCPv6 Address Assignment, on page 1429

Configuring IPv6 ND Suppress Policy (CLI)

The IPv6 neighbor discovery (ND) multicast suppress feature stops as many ND multicast neighbor solicit (NS) messages as possible by dropping them (and responding to solicitations on behalf of the targets) or converting them into unicast traffic. This feature runs on a layer 2 switch or a wireless controller and is used to reduce the amount of control traffic necessary for proper link operations.

When an address is inserted into the binding table, an address resolution request sent to a multicast address is intercepted, and the device either responds on behalf of the address owner or, at layer 2, converts the request into a unicast message and forwards it to its destination.

Before You Begin

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 nd suppress policy**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch(config)# enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 3	**ipv6 nd suppress policy** **Example:** `Switch (config)# ipv6 nd suppress policy`	Defines the ND suppress policy name and enters ND suppress policy configuration mode.

Related Topics

Router Solicitation, on page 1431
Router Advertisement, on page 1431

Neighbor Discovery, on page 1431
Neighbor Discovery Suppression, on page 1431

Configuring IPv6 Snooping on VLAN/PortChannel

Neighbor Discover (ND) suppress can be enabled or disabled on either the VLAN or a switchport.

Before You Begin

SUMMARY STEPS

1. **vlan config901**
2. **ipv6 nd suppress**
3. **end**
4. **interface gi1/0/1**
5. **ipv6 nd suppress**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**vlan config901** **Example:** `Switch(config)# vlan config901`	Creates a VLAN and enter the VLAN configuration mode
Step 2	**ipv6 nd suppress** **Example:** `Switch(config-vlan)# ipv6 nd suppress`	Applies the IPv6 nd suppress on VLAN.
Step 3	**end** **Example:** `Switch(config-vlan)# end`	Exits vlan configuration mode and enters the global configuration mode.
Step 4	**interface gi1/0/1** **Example:** `Switch (config)# interface gi1/0/1`	Creates a gigabitethernet port interface.
Step 5	**ipv6 nd suppress** **Example:** `Switch(config-vlan)# ipv6 nd suppress`	Applies the IPv6 nd suppress on the interface.
Step 6	**end** **Example:** `Switch(config-vlan)# end`	Exits vlan configuration mode and enters the global configuration mode.

Configuring IPv6 on Switch (CLI)

Use this configuration example to configure IPv6 on an interface.

Before You Begin

Enable IPv6 on the client and IPv6 support on the wired infrastructure.

SUMMARY STEPS

1. **interface vlan** 1
2. **ip address** fe80::1 link-local
3. **ipv6 enable**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**interface vlan** 1 **Example:** `Switch(config)# interface vlan 1`	Creates a interface and enters interface configuration mode.

Step 2	**ip address** fe80::1 link-local
	
	Example:
	`Switch(config-if)# ip address 198.51.100.1 255.255.255.0`
	
	`Switch(config-if)# ipv6 address fe80::1 link-local`
	`Switch(config-if)# ipv6 address 2001:DB8:0:1:FFFF:1234::5/64`
	`Switch(config-if)# ipv6 address 2001:DB8:0:0:E000::F/64`
		Configures IPv6 address on the interface using the link-local option.

Step 3	**ipv6 enable** **Example:** `Switch(config)# ipv6 enable`	(Optional) Enables IPv6 on the interface.
Step 4	**end** **Example:** `Switch(config)# end`	Exits from the interface mode.

Configuring DHCP Pool (CLI)

SUMMARY STEPS

1. **ipv6 dhcp pool** Vlan21
2. **address prefix** 2001:DB8:0:1:FFFF:1234::/64 **lifetime** 300 10
3. **dns-server** 2001:100:0:1::1
4. **domain-name** example.com
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**ipv6 dhcp pool** Vlan21 **Example:** `Switch(config)# ipv6 dhcp pool vlan1`	Enters the configuration mode and configures the IPv6 DHCP pool on the Vlan.
Step 2	**address prefix** 2001:DB8:0:1:FFFF:1234::/64 **lifetime** 300 10 **Example:** `Switch(config-dhcpv6)# address prefix 2001:DB8:0:1:FFFF:1234::/64 lifetime 300 10`	Enters the configuration-dhcp mode and configures the address pool and its lifetime on a Vlan.
Step 3	**dns-server** 2001:100:0:1::1 **Example:** `Switch(config-dhcpv6)# dns-server 2001:20:21::1`	Configures the DNS servers for the DHCP pool.
Step 4	**domain-name** example.com **Example:** `Switch(config-dhcpv6)# domain-name example.com`	Configures the domain name to complete unqualified host names.
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Related Topics

SLAAC Address Assignment, on page 1428
Stateful DHCPv6 Address Assignment, on page 1429

Configuring Stateless Auto Address Configuration Without DHCP (CLI)

SUMMARY STEPS

1. **interface vlan** 1
2. **ip address** fe80::1 link-local
3. **ipv6 enable**
4. **no ipv6 nd managed-config-flag**
5. **no ipv6 nd other-config-flag**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**interface vlan** 1 **Example:** `Switch(config)# interface vlan 1`	Creates a interface and enters interface configuration mode.

Step 2	**ip address** fe80::1 link-local
	Example:
	`Switch(config-if)# ip address 198.51.100.1 255.255.255.0`
	`Switch(config-if)# ipv6 address fe80::1 link-local`
	`Switch(config-if)# ipv6 address 2001:DB8:0:1:FFFF:1234::5/64`
	`Switch(config-if)# ipv6 address 2001:DB8:0:0:E000::F/64`
		Configures IPv6 address on the interface using the link-local option.

Step 3	**ipv6 enable** **Example:** `Switch(config)# ipv6 enable`	(Optional) Enables IPv6 on the interface.

Step 4	**no ipv6 nd managed-config-flag**
	Example:
	`Switch(config)#interface vlan 1`
	`Switch(config-if)# no ipv6 nd managed-config-flag`
		Ensures the attached hosts do not use stateful autoconfiguration to obtain addresses.

Step 5	**no ipv6 nd other-config-flag** **Example:** `Switch(config-if)# no ipv6 nd other-config-flag`	Ensures the attached hosts do not use stateful autoconfiguration to obtain non-address options from DHCP (domain etc).
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Related Topics

SLAAC Address Assignment, on page 1428
Stateful DHCPv6 Address Assignment, on page 1429

Configuring Stateless Auto Address Configuration With DHCP (CLI)

SUMMARY STEPS

1. **interface vlan** 1
2. **ip address** fe80::1 link-local
3. **ipv6 enable**
4. **no ipv6 nd managed-config-flag**
5. **ipv6 nd other-config-flag**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**interface vlan** 1 **Example:** `Switch(config)# interface vlan 1`	Creates a interface and enters interface configuration mode.

Step 2	**ip address** fe80::1 link-local
	
	Example:
	`Switch(config-if)# ip address 198.51.100.1 255.255.255.0`
	`Switch(config-if)# ipv6 address fe80::1 link-local`
	
	`Switch(config-if)# ipv6 address 2001:DB8:0:1:FFFF:1234::5/64`
	`Switch(config-if)# ipv6 address 2001:DB8:0:0:E000::F/64`
		Configures IPv6 address on the interface using the link-local option.

Step 3	**ipv6 enable** **Example:** `Switch(config)# ipv6 enable`	(Optional) Enables IPv6 on the interface.

Step 4	**no ipv6 nd managed-config-flag**
	
	Example:
	`Switch(config)#interface vlan 1`
	`Switch(config-if)# no ipv6 nd managed-config-flag`
		Ensures the attached hosts do not use stateful autoconfiguration to obtain addresses.

	Command or Action	Purpose
Step 5	**ipv6 nd other-config-flag** **Example:** `Switch(config-if)# no ipv6 nd other-config-flag`	Ensures the attached hosts do not use stateful autoconfiguration to obtain non-address options from DHCP (domain etc).
Step 6	**end** **Example:** `Switch(config)# end`	Exits from the interface mode.

Related Topics

SLAAC Address Assignment, on page 1428
Stateful DHCPv6 Address Assignment, on page 1429

Configuring Stateful DHCP Locally (CLI)

This interface configuration is for a Cisco IOS IPv6 router implementing stateful DHCPv6 on a local Switch

Before You Begin

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 unicast-routing**
3. **ipv6 dhcp pool** IPv6_DHCPPOOL
4. **address prefix** 2001:DB8:0:1:FFFF:1234::/64
5. **dns-server** 2001:100:0:1::1
6. **domain-name** example.com
7. **exit**
8. **interface** vlan1
9. **description IPv6-DHCP-Stateful**
10. **ipv6 address** 2001:DB8:0:20::1/64
11. **ip address** 192.168.20.1 255.255.255.0
12. **ipv6 nd prefix** 2001:db8::/64 **no-advertise**
13. **ipv6 nd managed-config-flag**
14. **ipv6 nd other-config-flag**
15. **ipv6 dhcp server** IPv6_DHCPPOOL

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ipv6 unicast-routing** **Example:** `Switch(config)# ipv6 unicast-routing`	Configures IPv6 for unicasting.
Step 3	**ipv6 dhcp pool** IPv6_DHCPPOOL **Example:** `Switch (config)# ipv6 dhcp pool IPv6_DHCPPOOL`	Enters the configuration mode and configures the IPv6 DHCP pool on the VLAN.
Step 4	**address prefix** 2001:DB8:0:1:FFFF:1234::/64 **Example:** `Switch (config-dhcpv6)# address prefix 2001:DB8:0:1:FFFF:1234::/64`	Specifies the address range to provide in the pool.
Step 5	**dns-server** 2001:100:0:1::1 **Example:** `Switch (config-dhcpv6)# dns-server 2001:100:0:1::1`	Provides the DNS server option to DHCP clients.
Step 6	**domain-name** example.com **Example:** `Switch (config-dhcpv6)# domain-name example.com`	Provides the domain name option to DHCP clients.
Step 7	**exit** **Example:** `Switch (config-dhcpv6)# exit`	Returns to the previous mode.
Step 8	**interface** vlan1 **Example:** `Switch (config)# interface vlan 1`	Enters the interface mode to configure the stateful DHCP.
Step 9	**description IPv6-DHCP-Stateful** **Example:** `Switch (config-if)# description IPv6-DHCP-Stateful`	Enter description for the stateful IPv6 DHCP.
Step 10	**ipv6 address** 2001:DB8:0:20::1/64 **Example:** `Switch (config-if)# ipv6 address 2001:DB8:0:20::1/64`	Enters the IPv6 address for the stateful IPv6 DHCP.

	Command or Action	Purpose
Step 11	**ip address** 192.168.20.1 255.255.255.0 **Example:** `Switch (config-if)# ip address 192.168.20.1 255.255.255.0`	Enters the IPv6 address for the stateful IPv6 DHCP.
Step 12	**ipv6 nd prefix** 2001:db8::/64 **no-advertise** **Example:** `Switch (config-if)# ipv6 nd prefix 2001:db8::/64 no-advertise`	Configures the IPv6 routing prefix advertisement that must not be advertised.
Step 13	**ipv6 nd managed-config-flag** **Example:** `Switch (config-if)# ipv6 nd managed-config-flag`	Configures IPv6 interfaces neighbor discovery to allow the hosts to uses DHCP for address configuration.
Step 14	**ipv6 nd other-config-flag** **Example:** `Switch (config-if)# ipv6 nd other-config-flag`	Configures IPv6 interfaces neighbor discovery to allow the hosts to uses DHCP for non-address configuration.
Step 15	**ipv6 dhcp server** IPv6_DHCPPOOL **Example:** `Switch (config-if)# ipv6 dhcp server IPv6_DHCPPOOL`	Configures the DHCP server on the interface.

Related Topics

SLAAC Address Assignment, on page 1428
Stateful DHCPv6 Address Assignment, on page 1429

Configuring Stateful DHCP Externally (CLI)

This interface configuration is for a Cisco IOS IPv6 router implementing stateful DHCPv6 on an external DHCP server.

Before You Begin

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 unicast-routing**
3. **dns-server** 2001:100:0:1::1
4. **domain-name** example.com
5. **exit**
6. **interface** vlan1
7. **description IPv6-DHCP-Stateful**
8. **ipv6 address** 2001:DB8:0:20::1/64
9. **ip address** 192.168.20.1 255.255.255.0
10. **ipv6 nd prefix** 2001:db8::/64 **no-advertise**
11. **ipv6 nd managed-config-flag**
12. **ipv6 nd other-config-flag**
13. **ipv6 dhcp_relaydestination** 2001:DB8:0:20::2

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ipv6 unicast-routing** **Example:** `Switch(config)# ipv6 unicast-routing`	Configures the IPv6 for unicasting.
Step 3	**dns-server** 2001:100:0:1::1 **Example:** `Switch (config-dhcpv6)# dns-server 2001:100:0:1::1`	Provides the DNS server option to DHCP clients.
Step 4	**domain-name** example.com **Example:** `Switch (config-dhcpv6)# domain-name example.com`	Provides the domain name option to DHCP clients.
Step 5	**exit** **Example:** `Switch (config-dhcpv6)# exit`	Returns to the previous mode.

	Command or Action	Purpose
Step 6	**interface** vlan1 **Example:** `Switch (config)# interface vlan 1`	Enters the interface mode to configure the stateful DHCP.
Step 7	**description IPv6-DHCP-Stateful** **Example:** `Switch (config-if)# description IPv6-DHCP-Stateful`	Enter description for the stateful IPv6 DHCP.
Step 8	**ipv6 address** 2001:DB8:0:20::1/64 **Example:** `Switch (config-if)# ipv6 address 2001:DB8:0:20::1/64`	Enters the IPv6 address for the stateful IPv6 DHCP.
Step 9	**ip address** 192.168.20.1 255.255.255.0 **Example:** `Switch (config-if)# ip address 192.168.20.1 255.255.255.0`	Enters the IPv6 address for the stateful IPv6 DHCP.
Step 10	**ipv6 nd prefix** 2001:db8::/64 **no-advertise** **Example:** `Switch (config-if)# ipv6 nd prefix 2001:db8::/64 no-advertise`	Configures the IPv6 routing prefix advertisement that must not be advertised.
Step 11	**ipv6 nd managed-config-flag** **Example:** `Switch (config-if)# ipv6 nd managed-config-flag`	Configures IPv6 interfaces neighbor discovery to allow the hosts to uses DHCP for address configuration.
Step 12	**ipv6 nd other-config-flag** **Example:** `Switch (config-if)# ipv6 nd other-config-flag`	Configures IPv6 interfaces neighbor discovery to allow the hosts to uses DHCP for non-address configuration.
Step 13	**ipv6 dhcp_relaydestination** 2001:DB8:0:20::2 **Example:** `Switch (config-if)# ipv6 dhcp_relay destination 2001:DB8:0:20::2`	Configures the DHCP server on the interface.

Related Topics

SLAAC Address Assignment, on page 1428
Stateful DHCPv6 Address Assignment, on page 1429

Monitoring IPv6 Clients (GUI)

To view the IPv6 clients associated with the Switch

Before You Begin

Select **Monitor** > **Clients**
The Clients page is displayed. The Clients page contains the following details:

- Client MAC Address— Displays the MAC address of the client.
- AP Name— Displays the access point name to which the client is connected to.
- WLAN— Displays the WLAN associated with the client.
- State— Displays the client authentication.
- Protocol— Displays the protocol used.

To view the client related general details, click the **Client MAC Address** parameter in the Clients page. The **Client > Detail** page displays IPv6 addresses of the client under the **General** tab.

Verifying IPv6 Address Learning Configuration

This example displays the output of the **show ipv6 dhcp pool** command. This command displays the IPv6 service configuration on the switch. The vlan 21 configured pool detail displays 6 clients that are currently using addresses from the pool.

SUMMARY STEPS

1. **show ipv6 dhcp pool**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**show ipv6 dhcp pool**
	
	Example:
	`Switchshow ipv6 dhcp pool`
	`DHCPv6 pool: vlan21`
	`Address allocation prefix: 2001:DB8:0:1:FFFF:1234::/64 valid 86400`
	` preferred 86400 (6 in use, 0 conflicts)`
	`DNS server: 2001:100:0:1::1`
	`Domain name: example.com`
	`Active clients: 6`
		Displays the IPv6 service configuration on the switch.

Additional References

Related Documents

Related Topic	Document Title
IPv6 command reference	*IPv6 Command Reference (Catalyst 3850 Switches)*
IP command reference	*IP Command Reference (Catalyst 3850 Switches)*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for IPv6 Client Address Learning

This table lists the features in this module and provides links to specific configuration information:

Feature	Release	Modification
IPv6 Client Address Learning Functionality	Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 67

Configuring IPv6 WLAN Security

Prerequisites for IPv6 WLAN Security

A client VLAN must be mapped to the WLAN configured on the switch

Restrictions for IPv6 WLAN Security

RADIUS Server Support

- If multiple RADIUS servers are configured for redundancy, the user database must be identical in all the servers for the backup to work properly.

Radius ACS Support

- You must configure RADIUS on both your Cisco Secure Access Control Server (ACS) and your switch
- RADIUS is supported on Cisco Secure ACS version 3.2 and later releases.

Information About IPv6 WLAN Security

Information About RADIUS

Remote Authentication Dial-In User Service (RADIUS) is a client/server protocol that provides centralized security for users attempting to gain management access to a network. It serves as a back-end database similar to Local EAP and provides authentication and accounting services.

- Authentication—The process of verifying users when they attempt to log into the switch

  Users must enter a valid username and password for the switch to authenticate users to the RADIUS server. If multiple databases are configured, then specify the sequence in which the backend database must be tried.

- Accounting— The process of recording user actions and changes.

  Whenever a user successfully executes an action, the RADIUS accounting server logs the changed attributes, the user ID of the person who made the change, the remote host where the user is logged in, the date and time when the command was executed, the authorization level of the user, and a description of the action performed and the values provided. If the RADIUS accounting server is unreachable, the users can continue their sessions uninterrupted.

User Datagram Protocol— RADIUS uses User Datagram Protocol (UDP) for its transport. It maintains a database and listens on UDP port 1812 for incoming authentication requests and UDP port 1813 for incoming accounting requests. The switch, which requires access control, acts as the client and requests AAA services from the server. The traffic between the switch and the server is encrypted by an algorithm defined in the protocol and a shared secret key configured on both devices.

Configures multiple RADIUS accounting and authentication servers. For example, you can have one central RADIUS authentication server but several RADIUS accounting servers in different regions. If you configure multiple servers of the same type and the first one fails or becomes unreachable, the controller automatically tries the second one, then the third one if necessary, and so on.

When RADIUS method is configured for the WLAN, the switch will use the RADIUS method configured for the WLAN. When the WLAN is configured to use local EAP, the RADIUS method configured on the WLAN points to Local. The WLAN must also be configured with the name of the local EAP profile to use.

If no RADIUS method is configured in the WLAN, the switch will use the default RADIUS method defined in global mode.

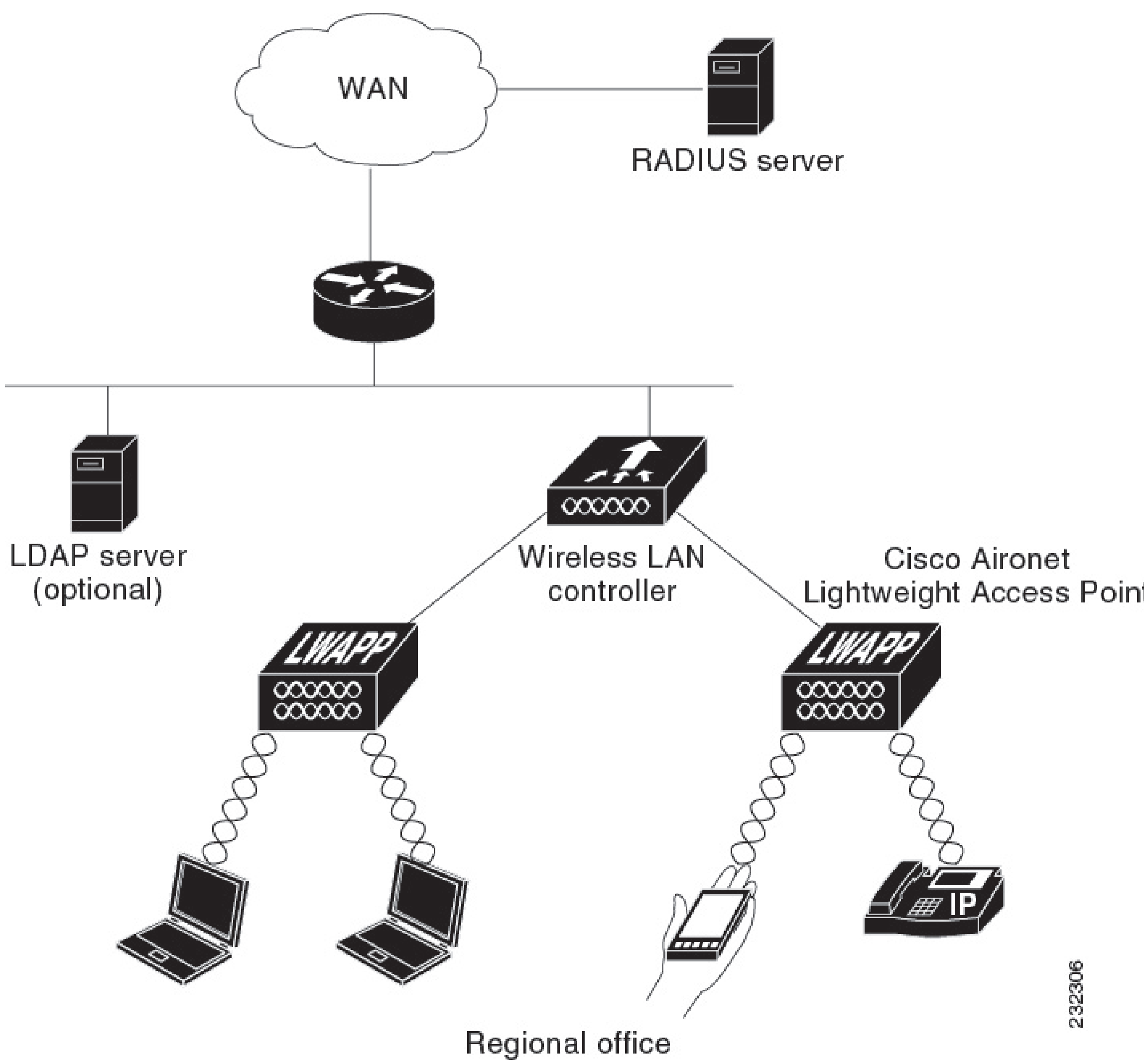
Information About Local EAP

Local EAP is an authentication method that allows users and wireless clients to be authenticated locally. It is designed for use in remote offices that maintain connectivity to wireless clients when the back-end system is disrupted or the external authentication server goes down. When you enable local EAP, the switch serves as the authentication server and the local user database, which removes dependence on an external authentication server. Local EAP retrieves user credentials from the local user database or the LDAP back-end database to authenticate users. Local EAP supports LEAP, EAP-FAST, EAP-TLS, PEAPv0/MSCHAPv2, and PEAPv1/GTC authentication between the controller and wireless clients.

Note The LDAP back-end database supports these local EAP methods: EAP-TLS, EAP-FAST/GTC, and PEAPv1/GTC. LEAP, EAP-FAST/MSCHAPv2, and PEAPv0. MSCHAPv2 is supported only if the LDAP server is set up to return a clear-text password.

Note Switch support Local EAP authentication against external LDAP databases such as Microsoft Active Directory and Novell's eDirectory. For more information about configuring the controller for Local EAP authentication against Novell's eDirectory, see the Configure Unified Wireless Network for Authentication Against Novell's eDirectory Database whitepaper.

Figure 69: Local EAP Example



Related Topics

Creating a Local User, on page 1456
Creating an Client VLAN and Interface, on page 1457
Configuring a EAP Profile, on page 1458
Creating a Client VLAN, on page 1471
Creating 802.1x WLAN Using an External RADIUS Server, on page 1472

How to Configure IPv6 WLAN Security

Configuring Local Authentication

Creating a Local User

SUMMARY STEPS

1. **configure terminal**
2. **username** aaa_test
3. **password 0** aaa_test
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.
Step 2	**username** aaa_test **Example:** `Switch(config)# username aaa_test`	Creates a username.
Step 3	**password 0** aaa_test **Example:** `Switch(config)# usernameaaa_test password 0 aaa_test`	Assigns a password for the username.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit the global configuration mode.

```
Switch# configure terminal
Switch(config)# username aaa_test password 0 aaa_test
Switch(config)# end
```

Related Topics

Information About IPv6 WLAN Security, on page 1454

Creating an Client VLAN and Interface

SUMMARY STEPS

1. **configure terminal**
2. **vlan**
3. **exit**
4. **interface vlan** vlan_ID
5. **ip address**
6. **ipv6 address**
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.
Step 2	**vlan** **Example:** `Switch(config)# vlan 137`	Creates a VLAN.
Step 3	**exit** **Example:** `Switch (config-vlan)# exit`	Exits VLAN configuration mode.
Step 4	**interface vlan** vlan_ID **Example:** `Switch (config)# interface vlan 137`	Associates the VLAN to an interface.
Step 5	**ip address** **Example:** `Switch(config-if)# ip address 10.7.137.10 255.255.255.0`	Assigns an IP address to the VLAN interface.

	Command or Action	Purpose
Step 6	**ipv6 address** **Example:** `Switch(config-if)#ipv6 address 2001:db8::20:1/64`	Assigns an IPv6 address to the VLAN interface.
Step 7	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit the global configuration mode.

```
Switch# configure terminal
Switch(config)# vlan 137
Switch(config-vlan)#exit
Switch(config)#interface vlan 137
Switch(config-if)#ip address 10.7.137.10 255.255.255.0
Switch(config-if)#ipv6 address 2001:db8::20:1/64
Switch(config-if)#end
```

Related Topics

Information About IPv6 WLAN Security, on page 1454

Configuring a EAP Profile

SUMMARY STEPS

1. **eap profile** name
2. **method leap**
3. **method tls**
4. **method peap**
5. **method mschapv2**
6. **method md5**
7. **method gtc**
8. **method fast profile** my-fast
9. **description** my_local**eap profile**
10. **exit**
11. **eap method fast profile**myFast
12. **authority-id** [identity|information]
13. **local-key 0** key-name
14. **pac-password 0** password
15. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**eap profile** name **Example:** `Switch(config)# eap profile wcm_eap_prof`	Creates a EAP profile.
Step 2	**method leap** **Example:** `Switch(config-eap-profile)# method leap`	Configures EAP-LEAP method on the profile.
Step 3	**method tls** **Example:** `Switch(config-eap-profile)# method tls`	Configures EAP-TLS method on the profile.
Step 4	**method peap** **Example:** `Switch(config-eap-profile)# method peap`	Configures PEAP method on the profile.
Step 5	**method mschapv2** **Example:** `Switch(config-eap-profile)# method mschapv2`	Configures EAP-MSCHAPV2 method on the profile.
Step 6	**method md5** **Example:** `Switch(config-eap-profile)# method md5`	Configures EAP-MD5 method on the profile.
Step 7	**method gtc** **Example:** `Switch(config-eap-profile)# method gtc`	Configures EAP-GTC method on the profile.

Step 8	**method fast profile** my-fast
	
	Example:
	`Switch(config-eap-profile)# eap method fast profile my-fast`
	`Switch (config-eap-profile)#description my_local eap profile`
		Creates a EAP profile named my-fast.

Step 9	**description** my_local**eap profile** **Example:** `Switch (config-eap-profile)#description my_local eap profile`	Provides a description for the local profile.
Step 10	**exit** **Example:** `Switch (config-eap-profile)# exit`	Exits the eap-profile configuration mode.

	Command or Action	Purpose
Step 11	**eap method fast profile**myFast **Example:** `Switch (config)# eap method fast profile myFast`	Configures the EAP method profile.

Step 12	**authority-id** [identity|information]
	
	Example:
	`Switch(config-eap-method-profile)# authority-id identity my_identity`
	`Switch(config-eap-method-profile)#authority-id information my_information`
		Configure the authority ID and information for the EAP method profile.

Step 13	**local-key 0** key-name **Example:** `Switch(config-eap-method-profile)# local-key 0 test`	Configures the local server key.
Step 14	**pac-password 0** password **Example:** `Switch(config-eap-method-profile)# pac-password 0 test`	Configures the PAC password for manual PAC provisioning.
Step 15	**end** **Example:** `Switch(config)#` **`end`**	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit the global configuration mode.

```
Switch(config)#eap profile wcm_eap_prof
Switch(config-eap-profile)#method leap
Switch(config-eap-profile)#method tls
Switch(config-eap-profile)#method peap
Switch(config-eap-profile)#method mschapv2
Switch(config-eap-profile)#method md5
Switch(config-eap-profile)#method gtc
Switch(config-eap-profile)#eap method fast profile my-fast
Switch (config-eap-profile)#description my_local eap profile
Switch(config-eap-profile)# exit
Switch (config)# eap method fast profile myFast
Switch(config-eap-method-profile)#authority-id identity my_identity
Switch(config-eap-method-profile)#authority-id information my_information
Switch(config-eap-method-profile)#local-key 0 test
Switch(config-eap-method-profile)#pac-password 0 test
Switch(config-eap-method-profile)# end
```

Related Topics

Information About IPv6 WLAN Security, on page 1454

Creating a Local Authentication Model

SUMMARY STEPS

1. **aaa** new-model
2. **authentication dot1x default local**
3. **dot1x** method_list **local**
4. **aaa authentication dot1x** dot1x_name **local**
5. **aaa authorization credential-download** name **local**
6. **aaa local authentication** auth-name **authorization** authorization-name
7. **session ID**
8. **dot1x system-auth-control**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**aaa** new-model **Example:** `Switch(config)# aaa new-model`	Creates a AAA authentication model.
Step 2	**authentication dot1x default local** **Example:** `Switch(config)# aaa authentication dot1x default local`	Implies that the dot1x must use the default local RADIUS when no other method is found.
Step 3	**dot1x** method_list **local** **Example:** `Switch(config)# aaa authentication dot1x wcm_local local`	Assigns the local authentication for wcm_local method list.
Step 4	**aaa authentication dot1x** dot1x_name **local** **Example:** `Switch(config)# aaa authentication dot1x aaa_auth local`	Configures the local authentication for the dot1x method.
Step 5	**aaa authorization credential-download** name **local** **Example:** `Switch(config)# aaa authorization credential-download wcm_author local`	Configures local database to download EAP credentials from Local/RADIUS/LDAP.
Step 6	**aaa local authentication** auth-name **authorization** authorization-name **Example:** `Switch(config)# aaa local authentication wcm_local authorization wcm_author`	Selects local authentication and authorization.

	Command or Action	Purpose
Step 7	**session ID** **Example:** `Switch(config)# aaa session-id common`	Configures a session ID for AAA.
Step 8	**dot1x system-auth-control** **Example:** `Switch(config)# dot1x system-auth-control`	Enables dot.1x system authentication control.

```
Switch(config)# aaa new-model
Switch(config)# aaa authentication dot1x default local
Switch(config)# aaa authentication dot1x wcm-local local
Switch(config)# aaa authentication dot1x aaa_auth local
Switch(config)# aaa authorization credential-download wcm_author local
Switch(config)# aaa local authentication wcm_local authorization wcm_author
Switch(config)# aaa session-id common
Switch(config)# dot1x system-auth-control
```

Creating a Client WLAN

Note This example uses 802.1x with dynamic WEP. You can use any other security mechanism supported by the wireless client and configurable on the switch

SUMMARY STEPS

1. **configure terminal**
2. **wlan** wlan name **<identifier>** SSID
3. **broadcast-ssid**
4. **no security wpa**
5. **security dot1x**
6. **security** dot1x authentication-list *wcm-local*
7. **local-auth** wcm_eap_prof
8. **client vlan** *137*
9. **no shutdown**
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.
Step 2	**wlan** wlan name **<identifier>** SSID **Example:** `Switch(config)# wlan wlanProfileName 1 ngwcSSID`	Creates a WLAN.
Step 3	**broadcast-ssid** **Example:** `Switch(config-wlan)# broadcast-ssid`	Configures to broadcast the SSID on a WLAN.
Step 4	**no security wpa** **Example:** `Switch(config-wlan)# no security wpa`	Disables the wpa for WLAN to enable 802.1x.
Step 5	**security dot1x** **Example:** `Switch(config-wlan)# security dot1x`	Configures the 802.1x encryption security for the WLAN.
Step 6	**security** dot1x authentication-list *wcm-local* **Example:** `Switch(config-wlan)# security dot1x authentication-list wcm-local`	Configures the server group mapping to the WLAN for dot1x authentication.
Step 7	**local-auth** wcm_eap_prof **Example:** `Switch (config-wlan)# local-auth wcm_eap_profile`	Configures the eap profile on the WLAN for local authentication.
Step 8	**client vlan** *137* **Example:** `Switch(config-wlan)# client vlan 137`	Associates the VLAN to a WLAN.
Step 9	**no shutdown** **Example:** `Switch(config-wlan)# no shutdown`	Enables the WLAN.
Step 10	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit the global configuration mode.

```
Switch# config terminal
Switch(config)#wlan wlanProfileName 1 ngwcSSID
Switch(config-wlan)#broadcast-ssid
Switch(config-wlan)#no security wpa
Switch(config-wlan)#security dot1x
Switch(config-wlan)#security dot1x authentication-list wcm-local
Switch (config-wlan)# local-auth wcm_eap_prof
Switch(config-wlan)#client vlan 137
Switch(config-wlan)#no shutdown
Switch(config-wlan)#end
Switch#
```

Related Topics

Creating Client VLAN for WPA2+AES, on page 1465

Configuring Local Authentication with WPA2+AES

SUMMARY STEPS

1. **configure terminal**
2. **aaa** new model
3. **dot1x system-auth-control**
4. **aaa authentication dot1x default local**
5. **aaa local authorization credential-download default local**
6. **aaa local authentication default authorization default**
7. **eap profile** wcm_eap_profile
8. **method leap**
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.
Step 2	**aaa** new model **Example:** `Switch(config)# aaa new-model`	Creates a AAA authentication model.
Step 3	**dot1x system-auth-control** **Example:** `Switch(config)# dot1x system-auth-control`	Enables dot1x system authentication control.

	Command or Action	Purpose
Step 4	**aaa authentication dot1x default local** **Example:** `Switch(config)# aaa authentication dot1x default local`	Configures the local authentication for the default dot1x method.
Step 5	**aaa local authorization credential-download default local** **Example:** `Switch(config)# aaa authorization credential-download default local`	Configures default database to download EAP credentials from local server.
Step 6	**aaa local authentication default authorization default** **Example:** `Switch(config)# aaa local authentication default authorization default`	Selects the default local authentication and authorization.
Step 7	**eap profile** wcm_eap_profile **Example:** `Switch(config)#eap profile` **`wcm_eap_profile`**	Creates an EAP profile.
Step 8	**method leap** **Example:** `Switch(config)# method leap`	Configures EAP-LEAP method on the profile.
Step 9	**end** **Example:** `Switch(config)#` **`end`**	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit the global configuration mode.

```
Switch# configure terminal
Switch(config)# aaa new-model
Switch(config)# dot1x system-auth-control
Switch(config)# aaa authentication dot1x default local
Switch(config)# aaa authorization credential-download default local
Switch(config)# aaa local authentication default authorization default
Switch(config)#eap profile wcm_eap_profile
Switch(config)# method leap
Switch(config)# end
```

Creating Client VLAN for WPA2+AES

Create a VLAN for the WPA2+AES type of local authentication. This VLAN is later mapped to a WLAN.

SUMMARY STEPS

1. **configure terminal**
2. **vlan** vlan_ID
3. **exit**
4. **interface** vlan vlan_ID
5. **ip address**
6. **ipv6 address**
7. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.
Step 2	**vlan** vlan_ID **Example:** `Switch (config)# vlan 105`	Creates a VLAN.
Step 3	**exit** **Example:** `Switch (config-vlan)# exit`	Exits from the VLAN mode.
Step 4	**interface** vlan vlan_ID **Example:** `Switch(config)# interface vlan 105`	Associates the VLAN to the interface.
Step 5	**ip address** **Example:** `Switch(config-if)# ip address  10.8.105.10 255.255.255.0`	Assigns IP address to the VLAN interface.
Step 6	**ipv6 address** **Example:** `Switch(config-if)#ipv6 address 2001:db8::10:1/64`	Assigns IPv6 address to the VLAN interface.
Step 7	**exit** **Example:** `Switch (config-if)# exit`	Exits from the interface mode.

```
Switch# configure terminal
Switch(config)# vlan105
Switch (config-vlan)# exit
Switch (config)# interface vlan 105
Switch(config-if)#ip address 10.8.105.10 255.255.255.0
Switch(config-if)#ipv6 address 2001:db8::10:1/64
Switch(config-if)#exit
Switch(config)#
```

Related Topics

Creating a Client WLAN , on page 1462

Creating WLAN for WPA2+AES

Create a WLAN and map it to the client VLAN created for WPA2+AES.

SUMMARY STEPS

1. **configure terminal**
2. **wlan** wpas2-aes-wlan 1 wpas2-aes-wlan
3. **client vlan** 105
4. **local-auth** wcm_eap_profile
5. **security dot1x authentication-list default**
6. **no shutdown**
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.

Step 2	**wlan** wpas2-aes-wlan 1 wpas2-aes-wlan
	
	Example:
	`Switch(config)#wlan wpa2-aes-wlan 1 wpa2-aes-wlan`
	`Switch(config-wlan)#`
		Creates a WLAN.

Step 3	**client vlan** 105
	
	Example:
	`Switch(config-wlan)#client vlan 105`
	`Switch(config-wlan)#`
		Maps the WLAN to the client VLAN.

Step 4	**local-auth** wcm_eap_profile **Example:** `Switch(config-wlan)#local-auth wcm_eap_profile`	Creates and sets the EAP profile on the WLAN.

	Command or Action	Purpose
Step 5	**security dot1x authentication-list default** **Example:** `Switch(config-wlan)#security dot1x authentication-list default`	Uses the default dot1x authentication list.

Step 6	**no shutdown**
	
	Example:
	`Switch(config-wlan)#no shutdown`
	`Switch(config-wlan)#`
		Enables the WLAN.

Step 7	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

```
Switch# configure terminal
Switch(config)#wlan wpa2-aes-wlan 1 wpa2-aes-wlan
Switch(config-wlan)#client vlan 105
Switch(config-wlan)#local-auth wcm_eap_profile
Switch(config-wlan)#security dot1x authentication-list default
Switch(config-wlan)#no shutdown
Switch(config-wlan)# exit
```

Configuring External RADIUS Server

Configuring RADIUS Authentication Server Host

SUMMARY STEPS

1. **configure terminal**
2. **radius server One**
3. **address ipv4** address **auth-port**auth_port_number **acct-port** acct_port_number
4. **address ipv6** address **auth-port**auth_port_number **acct-port** acct_port_number
5. **key** 0cisco
6.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.

	Command or Action	Purpose
Step 2	**radius server One** **Example:** `Switch (config)# radius server One`	Creates a radius server.
Step 3	**address ipv4** address **auth-port**auth_port_number **acct-port** acct_port_number **Example:** `Switch (config-radius-server)# address ipv4 10.10.10.10 auth-port 1812 acct-port 1813`	Configures the IPv4 address for the radius server.
Step 4	**address ipv6** address **auth-port**auth_port_number **acct-port** acct_port_number **Example:** `Switch (config-radius-server)# address ipv6 2001:db8::25:2 auth-port 1812 acct-port 1813`	Configures the IPv6 address for the radius server.
Step 5	**key** 0cisco **Example:** `Switch (config-radius-server)# key 0 cisco`	**exit**
Step 6	**Example:** `Switch (config-radius-server)# exit`	Exits from the radius server mode.

```
Switch# configure terminal
Switch (config)# radius server One
Switch (config-radius-server)# address ipv4 10.10.10.10 auth-port 1812 acct-port 1813
Switch (config-radius-server)# address ipv6 2001:db8::25:2 auth-port 1812 acct-port 1813
Switch (config-radius-server)# key 0 cisco
Switch (config-radius-server)#exit
```

Related Topics

Configuring RADIUS Authentication Server Group , on page 1469

Configuring RADIUS Authentication Server Group

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **aaa group server radius** wcm_rad
4. **server** <ip address>**auth-port**1812**acct-port**1813
5. **aaa authentication dot1x** method_list **group** wcm_rad
6. **dot1x system-auth-control**
7. **aaa session-idcommon**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.
Step 2	**aaa new-model** **Example:** `Switch(config)#aaa new-model`	Creates a AAA authentication model.

Step 3	**aaa group server radius** wcm_rad
	
	Example:
	`Switch(config)# aaa group server radius wcm_rad`
	`Switch(config-sg-radius)#`
		Creates an radius server-group.

Step 4	**server** <ip address>**auth-port**1812**acct-port**1813
	
	Example:
	`Switch(config-sg-radius)# server One auth-port 1812 acct-port 1813`
	`Switch(config-sg-radius)# server Two auth-port 1812 acct-port 1813`
	`Switch(config-sg-radius)# server Three auth-port 1812 acct-port 1813`
		Adds servers to the radius group created in Step 3. Configures the UDP port for RADIUS accounting server and authentication server.

Step 5	**aaa authentication dot1x** method_list **group** wcm_rad **Example:** `Switch(config)# aaa authentication dot1x method_list group wcm_rad`	Maps the method list to the radius group.
Step 6	**dot1x system-auth-control** **Example:** `Switch(config)# dot1x system-auth-control`	Enables the system authorization control for the radius group.
Step 7	**aaa session-idcommon** **Example:** `Switch(config)# aaa session-id common`	Ensures that all session IDs information sent out, from the radius group, for a given call are identical.

```
Switch# configure terminal
Switch(config)# aaa new-model
Switch(config)# aaa group server radius wcm_rad
Switch(config-sg-radius)# server One auth-port 1812 acct-port 1813
Switch(config-sg-radius)# server Two auth-port 1812 acct-port 1813
Switch(config-sg-radius)# server Three auth-port 1812 acct-port 1813
Switch(config)# aaa authentication dot1x method_list group wcm_rad
Switch(config)# dot1x system-auth-control
Switch(config)# aaa session-id common
Switch(config)#
```

Related Topics

Configuring RADIUS Authentication Server Host , on page 1468

Creating a Client VLAN

SUMMARY STEPS

1. **configure terminal**
2. **vlan** 137
3. **exit**
4. **interface vlan** 137
5. **ip address** 10.7.137.10 255.255.255.0
6. **ipv6 address** 2001:db8::30:1/64
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.
Step 2	**vlan** 137 **Example:** `Switch(config)# vlan 137`	Creates a VLAN and associate it to the interface.
Step 3	**exit** **Example:** `Switch (config-vlan)# exit`	Exits from the VLAN mode.
Step 4	**interface vlan** 137 **Example:** `Switch (config)# interface vlan 137`	Assigns a VLAN to an interface.
Step 5	**ip address** 10.7.137.10 255.255.255.0 **Example:** `Switch(config-if)# ip address 10.7.137.10 255.255.255.0`	Assigns an IPv4 address to the VLAN interface.
Step 6	**ipv6 address** 2001:db8::30:1/64 **Example:** `Switch(config-if)# ipv6 address 2001:db8::30:1/64`	Assigns an IPv6 address to the VLAN interface.

	Command or Action	Purpose
Step 7	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

```
Switch# configure terminal
Switch(config)# vlan137
Switch(config-vlan)# exit
Switch(config)# interface vlan137
Switch(config-if)# ip address 10.7.137.10 255.255.255.0
Switch(config-if)# ipv6 address 2001:db8::30:1/64
Switch(config-if)# end
```

Related Topics

Information About IPv6 WLAN Security, on page 1454
Creating 802.1x WLAN Using an External RADIUS Server, on page 1472

Creating 802.1x WLAN Using an External RADIUS Server

SUMMARY STEPS

1. **configure terminal**
2. **wlan** ngwc-1x<**ssid**>ngwc-1x
3. **broadcast-ssid**
4. **no security wpa**
5. **security dot1x**
6. **security dot1x authentication-list** wcm-rad
7. **client vlan** 137
8. **no shutdown**
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global command mode.
Step 2	**wlan** ngwc-1x<**ssid**>ngwc-1x **Example:** `Switch(config)# wlan ngwc_8021x 2 ngwc_8021x`	Creates a new WLAN for 802.1x authentication.

	Command or Action	Purpose
Step 3	**broadcast-ssid** **Example:** `Switch(config-wlan)# broadcast-ssid`	Configures to broadcast the SSID on WLAN.
Step 4	**no security wpa** **Example:** `Switch(config-wlan)# no security wpa`	Disables the WPA for WLAN to enable 802.1x.
Step 5	**security dot1x** **Example:** `Switch(config-wlan)# security dot1x`	Configures the 802.1x encryption security for the WLAN.
Step 6	**security dot1x authentication-list** wcm-rad **Example:** `Switch(config-wlan)# security dot1x authentication-list wcm-rad`	Configures the server group mapping to the WLAN for dot1x authentication.
Step 7	**client vlan** 137 **Example:** `Switch(config-wlan)# client vlan 137`	Associates the VLAN to a WLAN.
Step 8	**no shutdown** **Example:** `Switch(config-wlan)# no shutdown`	Enables the WLAN.
Step 9	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit the global configuration mode.

```
Switch# configure terminal
Switch(config)#wlan ngwc_8021x 2 ngwc_8021x
Switch(config-wlan)# broadcast-ssid
Switch(config-wlan)# no security wpa
Switch(config-wlan)# security dot1x
Switch(config-wlan)# security dot1x authentication-list wcm-rad
Switch(config-wlan)# client vlan 137
Switch(config-wlan)# no shutdown
Switch(config-wlan)# end
```

Related Topics

Creating a Client VLAN, on page 1471
Information About IPv6 WLAN Security, on page 1454

Additional References

Related Documents

Related Topic	Document Title
IPv6 command reference	*IPv6 Command Reference (Catalyst 3850 Switches)*
WLAN command reference	*WLAN Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*
WLAN configuration	*WLAN Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for IPv6 WLAN Security

This table lists the features in this module and provides links to specific configuration information:

Feature	Release	Modification
IPv6 WLAN Security Functionality	Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 68

Configuring IPv6 ACL

Prerequisites for IPv6 ACL

You can filter IP Version 6 (IPv6) traffic by creating IPv6 access control lists (ACLs) and applying them to interfaces similarly to the way that you create and apply IP Version 4 (IPv4) named ACLs. You can also create and apply input router ACLs to filter Layer 3 management traffic when the switch is running the IP base feature set.

Related Topics

Creating IPv6 ACL, on page 1481

Restrictions for IPv6 ACL

With IPv4, you can configure standard and extended numbered IP ACLs, named IP ACLs, and MAC ACLs. IPv6 supports only named ACLs.

The switch supports most of the Cisco IOS-supported IPv6 ACLs with some exceptions:

- The switch does not support matching on these keywords: **flowlabel**, **routing header**, and **undetermined-transport**.

- The switch does not support reflexive ACLs (the **reflect** keyword).
- The switch does not apply MAC-based ACLs on IPv6 frames.
- When configuring an ACL, there is no restriction on keywords entered in the ACL, regardless of whether or not they are supported on the platform. When you apply the ACL to an interface that requires hardware forwarding (physical ports or SVIs), the switch checks to determine whether or not the ACL can be supported on the interface. If not, attaching the ACL is rejected.
- If an ACL is applied to an interface and you attempt to add an access control entry (ACE) with an unsupported keyword, the switch does not allow the ACE to be added to the ACL that is currently attached to the interface

Information About IPv6 ACL

An access control list (ACL) is a set of rules used to limit access to a particular interface (for example, if you want to restrict a wireless client from pinging the management interface of the controller). ACLs are configured on the switchnd applied to the management interface, the AP-manager interface, any of the dynamic interfaces, or a WLAN to control data traffic to and from wireless clients or to the controller central processing unit (CPU) to control all traffic destined for the CPU.

You can also create a preauthentication ACL for web authentication. Such an ACL is used to allow certain types of traffic before authentication is complete.

IPv6 ACLs support the same options as IPv4 ACLs including source, destination, source and destination ports.

Note You can enable only IPv4 traffic in your network by blocking IPv6 traffic. That is, you can configure an IPv6 ACL to deny all IPv6 traffic and apply it on specific or all WLANs.

Understanding IPv6 ACLs

A switch supports two types of IPv6 ACLs:

- IPv6 router ACLs are supported on outbound or inbound traffic on Layer 3 interfaces, which can be routed ports, switch virtual interfaces (SVIs), or Layer 3 EtherChannels. IPv6 router ACLs apply only to IPv6 packets that are routed.
- IPv6 port ACLs are supported on inbound traffic on Layer 2 interfaces only. IPv6 port ACLs are applied to all IPv6 packets entering the interface.

A switch running the IP base feature set supports only input router IPv6 ACLs. It does not support port ACLs or output IPv6 router ACLs.

Note If you configure unsupported IPv6 ACLs, an error message appears and the configuration does not take affect.

The switch does not support VLAN ACLs (VLAN maps) for IPv6 traffic.

You can apply both IPv4 and IPv6 ACLs to an interface. As with IPv4 ACLs, IPv6 port ACLs take precedence over router ACLs:

- When an input router ACL and input port ACL exist in an SVI, packets received on ports to which a port ACL is applied are filtered by the port ACL. Routed IP packets received on other ports are filtered by the router ACL. Other packets are not filtered.
- When an output router ACL and input port ACL exist in an SVI, packets received on the ports to which a port ACL is applied are filtered by the port ACL. Outgoing routed IPv6 packets are filtered by the router ACL. Other packets are not filtered.

Note If any port ACL (IPv4, IPv6, or MAC) is applied to an interface, that port ACL is used to filter packets, and any router ACLs attached to the SVI of the port VLAN are ignored.

Related Topics

Creating IPv6 ACL, on page 1481
Applying an IPv6 to an Interface, on page 1485
Creating WLAN IPv6 ACL, on page 1487
Displaying IPv6 ACLs, on page 1487

Types of ACL

Per User IPv6 ACL

For the per-user ACL, the full access control entries (ACE) as the text strings are configured on the ACS.

The ACE is not configured on the Controller. The ACE is sent to the switch in the `ACCESS-Accept` attribute and applies it directly for the client. When a wireless client roams into an foreign switch, the ACEs are sent to the foreign switch as an AAA attribute in the mobility Handoff message.

Filter ID IPv6 ACL

For the filter-Id ACL, the full ACEs and the `acl name(filter-id)` is configured on the switch and only the `filter-id` is configured on the ACS. The `filter-id` is sent to the switch in the ACCESS-Accept attribute, and the switch looks up the filter-id for the ACEs, and then applies the ACEs to the client. When the client L2 roams to the foreign switch, only the filter-id is sent to the foreign switch in the mobility Handoff message. The foreign switch has to configure the filter-id and ACEs beforehand.

Downloadable IPv6 ACL

For the downloadable ACL(dACL), the full ACEs and the `dacl` name are all configured on the ACS only.

Note The controller does not configure any ACL.

The ACS sends the `dacl` name to the switch in its `ACCESS-Accept` attribute, which takes the `dacl` name and sends the `dACL` name back to the ACS, for the ACEs, using the `access-request` attribute.

The ACS responds to the corresponding ACEs of the switch in the `access-accept` attribute. When the wireless client roams to an foreign switch, only the `dacl` name is sent to the foreign switch in the mobility Handoff message. The foreign switch contacts the ACS server with the `dacl` name to retrieve the ACEs.

IPv6 ACLs and Switch Stacks

The stack master supports IPv6 ACLs in hardware and distributes the IPv6 ACLs to the stack members.

Note For full IPv6 functionality in a switch stack, all stack members must be running the IP services feature set.

If a new switch takes over as stack master, it distributes the ACL configuration to all stack members. The member switches sync up the configuration distributed by the new stack master and flush out entries that member switches sync up the configuration distributed by the new stack master and flush out entries that are not required.

When an ACL is modified, attached to, or detached from an interface, the stack master distributes the change to all stack members.

Configuring IPv6 ACLs

To filter IPv6 traffic, you perform these steps:

Before You Begin

Before configuring IPv6 ACLs, you must select one of the dual IPv4 and IPv6 SDM templates.

SUMMARY STEPS

1. Create an IPv6 ACL, and enter IPv6 access list configuration mode.
2. Configure the IPv6 ACL to block (deny) or pass (permit) traffic.
3. Apply the IPv6 ACL to the interface where the traffic needs to be filtered.
4. Apply the IPv6 ACL to an interface. For router ACLs, you must also configure an IPv6 address on the Layer 3 interface to which the ACL is applied.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Create an IPv6 ACL, and enter IPv6 access list configuration mode.	
Step 2	Configure the IPv6 ACL to block (deny) or pass (permit) traffic.	
Step 3	Apply the IPv6 ACL to the interface where the traffic needs to be filtered.	
Step 4	Apply the IPv6 ACL to an interface. For router ACLs, you must also configure an IPv6 address on the Layer 3 interface to which the ACL is applied.	

Default IPv6 ACL Configuration

There are no IPv6 ACLs configured or applied.

Interaction with Other Features and Switches

- If an IPv6 router ACL is configured to deny a packet, the packet is not routed. A copy of the packet is sent to the Internet Control Message Protocol (ICMP) queue to generate an ICMP unreachable message for the frame.
- If a bridged frame is to be dropped due to a port ACL, the frame is not bridged.
- You can create both IPv4 and IPv6 ACLs on a switch or switch stack, and you can apply both IPv4 and IPv6 ACLs to the same interface. Each ACL must have a unique name; an error message appears if you try to use a name that is already configured.

  You use different commands to create IPv4 and IPv6 ACLs and to attach IPv4 or IPv6 ACLs to the same Layer 2 or Layer 3 interface. If you use the wrong command to attach an ACL (for example, an IPv4 command to attach an IPv6 ACL), you receive an error message.
- You cannot use MAC ACLs to filter IPv6 frames. MAC ACLs can only filter non-IP frames.
- If the hardware memory is full, for any additional configured ACLs, packets are dropped to the CPU, and the ACLs are applied in software. When the hardware is full a message is printed to the console indicating the ACL has been unloaded and the packets will be dropped on the interface.

  Note Only packets of the same type as the ACL that could not be added (ipv4, ipv6, MAC) will be dropped on the interface.

How To Configure an IPv6 ACL

Creating IPv6 ACL

Beginning in privileged EXEC mode, follow these steps to create an IPv6 ACL:

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 access-list** *acl_name*
3. **{deny|permit} protocol**
4. **{deny|permit} tcp**
5. **{deny|permit} udp**
6. **{deny|permit} icmp**
7. **end**
8. **show ipv6 access-list**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ipv6 access-list** *acl_name* **Example:** `ipv6 access-list access-list-name`	Use a name to define an IPv6 access list and enter IPv6 access-list configuration mode.
Step 3	**{deny|permit} protocol** **Example:** `{deny | permit} protocol {source-ipv6-prefix/prefix-length | any | host source-ipv6-address} [operator [port-number]]{destination-ipv6-prefix/prefix-length | any |host destination-ipv6-address} [operator [port-number]][dscp value] [fragments][log] [log-input] [routing][sequence value] [time-range name]`	Enter deny or permit to specify whether to deny or permit the packet if conditions are matched. These are the conditions: • For protocol, enter the name or number of an Internet protocol: ahp, esp, icmp, ipv6, pcp, stcp, tcp, or udp, or an integer in the range 0 to 255 representing an IPv6 protocol number. • The source-ipv6-prefix/prefix-length or destination-ipv6-prefix/ prefix-length is the source or destination IPv6 network or class of networks for which to set deny or permit conditions, specified in hexadecimal and using 16-bit values between colons (see RFC 2373). • Enter any as an abbreviation for the IPv6 prefix ::/0. • For host source-ipv6-address or destination-ipv6-address, enter the source or destination IPv6 host address for which to set deny or permit conditions, specified in hexadecimal using 16-bit values between colons. • (Optional) For operator, specify an operand that compares the source or destination ports of the specified protocol. Operands are lt (less than), gt (greater than), eq (equal), neq (not equal), and range.

	Command or Action	Purpose
		If the operator follows the source-ipv6-prefix/prefix-length argument, it must match the source port. If the operator follows the destination-ipv6- prefix/prefix-length argument, it must match the destination port. • (Optional) The port-number is a decimal number from 0 to 65535 or the name of a TCP or UDP port. You can use TCP port names only when filtering TCP. You can use UDP port names only when filtering UDP. • (Optional) Enter dscp value to match a differentiated services code point value against the traffic class value in the Traffic Class field of each IPv6 packet header. The acceptable range is from 0 to 63. • (Optional) Enter fragments to check noninitial fragments. This keyword is visible only if the protocol is ipv6. • (Optional) Enter log to cause an logging message to be sent to the console about the packet that matches the entry. Enter log-input to include the input interface in the log entry. Logging is supported only for router ACLs. • (Optional) Enter routing to specify that IPv6 packets be routed. • (Optional) Enter sequence value to specify the sequence number for the access list statement. The acceptable range is from 1 to 4294967295 • (Optional) Enter time-range name to specify the time range that applies to the deny or permit statement.
Step 4	**{deny|permit} tcp** **Example:** `{deny | permit} tcp {source-ipv6-prefix/prefix-length | any | hostsource-ipv6-address} [operator [port-number]]{destination-ipv6-prefix/prefix-length | any |hostdestination-ipv6-address} [operator [port-number]][ack] [dscp value][established] [fin] [log][log-input] [neq {port |protocol}] [psh] [range{port | protocol}] [rst][routing] [sequence value] [syn] [time-range name][urg]`	(Optional) Define a TCP access list and the access conditions. Enter tcp for Transmission Control Protocol. The parameters are the same as those described in Step 3, with these additional optional parameters: • ack—Acknowledgment bit set. • established—An established connection. A match occurs if the TCP datagram has the ACK or RST bits set. • fin—Finished bit set; no more data from sender. • neq {port | protocol}—Matches only packets that are not on a given port number. • psh—Push function bit set. • range {port | protocol}—Matches only packets in the port number range. • rst—Reset bit set.

	Command or Action	Purpose
		• syn—Synchronize bit set. • urg—Urgent pointer bit set.

Step 5	**{deny|permit} udp**
	
	Example:
	`{deny | permit} udp {source-ipv6-prefix/prefix-length | any | hostsource-ipv6-address} [operator [port-number]]{destination-ipv6-prefix/prefix-length | any | hostdestination-ipv6-address} [operator [port-number]][dscp value] [log][log-input]`
	
	`[neq {port |protocol}] [range {port |protocol}] [routing][sequence value][time-range name]`
		(Optional) Define a UDP access list and the access conditions.
		
		Enter udp for the User Datagram Protocol. The UDP parameters are the same as those described for TCP, except that the operator [port]] port number or name must be a UDP port number or name, and the established parameter is not valid for UDP.

Step 6	**{deny|permit} icmp** **Example:** `{deny | permit} icmp {source-ipv6-prefix/prefix-length | any | hostsource-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length | any | hostdestination-ipv6-address} [operator [port-number]][icmp-type [icmp-code] |icmp-message] [dscpvalue] [log] [log-input] [routing] [sequence value][time-range name]`	(Optional) Define an ICMP access list and the access conditions. Enter icmp for Internet Control Message Protocol. The ICMP parameters are the same as those described for most IP protocols in Step 3a, with the addition of the ICMP message type and code parameters. These optional keywords have these meanings: • icmp-type—Enter to filter by ICMP message type, a number from 0 to 255. • icmp-code—Enter to filter ICMP packets that are filtered by the ICMP message code type, a number from 0 to 255. • icmp-message—Enter to filter ICMP packets by the ICMP message type name or the ICMP message type and code name. To see a list of ICMP message type names and code names, use the ? key or see command reference for this release.
Step 7	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.
Step 8	**show ipv6 access-list** **Example:** `show ipv6 access-list`	Verify the access list configuration.
Step 9	**copy running-config startup-config** **Example:** `copy running-config startup-config`	(Optional) Save your entries in the configuration file.

Related Topics

Prerequisites for IPv6 ACL, on page 1477
Understanding IPv6 ACLs, on page 1478
Applying an IPv6 to an Interface, on page 1485
Creating WLAN IPv6 ACL, on page 1487
Displaying IPv6 ACLs, on page 1487

Applying an IPv6 to an Interface

This section describes how to apply IPv6 ACLs to network interfaces. You can apply an IPv6 ACL to outbound or inbound traffic on layer 2 and Layer 3 interfaces. You can apply IPv6 ACLs only to inbound management traffic on Layer 3 interfaces.

Beginning in privileged EXEC mode, follow these steps to control access to an interface:

SUMMARY STEPS

1. **configure terminal**
2. **interface** interface_id
3. **no switchport**
4. **ipv6 address** *ipv6_address*
5. **ipv6 traffic-filter** *acl_name*
6. **end**
7. **show running-config interface tenGigabitEthernet 1/0/3**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** interface_id **Example:** `Switch# interface interface-id`	Identifies a Layer 2 interface (for port ACLs) or Layer 3 Switch Virtual interface (for router ACLs) on which to apply an access list, and enters interface configuration mode.
Step 3	**no switchport** **Example:** `Switch# no switchport`	Changes the interface from Layer 2 mode (the default) to Layer 3 mode (only if applying a router ACL).

	Command or Action	Purpose
Step 4	**ipv6 address** *ipv6_address* **Example:** `Switch# ipv6 address ipv6-address`	Configures an IPv6 address on a Layer 3 interface (for router ACLs). **Note** This command is not required on Layer 2 interfaces or if the interface has already been configured with an explicit IPv6 address.
Step 5	**ipv6 traffic-filter** *acl_name* **Example:** `Switch# ipv6 traffic-filter access-list-name {in | out}`	Applies the access list to incoming or outgoing traffic on the interface.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Step 7	**show running-config interface tenGigabitEthernet 1/0/3**
	
	Example:
	`Switch# show running-config interface tenGigabitEthernet 1/0/3`
	`..........................`
	`..........................`
	`Building configuration ...........`
	`..........................`
	`Current configuration : 98 bytes`
	`!`
	`interface TenGigabitEthernet1/0/3`
	` switchport mode trunk`
	` ipv6 traffic-filter MyFilter out`
	`end`
		Shows the configuration summary.

Step 8	**copy running-config startup-config** **Example:** `copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Creating IPv6 ACL, on page 1481
Understanding IPv6 ACLs, on page 1478
Creating WLAN IPv6 ACL, on page 1487
Displaying IPv6 ACLs, on page 1487

Creating WLAN IPv6 ACL

SUMMARY STEPS

1. **ipv6 traffic-filter acl** *acl_name*
2. **ipv6 traffic-filter acl web**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**ipv6 traffic-filter acl** *acl_name* **Example:** `Switch(config-wlan)# ipv6 traffic-filter acl <acl_name>`	Creates a named WLAN ACL.
Step 2	**ipv6 traffic-filter acl web** **Example:** `Switch(config-wlan)# ipv6 traffic-filter acl web <acl_name-preauth>`	Creates a pre-authentication for WLAN ACL.

```
Switch(config-wlan)# ipv6 traffic-filter acl <acl_name>
Switch(config-wlan)#ipv6 traffic-filter acl web <acl_name-preauth>
```

Related Topics

Creating IPv6 ACL, on page 1481
Applying an IPv6 to an Interface, on page 1485
Understanding IPv6 ACLs, on page 1478
Displaying IPv6 ACLs, on page 1487

Verifying IPv6 ACL

Displaying IPv6 ACLs

You can display information about all configured access lists, all IPv6 access lists, or a specific access list by using one or more of the privileged EXEC commands.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show access-list** **Example:** `Switch# show access-lists`	Displays all access lists configured on the switch
Step 2	**show ipv6 access-list** *acl_name* **Example:** `Switch# show ipv6 access-list [access-list-name]`	Displays all configured IPv6 access list or the access list specified by name.

Related Topics

Creating IPv6 ACL, on page 1481
Applying an IPv6 to an Interface, on page 1485
Creating WLAN IPv6 ACL, on page 1487
Understanding IPv6 ACLs, on page 1478

Configuration Examples for IPv6 ACL

Example: Creating IPv6 ACL

This example configures the IPv6 access list named CISCO. The first deny entry in the list denies all packets that have a destination TCP port number greater than 5000. The second deny entry denies packets that have a source UDP port number less than 5000. The second deny also logs all matches to the console. The first permit entry in the list permits all ICMP packets. The second permit entry in the list permits all other traffic. The second permit entry is necessary because an implicit deny -all condition is at the end of each IPv6 access list.

Note Logging is supported only on Layer 3 interfaces.

```
Switch(config)# ipv6 access-list CISCO
Switch(config-ipv6-acl)# deny tcp any any gt 5000
Switch (config-ipv6-acl)# deny ::/0 lt 5000 ::/0 log
Switch(config-ipv6-acl)# permit icmp any any
Switch(config-ipv6-acl)# permit any any
```

Example: Applying IPv6 ACLs

This example shows how to apply the access list Cisco to outbound traffic on a Layer 3 interface.

```
Switch(config-if)# no switchport
Switch(config-if)# ipv6 address 2001::/64 eui-64
Switch(config-if)# ipv6 traffic-filter CISCO out
```

Example: Displaying IPv6 ACLs

This is an example of the output from the **show access-lists** privileged EXEC command. The output shows all access lists that are configured on the switch or switch stack.

```
Switch #show access-lists
Extended IP access list hello
10 permit ip any any
IPv6 access list ipv6
permit ipv6 any any sequence 10
```

This is an example of the output from the show ipv6 access-lists privileged EXEC command. The output shows only IPv6 access lists configured on the switch or switch stack.

```
Switch# show ipv6 access-list
IPv6 access list inbound
permit tcp any any eq bgp (8 matches) sequence 10
permit tcp any any eq telnet (15 matches) sequence 20
permit udp any any sequence 30

IPv6 access list outbound
deny udp any any sequence 10
deny tcp any any eq telnet sequence 20
```

Example: Configuring RA Throttling and NS Suppression

This task describes how to create an RA throttle policy in order to help the power-saving wireless clients from being disturbed by frequent unsolicited periodic RA's. The unsolicited multicast RA is throttled by the controller.

Before You Begin

Enable IPv6 on the client machine.

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 nd ra-throttler policy** Mythrottle
3. **throttle-period** 20
4. **max-through** 5
5. **allow at-least** 3 **at-most** 5
6. **switch (config)# vlan configuration 100**
7. **ipv6 nd suppress**
8. **ipv6 nd ra-th attach-policy** attach-policy_name
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**ipv6 nd ra-throttler policy** Mythrottle **Example:** `Switch (config)# ipv6 nd ra-throttler policy Mythrottle`	Creates a RA throttler policy called Mythrottle.
Step 3	**throttle-period** 20 **Example:** `Switch (config-nd-ra-throttle)# throttle-period 20`	Determines the time interval segment during which throttling applies.
Step 4	**max-through** 5 **Example:** `Switch (config-nd-ra-throttle)# max-through 5`	Determines how many initial RA's are allowed.
Step 5	**allow at-least** 3 **at-most** 5 **Example:** `Switch (config-nd-ra-throttle)# allow at-least 3 at-most 5`	Determines how many RA's are allowed after the initial RAs have been transmitted, until the end of the interval segment.
Step 6	**switch (config)# vlan configuration 100** **Example:** `Switch (config)# vlan configuration 100`	Creates a per vlan configuration.
Step 7	**ipv6 nd suppress** **Example:** `Switch (config)# ipv6 nd suppress`	Disables the neighbor discovery on the Vlan.
Step 8	**ipv6 nd ra-th attach-policy** attach-policy_name **Example:** `Switch (config)# ipv6 nd ra-throttle attach-policy attach-policy_name`	Enables the router advertisement throttling.
Step 9	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Example: Configuring RA Guard Policy

SUMMARY STEPS

1. **ipv6 nd raguard policy** MyPloicy
2. **trusted-port**
3. **device-role router**
4. **interface tenGigabitEthernet** 1/0/1
5. **ipv6 nd raguard attach-policy**MyPolicy
6. **vlan configuration** 19-21,23
7. **ipv6 nd suppress**
8. **ipv6 snooping**
9. **ipv6 nd raguard attach-policy** MyPolicy
10. **ipv6 nd ra-throttler attach-policy** Mythrottle

DETAILED STEPS

	Command or Action	Purpose
Step 1	**ipv6 nd raguard policy** MyPloicy **Example:** `Switch (config)# ipv6 nd raguard policy MyPolicy`	
Step 2	**trusted-port** **Example:** `Switch (config-nd-raguard)# trusted-port`	Configures the trusted port for the policy created above.

Step 3	**device-role router**
	Example:
	`Switch (config-nd-raguard)# device-role [host|monitor|router|switch]`
	`Switch (config-nd-raguard)# device-role router`
		Defines the trusted device that can send RAs to the trusted port created above.

Step 4	**interface tenGigabitEthernet** 1/0/1 **Example:** `Switch (config)# interface tenGigabitEthernet 1/0/1`	Configures the interface to the trusted device.
Step 5	**ipv6 nd raguard attach-policy**MyPolicy **Example:** `Switch (config-if)# ipv6 nd raguard attach-policy Mypolicy`	Configures and attaches the policy to trust the RA's received from the port.
Step 6	**vlan configuration** 19-21,23 **Example:** `Switch (config)# vlan configuration 19-21,23`	Configures the wireless client vlans.

	Command or Action	Purpose
Step 7	**ipv6 nd suppress** **Example:** `Switch (config-vlan-config)# ipv6 nd suppress`	Suppresses the ND messages over wireless.
Step 8	**ipv6 snooping** **Example:** `Switch (config-vlan-config)# ipv6 snooping`	Captures IPv6 traffic.
Step 9	**ipv6 nd raguard attach-policy** MyPolicy **Example:** `Switch (config-vlan-config)# ipv6 nd raguard attach-policy Mypolicy`	Attaches the RA Guard policy to the wireless client vlans.
Step 10	**ipv6 nd ra-throttler attach-policy** Mythrottle **Example:** `Switch (config-vlan-config)#ipv6 nd ra-throttler attach-policy Mythrottle`	Attaches the RA throttling policy to the wireless client vlans.

Example: Configuring IPv6 Neighbor Binding

SUMMARY STEPS

1. **ipv6 neighbor binding** [vlan]19 2001:db8::25:4 **interface tenGigabitEthernet** 1/0/3 aaa.bbb.ccc

DETAILED STEPS

	Command or Action	Purpose
Step 1	**ipv6 neighbor binding** [vlan]19 2001:db8::25:4 **interface tenGigabitEthernet** 1/0/3 aaa.bbb.ccc **Example:** `Switch (config)# ipv6 neighbor binding vlan 19 2001:db8::25:4 interface tenGigabitEthernet 1/0/3 aaa.bbb.ccc`	Sets and validates the neighbor 2001:db8::25: 4 only valid when transmitting on VLAN 19 through interface te1/0/3 with the source mac-address as aaa.bbb.ccc.

Additional References

Related Documents

Related Topic	Document Title
IPv6 command reference	*IPv6 Command Reference (Catalyst 3850 Switches)*
ACL configuration	*Security Configuration Guide (Catalyst 3850 Switches)*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for IPv6 ACLs

This table lists the features in this module and provides links to specific configuration information:

Feature	Release	Modification
IPv6 ACL Functionality	Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 69

Configuring IPv6 Web Authentication

Prerequisites for IPv6 Web Authentication

The following configurations must be in place before you start with IPv6 Web Authentication:

- IPv6 Device Tracking.
- IPv6 DHCP Snooping.
- Disable security of type 802.1x on the wlan.
- Each WLAN must have a vlan associated to it.
- Change the default wlan setting from **shutdown** to **no shutdown**.

Related Topics

Enabling Security on the WLAN, on page 1498

Restrictions for IPv6 Web Authentication

The following restrictions are implied when using IPv6 web authentication:

Related Topics

Enabling Security on the WLAN, on page 1498

Information About IPv6 Web Authentication

Web authentication is a Layer 3 security feature and the switch disallows IP traffic (except DHCP and DNS -related packets) from a particular client until it supplies a valid username and password. It is a simple authentication method without the need for a supplicant or client utility. Web authentication is typically used by customers who deploy a guest-access network. Traffic from both, HTTP and HTTPS, page is allowed to display the login page.

Note Web authentication does not provide data encryption and is typically used as simple guest access for either a hot spot or campus atmosphere, where connectivity is always a factor.

A WLAN is configured as **security webauth** for web based authentication. The switch supports the following types of web based authentication:

- Web Authentication – The client enters the credentials in a web page which is then validated by the Wlan controller.
- Web Consent – The Wlan controller presents a policy page with Accept/Deny buttons. Click Accept button to access the network.

A Wlan is typically configured for open authentication, that is without Layer 2 authentication, when web-based authentication mechanism is used.

Web Authentication Process

The following events occur when a WLAN is configured for web authentication:

- The user opens a web browser and enters a URL address, for example, *http://www.example.com*. The client sends out a DNS request for this URL to get the IP address for the destination. The switch bypasses the DNS request to the DNS server, which in turn responds with a DNS reply that contains the IP address of the destination *www.example.com*. This, in turn, is forwarded to the wireless clients.
- The client then tries to open a TCP connection with the destination IP address. It sends out a TCP SYN packet destined to the IP address of *www.example.com*.
- The switch has rules configured for the client and cannot act as a proxy for *www.example.com*. It sends back a TCP SYN-ACK packet to the client with source as the IP address of *www.example.com*. The client sends back a TCP ACK packet in order to complete the three-way TCP handshake and the TCP connection is fully established.
- The client sends an HTTP GET packet destined to *www.example.com*. The switch intercepts this packet and sends it for redirection handling. The HTTP application gateway prepares an HTML body and sends it back as the reply to the HTTP GET requested by the client. This HTML makes the client go to the default web-page of the switch, for example, *http://<Virtual-Server-IP>/login.html*.
- The client closes the TCP connection with the IP address, for example, *www.example.com*.
- If the client wants to go to virtual IP, the client tries to open a TCP connection with the virtual IP address of the switch. It sends a TCP SYN packet for virtual IP to the switch.
- The switch responds back with a TCP SYN-ACK and the client sends back a TCP ACK to the switch in order to complete the handshake.

- The client sends an HTTP GET for */login.html* destined to virtual IP in order to request for the login page.
- This request is allowed to the web server of the switch, and the server responds with the default login page. The client receives the login page in the browser window where the user can log in.

Related Topics

Disabling WPA, on page 1497
Enabling Security on the WLAN, on page 1498
Enabling a Parameter Map on the WLAN, on page 1499
Enabling Authentication List on WLAN, on page 1499
Configuring a Global WebAuth WLAN Parameter Map, on page 1500
Configuring the WLAN, on page 1501
Enabling IPv6 in Global Configuration Mode, on page 1502
Verifying the Parameter Map, on page 1503
Verifying Authentication List, on page 1503

How to Configure IPv6 Web Authentication

Disabling WPA

Before You Begin

Disable 802.1x. A typical web authentication does not use Layer 2 security. Use this configuration to remove Layer 2 security.

SUMMARY STEPS

1. **configure terminal**
2. **wlan** test1 2 test1
3. **no security** wpa

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**wlan** test1 2 test1 **Example:** `Switch(config)# wlan test1 2 test1`	Creates a WLAN and assign an SSID to it.

	Command or Action	Purpose
Step 3	**no security** wpa **Example:** `Switch(config-wlan)# no security wpa`	Disables the WPA support for Wlan.

What to Do Next

Enable the following:

- Security Web Authentication.
- Parameter Local.
- Authentication List.

Related Topics

Web Authentication Process, on page 1496

Enabling Security on the WLAN

SUMMARY STEPS

1. **parameter-map type web-auth global**
2. **virtual-ip ipv4** 192.0.2.1
3. **virtual-ip ipv6** 2001:db8::24:2

DETAILED STEPS

	Command or Action	Purpose
Step 1	**parameter-map type web-auth global** **Example:** `Switch(config)# parameter-map type web-auth global`	Applies the parameter map to all the web-auth wlans.
Step 2	**virtual-ip ipv4** 192.0.2.1 **Example:** `Switch(config-params-parameter-map)# virtual-ip ipv4 192.0.2.1`	Defines the virtual gateway IPv4 address.
Step 3	**virtual-ip ipv6** 2001:db8::24:2 **Example:** `Switch(config-params-parameter-map)# virtual-ip ipv6 2001:db8::24:2`	Defines the virtual gateway IPv6 address.

Related Topics

Prerequisites for IPv6 Web Authentication, on page 1495
Restrictions for IPv6 Web Authentication, on page 1495
Web Authentication Process, on page 1496

Enabling a Parameter Map on the WLAN

SUMMARY STEPS

1. **security web-auth parameter-map** <mapname>

DETAILED STEPS

	Command or Action	Purpose
Step 1	**security web-auth parameter-map** <mapname> **Example:** `Switch(config-wlan)# security web-auth parameter-map webparalocal`	Enables web authentication for the wlan and creates a parameter map.

Related Topics

Web Authentication Process, on page 1496

Enabling Authentication List on WLAN

SUMMARY STEPS

1. **security web-auth authentication-list** webauthlistlocal

DETAILED STEPS

	Command or Action	Purpose
Step 1	**security web-auth authentication-list** webauthlistlocal **Example:** `Switch(config-wlan)# security web-auth`	Enables web authentication for the wlan and creates a local web authentication list.

Related Topics

Web Authentication Process, on page 1496

Configuring a Global WebAuth WLAN Parameter Map

Use this example to configure a global web auth WLAN and add a parameter map to it.

SUMMARY STEPS

1. **parameter-map type webauth global**
2. **virtual-ip ipv6** 2001:db8:4::1
3. **ratelimit init-state-sessions** 120
4. **max-https-conns** 70

DETAILED STEPS

	Command or Action	Purpose
Step 1	**parameter-map type webauth global** **Example:** `Switch (config)# parameter-map type webauth global`	Configures a global webauth and adds a parameter map to it.
Step 2	**virtual-ip ipv6** 2001:db8:4::1 **Example:** `Switch (config-params-parameter-map)# virtual-ip ipv6 2001:db8:4::1`	Defines a virtual gateway IP address that appears to the wireless clients for authentication.
Step 3	**ratelimit init-state-sessions** 120 **Example:** `Switch (config-params-parameter-map)# ratelimit init-state-sessions 120`	Sets the global ratelimit to limit the bandwidth that the web clients can use on the switch to avoid over-flooding attacks.
Step 4	**max-https-conns** 70 **Example:** `Switch (config-params-parameter-map)# max-http-conns 70`	Sets the maximum number of attempted http connections on the switch to avoid over-flooding atatcks.

Related Topics

Web Authentication Process, on page 1496
Configuring the WLAN, on page 1501

Configuring the WLAN

Before You Begin

- The WLAN must have a Vlan associated with it. By default, a new Wlan is always associated with Vlan 1, which can be changed as per the configuration requirements.
- Configure and enable the WLAN to *no shutdown*. By default, the Wlan is configured with the *shutdown* parameter and is disabled.

SUMMARY STEPS

1. **wlan** *1*
2. **client vlan** *interface ID*
3. **security web-auth authentication list** webauthlistlocal
4. **security web-auth parameter-map** global
5. **no security wpa**
6. **no shutdown**
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**wlan** *1* **Example:** `Switch(config-wlan)# wlan 1 name vicweb ssid vicweb`	Creates a wlan and assign an SSID to it.
Step 2	**client vlan** *interface ID* **Example:** `Switch(config-wlan)# client vlan VLAN0136`	Assigns the client to vlan interface.
Step 3	**security web-auth authentication list** webauthlistlocal **Example:** `Switch(config-wlan)# security web-auth authentication-list webauthlistlocal`	Configures web authentication for the wlan.
Step 4	**security web-auth parameter-map** global **Example:** `Switch(config-wlan)# security web-auth parameter-map global`	Configures the parameter map on the wlan.
Step 5	**no security wpa** **Example:** `Switch(config-wlan)# no security wpa`	Configures the security policy for a wlan. This enables the wlan.

	Command or Action	Purpose
Step 6	**no shutdown** **Example:** `Switch(config-wlan)# no shutdown`	Configures and enables the Wlan.
Step 7	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Related Topics

Configuring a Global WebAuth WLAN Parameter Map, on page 1500

Web Authentication Process, on page 1496

Enabling IPv6 in Global Configuration Mode, on page 1502

Enabling IPv6 in Global Configuration Mode

Enable IPv6 in global configuration for web authentication.

SUMMARY STEPS

1. **configure terminal**
2. **web-auth global**
3. **virtual IPv6**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**web-auth global** **Example:** `Switch(config)# parameter-map type webauth global`	Globally configures the parameter map type as web authentication.
Step 3	**virtual IPv6** **Example:** `Switch(config-params-parameter-map)# virtual-ip ipv6`	Selects IPv6 as the virtaul IP for web authentication. **Note** You can also select IPv4 as the preferred IP for web authentication.

Related Topics

Configuring the WLAN, on page 1501

Web Authentication Process, on page 1496

Verifying the Parameter Map, on page 1503

Verifying IPv6 Web Authentication

Verifying the Parameter Map

Use the **show running configuration** command to verify the parameter map configured for Wlan.

SUMMARY STEPS

1. **show running config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show running config** **Example:** `Switchshow running config`	Displays the entire running configuration for the switch. Grep for parameter map to view the result.

```
wlan alpha 2 alpha
 no security wpa
 no security wpa akm dot1x
 no security wpa wpa2
 no security wpa wpa2 ciphers aes
 security web-auth
 security web-auth authentication-list webauthlistlocal
 security web-auth parameter-map webparalocal
```

Related Topics

Enabling IPv6 in Global Configuration Mode, on page 1502

Web Authentication Process, on page 1496

Verifying Authentication List, on page 1503

Verifying Authentication List

Use the **show running configuration** command to verify the authentication list configured for the Wlan.

SUMMARY STEPS

1. **show running configuration**
2. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show running configuration** **Example:** `Switch#show running-config`	Displays the Wlan configuration. `Switch# show running-config`
Step 2	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

```
Switch#show running-config
 ..................................
 ..................................
 ..................................
 wlan alpha 2 alpha
 no security wpa
 no security wpa akm dot1x
 no security wpa wpa2
 no security wpa wpa2 ciphers aes
 security web-auth
 security web-auth authentication-list webauthlistlocal
 security web-auth parameter-map webparalocal
 ..................................
 ..................................
 ..................................
```

Related Topics

Verifying the Parameter Map, on page 1503
Web Authentication Process, on page 1496

Additional References

Related Documents

Related Topic	Document Title
IPv6 command reference	*IPv6 Command Reference (Catalyst 3850 Switches)*
Web Authentication configuration	*Security Configuration Guide (Catalyst 3850 Switches)*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for IPv6 Web Authentication

This table lists the features in this module and provides links to specific configuration information:

Feature	Release	Modification
IPv6 Web Authentication Functionality	Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 70

Configuring IPv6 Client Mobility

Prerequisites for IPv6 Client Mobility

To enable wireless IPv6 client connectivity, the underlying wired network must support IPv6 routing and an address assignment mechanism such as SLAAC or DHCPv6. The switch must have L2 adjacency to the IPv6 router, and the VLAN needs to be tagged when the packets enter the switch. APs do not require connectivity on an IPv6 network, as all traffic is encapsulated inside the IPv4 CAPWAP tunnel between the AP and switch.

Restrictions For IPv6 Client Mobility

- When using the IPv6 Client Mobility, clients must support IPv6 with either static stateless auto configuration (such as Windows XP clients) or stateful DHCPv6 IP addressing (such as Windows 7 clients).
- To allow smooth operation of stateful DHCPv6 IP addressing, you must have a switch or router that supports the DHCP for IPv6 feature (such as the switch) that is configured to act like a DHCPv6 server, or you need a dedicated server such as a Windows 2008 server with a built-in DHCPv6 server. Cisco Catalyst 3850 switch and Cisco Catalyst 5700 switch can act as (internal) a DHCPv6 server.

Note To load the SDM IPv6 template in the Cisco Catalyst 3850 switch, enter the **sdm prefer dual-ipv4 and v6** default command and then reset the switch.

Information About IPv6 Client Mobility

The Switch supports IPv6 mobility for IPv6-only or dual-stack nodes. The IPv6 Client Mobility is divided into:

- Link Layer and
- Network Layer

The link layer is handled by the 802.11 protocol which enables the client to roam to any AP in the same BSS (basic service set) identified by the same SSID without losing the link layer connectivity.

However, link layer mobility is not enough to make wireless client Layer 3 applications continue to work seamlessly while roaming. Cisco IOSd's wireless mobility module uses mobility tunneling to retain seamless connectivity for the client's Layer 3 PoP (point of presence) when the client roams across different subnets on different switches.

IPv6 is the next-generation network layer Internet protocol intended to replace IPv4 in the TCP/IP suite of protocols. This new version increases the internet global address space to accommodate users and applications that require unique global IP addresses. IPv6 incorporates 128-bit source and destination addresses, which provide significantly more addresses than the 32-bit IPv4 addresses.

To support IPv6 clients across controllers, ICMPv6 messages must be dealt with specially to ensure the IPv6 client remains on the same Layer 3 network. The switch keep track of IPv6 clients by intercepting the ICMPv6 messages to provide seamless mobility and protect the network from network attacks. The NDP (neighbor discovery packets) packets are converted from multicast to unicast and delivered individually per client. This unique solution ensures that Neighbor Discovery and Router Advertisement packets are not leaked across Vlans. Clients can receive specific Neighbor Discovery and Router Advertisement packets ensuring correct IPv6 addressing and avoids unnecessary multicast traffic.

The configuration for IPv6 mobility is the same as IPv4 mobility and requires no separate software on the client side to achieve seamless roaming. The switch must be part of the same mobility group. Both IPv4 and IPv6 client mobility are enabled by default.

IPv6 client mobility is used for the following:

- Retaining the client IPv6 multiple addresses in Layer-2 and Layer-3 roaming.
- IPv6 Neighbor Discovery Prootcol (NDP) packet management.
- Client IPv6 addresses learning.

Using Router Advertisment

The Neighbor Discovery Protocol(NDP) operates in the link-layer and is responsible for the discovery of other nodes on the link. It determines the link-layer addresses of other nodes, finds the available routers, and maintains reachability information about the paths to other active neighbor nodes.

Router Advertisement (RA) is one of the IPv6 Neighbor Discovery Protocol (NDP) packets that is used by the hosts to discover available routers, acquire the network prefix to generate the IPv6 addresses, link MTU, and so on. The routers send RA on a regular basis, or in response to hosts Router Solicitation messages.

IPv6 wireless client mobility manages the IPv6 RA packet . The converged access switch forwards the link-local all-nodes multicast RA packets to the local and roaming wireless nodes mapped on same VLAN the RA was received on.

Figure 1 illustrates the link-local all-nodes mcast RA forwarding issue in the wireless node mobility.

Figure 70: Roaming Client Receiving Invalid RA from Router 2

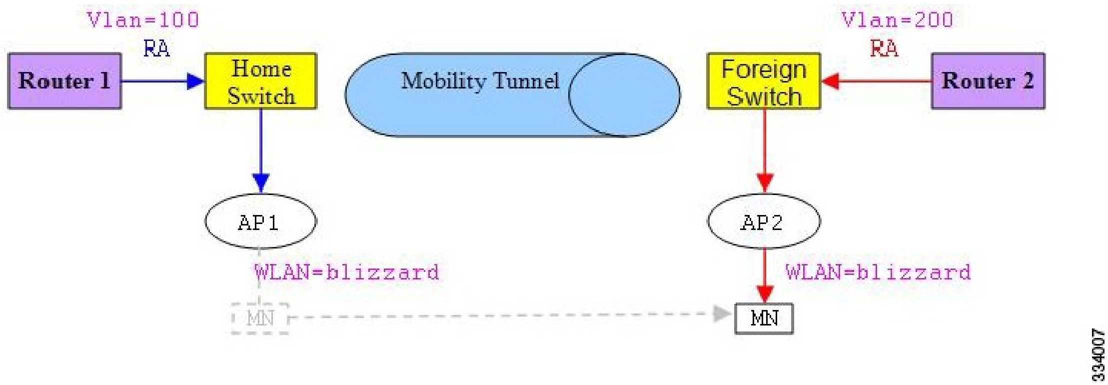
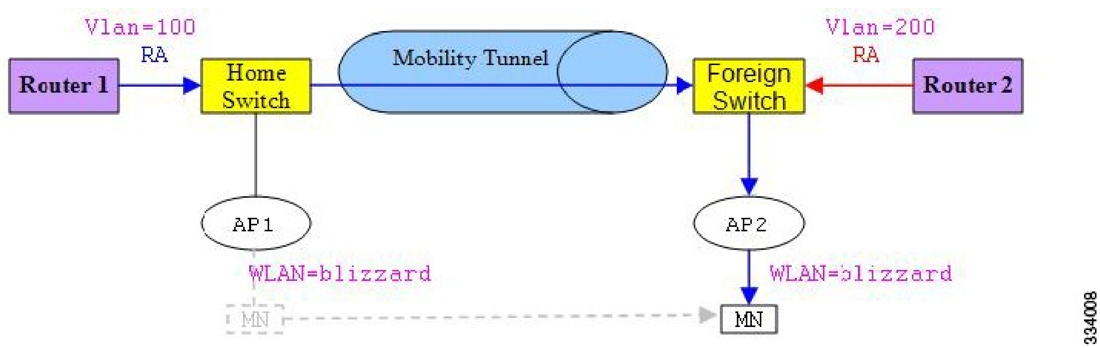


Figure 2 illustrates how a roaming client "MN" receives RA from VLAN 200 in a foreign switch and how it acquires an new IP address and breaks into L3 mobility's point of presence.

Figure 71: Roaming Client Receives Valid RA from Router 1



Related Topics

Verifying IPv6 Client Mobility, on page 1511
Monitoring IPv6 Client Mobility, on page 1512

RA Throttling and NS suppression

To safeguard the power-saving wireless clients form being disturbed by frequent unsolicited periodic RAs, the controller can throttle the unsolicited multicast RA.

Related Topics

Verifying IPv6 Client Mobility, on page 1511
Monitoring IPv6 Client Mobility, on page 1512

IPv6 Address Learning

There are three ways for IPv6 client to acquire IPv6 addresses:

- Stateless Address Auto-Configuration (SLAAC)
- Stateful DHCPv6
- Static configuration

For these methods, the IPv6 client always sends NS DAD (duplicate address detection) to ensure that there is no duplicated IP address on the network. The switch snoops the clients NDP and DHCPv6 packets to learn about its client IP addresses and then updates the controllers database. The database then informs the controller for the clients new IP address.

Related Topics

Verifying IPv6 Client Mobility, on page 1511
Monitoring IPv6 Client Mobility, on page 1512

Handling Multiple IP Addresses

In the case when the new IP address is received after RUN state, whether an addition or removal, the controller updates the new IP addresses on its local database for display purposes. Essentially, the IPv6 uses the existing or same PEM state machine code flow as in IPv4. When the IP addresses are requested by external entities, for example, from Prime Infrastructure, the controller will include all the available IP addresses, IPv4 and IPv6, in the API/SPI interface to the external entities.

An IPv6 client can acquire multiple IP addresses from stack for different purposes. For example, a link-local address for link local traffic, and a routable unique local or global address.

When the client is in the DHCP request state and the controller receives the first IP address notification from the database for either an IPv4 or IPv6 address, the PEM moves the client into the RUN state.

When a new IP address is received after the RUN state, either for addition or removal, the controller updates the new IP addresses on its local database for display purposes.

When the IP addresses are requested by external entities, for example, from Prime Infrastructure, the controller provides the available IP addresses, both IPv4 and IPv6, to the external entities.

Related Topics

Verifying IPv6 Client Mobility, on page 1511
Monitoring IPv6 Client Mobility, on page 1512

IPv6 Configuration

The switch supports IPv6 client as seamlessly as the IPv4 clients. The administrator must manually configure the Vlans to enable the IPV6, IPv6's snooping and throttling functionality. This will enable the NDP packets to throttle between the switch and its various clients

Related Topics

Verifying IPv6 Client Mobility, on page 1511
Monitoring IPv6 Client Mobility, on page 1512

High Availability

The switch will sync with the wireless clients when the clients IP address is hard to learn. When a switchover happens, the IPv6 neighbor binding table is synced to standby state. However, the wireless client will itself disassociate and reassociate to a new active state once the switchover is complete and the neighbor binding table is updated with latest information for that client.

If, during the reassociation, the client moves to another AP then the original entry in the binding table is marked as down for sometime and will be aged-out.

For the new entries joining the switch from another AP, the new IP address is learned and notified to the controller's database.

Note This feature is available only for the Cisco Catalyst 3850 Switch.

Related Topics

Verifying IPv6 Client Mobility, on page 1511
Monitoring IPv6 Client Mobility, on page 1512

Verifying IPv6 Client Mobility

The commands listed in the Table 1 applies to the IPv6 client mobility.

Table 156: Commands for Verifying IPv6 Client Mobility on Cisco 5760 WLC

Command	Description
debug mobility ipv6	Enables all the wireless client IPv6 mobility debugs.
debug **client** **mac-address** (mac-addr)	Displays wireless client debugging. Enter a MAC address for debugging information.

Related Topics

Using Router Advertisment, on page 1508
RA Throttling and NS suppression, on page 1509
IPv6 Address Learning, on page 1510
Handling Multiple IP Addresses, on page 1510
IPv6 Configuration, on page 1510
High Availability, on page 1511
Monitoring IPv6 Client Mobility, on page 1512

Monitoring IPv6 Client Mobility

The commands in Table 2 are used to monitor IPv6 Client mobility on the switch.

Table 157: Monitoring IPv6 Client Mobility Commands

Commands	**Description**
show wireless client summary	Displays the wireless specific configuration of active clients.
show wireless client mac-address (mac-addr)	Displays the wireless specific configuration of active clients based on their MAC address.

Related Topics

Verifying IPv6 Client Mobility, on page 1511
Using Router Advertisment, on page 1508
RA Throttling and NS suppression, on page 1509
IPv6 Address Learning, on page 1510
Handling Multiple IP Addresses, on page 1510
IPv6 Configuration, on page 1510
High Availability, on page 1511

Additional References

Related Documents

Related Topic	**Document Title**
IPv6 command reference	*IPv6 Command Reference (Catalyst 3850 Switches)*
Mobility configuration	*Mobility Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*

Error Message Decoder

Description	**Link**
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information For IPv6 Client Mobility

This table lists the features in this module and provides links to specific configuration information:

Feature	Release	Modification
IPv6 Client Mobility Functionality	Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 71

Configuring IPv6 Mobility

Pre-requisites for IPv6 Mobility

The mobility and its related infrastructure must be configured and ready for use.

Information About IPv6 Mobility

Mobility, or roaming, is a wireless LAN client's ability to maintain its association seamlessly from one access point to another securely and with as little latency as possible. This section explains how mobility works whenswitch are included in a wireless network.

When a wireless client associates and authenticates to an access point, the access point's switch places an entry for that client in its client database. This entry includes the client's MAC and IP addresses, security context and associations, quality of service (QoS) contexts, the WLAN, and the associated access point. The switch uses this information to forward frames and manage traffic to and from the wireless client.

When the wireless client moves its association from one access point to another, the switch simply updates the client database with the newly associated access point. If necessary, new security context and associations are established as well. The process becomes more complicated, however, when a client roams from an access point joined to one switch to an access point joined to a different switch. It also varies based on whether theswitch are operating on the same subnet.

Inter Controller Roaming

When the client associates to an access point joined to a new switch, the new switch exchanges mobility messages with the original switch, and the client database entry is moved to the new switch if sticky anchoring is disabled.

Related Topics

Monitoring IPv6 Mobility, on page 1516

Intra Subnet Roaming with Sticky Anchoring, and Inter Subnet Roaming

Inter-subnet roaming is similar to inter-controller roaming in that the switch exchange mobility messages on the client roam. However, instead of moving the client database entry to the new switch, the original switch marks the client with an "Anchor" entry in its own client database. The database entry is copied to the new switch client database and marked with a "Foreign" entry in the new switch. The roam remains transparent to the wireless client, and the client maintains its original IP address.

In inter-subnet roaming, WLANs on both anchor and foreign switch need to have the same network access privileges and no source-based routing or source-based firewalls in place. Otherwise, the clients may have network connectivity issues after the handoff.

For more information on configuring mobility see, the Cisco 5700 Wireless LAN Controller Mobility Configuration Guide, Cisco IOS XE, Release 3.2SE.

Related Topics

Monitoring IPv6 Mobility, on page 1516

How to Configure IPv6 Mobility

Monitoring IPv6 Mobility

This chapter displays the mobility related IPv6 configuration. To see the mobility related configurations refer to the Cisco 5700 Wireless LAN Controller Mobility Configuration Guide, Cisco IOS XE 3.2SE.

SUMMARY STEPS

1. **show ipv6 neighbors binding mac** C0C1.C06B.C4E2

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show ipv6 neighbors binding mac** C0C1.C06B.C4E2 **Example:** `Switch# show ipv6 neighbors binding mac C0C1.C06B.C4E2`	Displays the IPv6 related mobility configurations.

```
Switch# show ipv6 neighbors binding mac C0C1.C06B.C4E2
Binding Table has 45 entries, 37 dynamic (limit 100)
Codes: L - Local, S - Static, ND - Neighbor Discovery, DH - DHCP, PKT - Other Packet, API
- API created
Preflevel flags (prlvl):
0001:MAC and LLA match     0002:Orig trunk            0004:Orig access
0008:Orig trusted trunk    0010:Orig trusted access   0020:DHCP assigned
0040:Cga authenticated     0080:Cert authenticated    0100:Statically assigned


    IPv6 address                            Link-Layer addr Interface vlan prlvl  age
state    Time left
L   FE80:20:25::16                          2037.064C.BA71  Vl25        25  0100 3137mn
REACHABLE
L   FE80:20:24::16                          2037.064C.BA41  Vl24        24  0100 3137mn
REACHABLE
L   FE80:20:23::16                          2037.064C.BA44  Vl23        23  0100 3137mn
REACHABLE
ND  FE80:20:23::13                          2037.0653.6BC4  Te1/0/1     23  0005   85s
REACHABLE  223 s try 0
ND  FE80:20:22::17                          2037.064D.06F6  Te1/0/1     22  0005    3mn
REACHABLE  92 s try 0
L   FE80:20:22::16                          2037.064C.BA76  Vl22        22  0100 3137mn
REACHABLE
ND  FE80:20:22::13                          2037.0653.6BF6  Te1/0/1     22  0005  165s
REACHABLE  136 s try 0
ND  FE80:20:22::12                          2037.064C.94F6  Te1/0/1     22  0005   23s
REACHABLE  281 s try 0
ND  FE80:20:22::2                           0022.550E.8FC3  Te1/0/1     22  0005   18s
REACHABLE  295 s try 0
ND  FE80:20:21::17                          2037.064D.06E8  Te1/0/1     21  0005    4mn
REACHABLE  60 s try 0
L   FE80:20:21::16                          2037.064C.BA68  Vl21        21  0100 3137mn
REACHABLE
ND  FE80:20:21::13                          2037.0653.6BE8  Te1/0/1     21  0005   57s
REACHABLE  252 s try 0
ND  FE80:20:21::12                          2037.064C.94E8  Te1/0/1     21  0005    4s
REACHABLE  297 s
ND  FE80:20:21::2                           0022.550E.8FC2  Te1/0/1     21  0005    2s
REACHABLE  307 s try 0
ND  FE80::F866:8BE0:12E4:39CF               C0C1.C06B.C4E2  Ca4         21  0005    3mn
REACHABLE  89 s try 0
ND  FE80::6D0A:DB33:D69E:91C7               0050.B606.A6CE  Te1/0/1     22  0005  135s
REACHABLE  171 s try 0
ND  FE80::985:8189:9937:BB05                8CA9.8295.09CC  Ca0         21  0005   15s
REACHABLE  287 s
ND  FE80::20:24:13                          2037.0653.6BC1  Te1/0/1     24  0005  155s
REACHABLE  145 s try 0
L   2001:20:23::16                          2037.064C.BA44  Vl23        23  0100 3137mn
REACHABLE
DH  2001:20:22:0:C96C:AF29:5DDC:2689        0050.B606.A6CE  Te1/0/1     22  0024   19s
REACHABLE  286 s try 0(16574
DH  2001:20:22:0:A46B:90B2:F0DB:F952        0050.B606.A6CE  Te1/0/1     22  0024 2339mn
STALE      32401 s
DH  2001:20:22:0:7DFD:14EC:B1E4:1172        0050.B606.A6CE  Te1/0/1     22  0024 2339mn
STALE      24394 s
DH  2001:20:22:0:7CB3:D6DD:FD6A:50F         0050.B606.A6CE  Te1/0/1     22  0024 2333mn
STALE      29195 s
DH  2001:20:22:0:6D32:AF24:FDE1:2504        0050.B606.A6CE  Te1/0/1     22  0024  509mn
STALE      118821 s
DH  2001:20:22:0:5106:5AD:FE98:A2F0         0050.B606.A6CE  Te1/0/1     22  0024 2328mn
STALE      31362 s
ND  2001:20:22::201:13                      0050.B606.A6CE  Te1/0/1     22  0005   49s
REACHABLE  264 s try 0
L   2001:20:22::16                          2037.064C.BA76  Vl22        22  0100 3137mn
REACHABLE
ND  2001:20:22::13                          2037.0653.6BF6  Te1/0/1     22  0005  175s
REACHABLE  131 s try 0
ND  2001:20:22::2                           0022.550E.8FC3  Te1/0/1     22  0005   28s
```

```
REACHABLE  274 s try 0
ND  2001:20:21:0:F866:8BE0:12E4:39CF        C0C1.C06B.C4E2  Ca4        21  0005    4mn
REACHABLE  21 s try 0
ND  2001:20:21:0:C085:9D4C:4521:B777        0021.CC73.AA17  Te1/0/1    21  0005   11s
REACHABLE  290 s try 0
ND  2001:20:21:0:6233:4BFF:FE1A:744C        6033.4B1A.744C  Ca4        21  0005    3mn
REACHABLE  108 s try 0
ND  2001:20:21:0:447E:745D:2F48:1C68        8CA9.8295.09CC  Ca0        21  0005   34s
REACHABLE  276 s
ND  2001:20:21:0:3920:DDE8:B29:AD51         C0C1.C06B.C4E2  Ca4        21  0005    3mn
REACHABLE  87 s try 0
ND  2001:20:21:0:1016:A333:FAD5:6E66        0021.CC73.AA17  Te1/0/1    21  0005    4mn
REACHABLE  18 s try 0
ND  2001:20:21:0:C42:E317:BA9B:EB17         6033.4B1A.744C  Ca4        21  0005    4mn
REACHABLE  61 s try 0
ND  2001:20:21:0:985:8189:9937:BB05         8CA9.8295.09CC  Ca0        21  0005  135s
REACHABLE  173 s try 0
ND  2001:20:21::201:20                      0021.CC73.AA17  Te1/0/1    21  0005    4mn
REACHABLE  43 s try 0
ND  2001:20:21::17                          2037.064D.06E8  Te1/0/1    21  0005    4mn
REACHABLE  50 s try 0
L   2001:20:21::16                          2037.064C.BA68  Vl21       21  0100 3137mn
REACHABLE
ND  2001:20:21::13                          2037.0653.6BE8  Te1/0/1    21  0005   67s
REACHABLE  237 s try 0
ND  2001:20:21::12                          2037.064C.94E8  Te1/0/1    21  0005    5mn
REACHABLE  512 ms try 0
ND  2001:20:21::2                           0022.550E.8FC2  Te1/0/1    21  0005   12s
REACHABLE  294 s try 0
```

Related Topics

Inter Controller Roaming, on page 1516
Intra Subnet Roaming with Sticky Anchoring, and Inter Subnet Roaming, on page 1516

Additional References

Related Documents

Related Topic	Document Title
IPv6 command reference	*IPv6 Command Reference (Catalyst 3850 Switches)*
Mobility configurations	*Mobility Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for IPv6 Mobility

This table lists the features in this module and provides links to specific configuration information:

Feature	Release	Modification
IPv6 Mobility Functionality	Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 72

Configuring IPv6 NetFlow

Prerequisites For IPv6 Netflow

The networking device must be running a Cisco IOSd release that supports Cisco IOS Flexible NetFlow.

IPv6 Traffic

- One of the following must be enabled on your router and on any interfaces on which you want to enable Flexible NetFlow:
    - Cisco Express Forwarding IPv6 or
    - Distributed Cisco Express Forwarding IPv6.

Restrictions For IPv6 Netflow

The following restrictions apply to IPv6 Netflow configurations:

- Locally generated traffic (traffic that is generated by the router, Cisco WLC 5760, on which the Flexible NetFlow Output Accounting feature is configured) is not counted as flow traffic for the Output Flexible NetFlow Accounting feature.

- The Flexible NetFlow Output Accounting feature counts CEF-switched packets only. Process switched transit packets are not counted.

Information About IPv6 Netflow

NetFlow is a monitoring feature used on customer applications for network monitoring, user monitoring and profiling, network planning, security analysis, billing and accounting, and data warehousing and mining. You can use Flexible NetFlow on uplink ports to monitor user-defined flows, collect flow statistics, and perform per-flow policing. It collects and exports flow statistics to a collector device.

Note Flexible NetFlow is supported only on the Catalyst 3750-X and 3560-X switch running the IP base or IP services feature set and equipped with the network services module. It is not supported on switches running the NPE or the LAN base image.

Note Not all of the Flexible NetFlow commands in the command reference are available on the switch. Unsupported commands are either not visible or generate an error message if entered.

Understanding Flexible Netflow

With Flexible NetFlow, traffic is processed and packets are classified into flows. New flows are inserted in the NetFlow table, and statistics are automatically updated. You must configure both ingress and egress NetFlow monitoring. The network services module supports one monitor per interface per direction.

Flexible NetFlow consists of the following components:

- Records— These are combinations of key and non-key fields assigned to monitor Flexible NetFlow monitors to define the cache used to store data.
- Flow monitors— These are applied to interfaces to perform network traffic monitoring. A flow monitor includes a user-defined record, an optional flow exporter, and a cache that is automatically created when the monitor is applied to the first interface. The switch supports normal caches that age out according to settings.
- Flow exporters— These export the data in the flow monitor cache to a remote system, such as a server running NetFlow collector.
- Flow samplers— These reduce the load that Flexible NetFlow puts on the networking device to monitor traffic by limiting the number of packets that are analyzed.

You can configure unidirectional flow (destination or source-address based flows), and flow aging. The following features are supported on the network services module:

- Configuring collection statistics for Layer 2-switched (non-routing) traffic, Layer 3 (CAPWAP) IPv4 and IPv6 traffic, and Layer 4 TCP, IGMP, and ICMP traffic.
- NetFlow counting, maintenance, troubleshooting (debugging commands).

- NetFlow analysis is performed on traffic crossing the physical interfaces on the network services module. The switch processes egress (outbound) traffic after forwarding decisions are performed. Locally switched or routed traffic is forced through service module ports by configuring private VLANs or protected ports.

The following NetFlow characteristics are not supported:

- Netflow-5 protocol
- Predefined flow records
- ISL
- Policy-based NetFlow
- Cisco TrustSec monitoring

Though other modules that can be installed in the switch have 1-Gigabit and 10-Gigabit uplink interfaces, NetFlow is supported only on the network services module.

IPv6 Netflow

Flexible Netflow (FNF) allows the user to define a flow record (a particular set of key, non-key, counter and time-stamp fields of interest) that is optimal for a particular application by selecting the fields from a big collection of pre-defined fields, using CLI configuration commands.

The collection of the pre-defined fields includes the following fields:

- Data-link layer (L2) header fields
- IPv6 header fields
- Transport layer (L4) header fields
- Application layer (L5) header fields
- Routing attributes (generic, IPv4, IPv6)
- Interface fields
- Counter fields
- Timestamp fields

Related Topics

Configuring a Customized Flow Record , on page 1524
Configuring the Flow Exporters , on page 1526
Configuring a Customized Flow Monitor, on page 1530
Applying a Flow Monitor to an Interface, on page 1532
Configuring and Enabling Flow Sampling , on page 1534

How To Configure IPv6 Netflow

Configuring a Customized Flow Record

You can match the following fields for the flow record:

- IPv4 or IPv6 destination address
- Datalink fields, to identify Layer 2 source and destination address and VLAN for traffic entering or leaving the interfaces, providing the MAC address of the directly connected host. Class of Service (CoS) and Ethertype datalink header fields are also available.
- Transport field source and destination ports, to identify the type of application: ICMP, IGMP, or TCP traffic.

You can collect the following fields for the flow record:

- The total number of bytes, flows or packets sent by the exporter (exporter) or the number of bytes or packets in a 64-bit counter (long). The timestamp based on system uptime from the time the first packet was sent or from the time the most recent (last) packet was seen.
- The SNMP index of the input or output interface. The interface for traffic entering or leaving the service module is based on the switch forwarding cache. This field is typically used in conjunction with datalink, IPv4, and IPv6 addresses, and provides the actual first-hop interface for directly connected hosts.
    - A value of 0 means that interface information is not available in the cache.
    - Some NetFlow collectors require this information in the flow record.

The following steps configure the customized flow record:

SUMMARY STEPS

1. **configure terminal**
2. **flow record** recordname
3. **description** description
4. **match**{ipv4 | ipv6}{destination | hop-limit | protocol | source | traffic-class| version} **address**
5. **match datalink** [dot1q | ethertype | mac | vlan]
6. **match transport** [destination-port | icmp | source-port]
7. **match interface** [input |output]
8. **match flow direction**
9. **collect counter** {bytes [layer2 | long] | packets [long]}
10. **collect timestamp absolute** [first | last]
11. **collect interface** [input | output]
12. **collect transport tcp flags** {ack | cwr | ece | fin | psh | rst | syn | urg}
13. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**flow record** recordname **Example:** `Switch(config)# flow record TestRecordName`	Creates a flow record and enters Flexible NetFlow flow record configuration mode. This command can also modify an existing flow record.
Step 3	**description** description **Example:** `Switch(config-flow-record)# description SampleNetflowDescription`	(Optional) Creates a description for the flow record.
Step 4	**match**{ipv4 | ipv6}{destination | hop-limit | protocol | source | traffic-class| version} **address** **Example:** `Switch(config-flow-record)# match ipv6 destination address`	Configures key ipv4 and ipv6 fields for the flow record.
Step 5	**match datalink** [dot1q | ethertype | mac | vlan] **Example:** `Switch(config-flow-record)# match datalink [dot1q | ethertype | mac | vlan]`	Configures key datalink (layer 2) fields for the flow record.
Step 6	**match transport** [destination-port | icmp | source-port] **Example:** `Switch(config-flow-record)# match transport [destination-port | icmp | source-port]`	Configures key transport layer fields for the flow record.
Step 7	**match interface** [input |output] **Example:** `Switch(config-flow-record)# match interface input`	Configures key interface fields for the flow record.
Step 8	**match flow direction** **Example:** `Switch(config-flow-record)# match flow direction`	Configures key flow identity fields for the flow record.
Step 9	**collect counter** {bytes [layer2 | long] | packets [long]} **Example:** `Switch(config-flow-record)#collect counter bytes layer2 long`	Configures the counter key field for the flow record.

	Command or Action	Purpose
Step 10	**collect timestamp absolute** [first | last] **Example:** `Switch(config-flow-record)# collect timestamp absolute [first	last]`	Configures the timestamp key field for the flow record.
Step 11	**collect interface** [input | output] **Example:** `Switch(config-flow-record)# collect interface [input	output]`	Configures the interface key field for the flow record.
Step 12	**collect transport tcp flags** {ack | cwr | ece | fin | psh | rst | syn | urg} **Example:** `Switch(config-flow-record)# collect transport tcp flags ack`	Configures transports tcp flag fields for the flow record.
Step 13	**end** **Example:** `Switch(config)#` **`end`**	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

```
Switch(config)# flow record
Switch(config-flow-record)# description record to monitor network traffic
Switch(config-flow-record)# match ipv6 destination address
Switch(config-flow-record)# match datalink [dot1q | ethertype | mac | vlan]
Switch(config-flow-record)# match transport [destination-port | icmp |igmp | source-port]
Switch(config-flow-record)# match interface input
Switch(config-flow-record)# match flow direction
Switch(config-flow-record)#collect counter bytes layer2 long
Switch(config-flow-record)# collect timestamp absolute first
Switch(config-flow-record)# collect interface [input | output]
Switch(config-flow-record)# collect transport tcp flags ack
Switch(config-flow-record)# end
```

Related Topics

IPv6 Netflow, on page 1523
Configuring the Flow Exporters , on page 1526
Configuring a Customized Flow Monitor, on page 1530
Applying a Flow Monitor to an Interface, on page 1532
Configuring and Enabling Flow Sampling , on page 1534

Configuring the Flow Exporters

The following steps are used to configure the NetFlow exporter.

Note The optional export-protocol flow exporter configuration command specifies the NetFlow export protocol used by the exporter. The switch supports only netflow-v9. Though visible in the CLI help, netflow-5 is not supported.

SUMMARY STEPS

1. **configure terminal**
2. **flow exporter** exporter-name
3. **description** description
4. **destination** {hostname | ip-address} **vrf** vrf-name
5. **dscp** <0-63>
6. **source** interface-id
7. **option** {exporter-stats | interface-table | sampler-table} **timeout** seconds]
8. **export-protocolnetflow-v9**
9. **template data timeout** seconds
10. **transport udp** udp-port
11. **ttl** seconds
12. **end**

DETAILED STEPS

	Command or Action
Step 1	**configure terminal** **Example:** `Switch# configure terminal`
Step 2	**flow exporter** exporter-name **Example:** `Switch(config)# flow exporter TestNetFlowExporterName`
Step 3	**description** description **Example:** `Switch(config-flow-exporter)# description SampleNetFlowExporterDescription`

Command or Action

Step 4 **destination** {hostname | ip-address} **vrf** vrf-name

Example:
```
Switch(config-flow-exporter)# destination 198.51.100.120 vrf SampleVrfName
```

Step 5 **dscp** <0-63>

Example:
```
Switch(config-flow-exporter)# dscp 23
```

Step 6 **source** interface-id

Example:
```
Switch(config-flow-exporter)# source {
Auto-Template|Capwap|GigabitEthernet|GroupVI|InternalInterface|Loopback|Null|Port-channel|TenGigabitEthernet|Tunne
```

Step 7 **option** {exporter-stats | interface-table | sampler-table} **timeout** seconds]

Example:
```
Switch(config-flow-exporter)# option exporter-stats timeout 600
```

	Command or Action
Step 8	**export-protocolnetflow-v9** **Example:** `Switch(config-flow-exporter)# export-protocol netflow-v9`

Step 9	**template data timeout** seconds
	
	Example:
	`Switch(config-flow-exporter)# template data timeout 600`
	`Switch(config-flow-exporter)#`

Step 10	**transport udp** udp-port **Example:** `Switch(config-flow-exporter)# transport udp 67`
Step 11	**ttl** seconds **Example:** `Switch(config-flow-exporter)# ttl 100`
Step 12	**end** **Example:** `Switch(config)# end`

	Command or Action

```
Switch(config)# flow exporter QoS-Collector
Switch(config-flow-exporter)# description QoS Collector Bldg 19
Switch(config-flow-exporter)# destination 172.20.244.28
Switch(config-flow-exporter)# source vlan 1
Switch(config-flow-exporter)# dscp 3
Switch(config-flow-exporter)# transport udp 2055
Switch(config-flow-exporter)# end
```

What to Do Next

Configuring a Customized Flow Monitor.

Related Topics

Configuring a Customized Flow Record , on page 1524
IPv6 Netflow, on page 1523
Configuring a Customized Flow Monitor, on page 1530
Applying a Flow Monitor to an Interface, on page 1532
Configuring and Enabling Flow Sampling , on page 1534

Configuring a Customized Flow Monitor

The following steps are used to configure a NetFlow monitor.

SUMMARY STEPS

1. **configure terminal**
2. **flow monitor** monitor -name
3. **description** description
4. **record** {TestNetflowRecordName|TestRecord}
5. **cache** {timeout [active| inactive|update] (seconds) | type (normal)}
6. **cache** {timeout [active| inactive|update] (seconds) | type (normal)}
7. **exporter** TestNetFlowExporterName
8. **cache** {timeout [active| inactive|update] (seconds) | type (normal)}
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**flow monitor** monitor -name **Example:** `Switch(config)# flow monitor SampleMonitorName`	Creates a flow monitor and enters Flexible NetFlow flow monitor configuration mode. You can also use this command to modify an existing flow monitor.
Step 3	**description** description **Example:** `Switch(config-flow-monitor)# Description SampleNetFlowMonitorName`	(Optional) Configures a description for the flow monitor.
Step 4	**record** {TestNetflowRecordName|TestRecord} **Example:** `Switch(config-flow-monitor)#record TestNetflowRecordName`	Specifies the record for the flow monitor.
Step 5	**cache** {timeout [active| inactive|update] (seconds) | type (normal)} **Example:** `Switch(config-flow-monitor)# cache type normal`	(Optional) Modifies the flow monitor cache parameters such as timeout values, number of cache entries, and the cache type. • **timeout active** *seconds*—Configures the active flow timeout. This defines the granularity of the traffic analysis. The range is from 1 to 604800 seconds. The default is 1800. Typical values are 60 or 300 seconds. See the Configuring Data Export for Cisco IOS Flexible NetFlow with Flow Exporters document for recommended values. • **type normal**—Configures normal flow removal from the flow cache. **Note** Although visible in the command line help, the entries keyword and inactive and update timeouts are not supported.
Step 6	**cache** {timeout [active| inactive|update] (seconds) | type (normal)} **Example:** `Switch(config-flow-monitor)# cache type normal`	Repeat step 5 to configure additional cache parameters for the flow monitor.
Step 7	**exporter** TestNetFlowExporterName **Example:** `Switch(config-flow-monitor)# exporter TestNetFlowExporterName`	(Optional) Specifies the name of an exporter that was created previously.

	Command or Action	Purpose
Step 8	**cache** {timeout [active| inactive|update] (seconds) | type (normal)} **Example:** `Switch(config-flow-monitor)# cache type normal`	Repeat step 5 to configure additional cache parameters for the flow monitor.
Step 9	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

```
Switch(config)# flow monitor FLOW-MONITOR-1
Switch(config-flow-monitor)# Used for ipv6 traffic analysis
Switch(config-flow-monitor)# record FLOW-RECORD-1
Switch(config-flow-monitor)# cache timeout active 300
Switch(config-flow-monitor)# cache type normal
Switch(config-flow-monitor)# exporter EXPORTER-1
Switch(config-flow-monitor)# exit
```

What to Do Next

Apply a flow monitor to an interface

Related Topics

Configuring a Customized Flow Record , on page 1524
Configuring the Flow Exporters , on page 1526
IPv6 Netflow, on page 1523
Applying a Flow Monitor to an Interface, on page 1532
Configuring and Enabling Flow Sampling , on page 1534

Applying a Flow Monitor to an Interface

The following are used to configure a NetFlow monitor to an interface.

SUMMARY STEPS

1. **configure terminal**
2. **interface** interface-id
3. **wlan** ssid
4. [ip | ipv6 | datalink] **flow monitor** monitor -name **sampler** [sampler | input | output]
5. **exit**
6. Repeat steps 2 and 3
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** interface-id **Example:** `Switch(config)# interface tengigabitEthernet 1/0/1`	Identifies an interface and enters interface configuration mode. Flexible Net Flow is supported only on the service module 1-Gigabit or 10-Gigabit Ethernet interfaces. **Note** You cannot attach a NetFlow monitor to a port channel interface. If both service module interfaces are part of an EtherChannel, you should attach the monitor to both physical interfaces.
Step 3	**wlan** ssid **Example:** `Switch (config)# wlan test 1 test`	Configures the flow monitor on WLAN.
Step 4	[ip | ipv6 | datalink] **flow monitor** monitor -name **sampler** [sampler | input | output] **Example:** `Switch(config-if)# ipv6 flow monitor SampleMonitorName input`	Activates a previously created flow monitor by assigning it to the interface to analyze incoming or outgoing traffic. • ip—Enters record matching IPv4 IP addresses. • ipv6—Enters record matching IPv6 IP addresses. **Note** This keyword is visible only when the dual IPv4 and IPv6 Switch Database Management (SDM) template is configured on the switch. • input—Applies the flow monitor on input traffic. • output—Applies the flow monitor on output traffic. • sampler—(Optional) Applies the flow monitor sampler.

Step 5	**exit**
	
	Example:
	`Switch(config-if)# exit`
	`Switch(config)#`
		Returns to global configuration mode.

Step 6	Repeat steps 2 and 3 **Example:**	Configures additional cache parameters for the flow monitor.
Step 7	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

```
Switch(config)# interface tengigabitethernet 1/0/1
Switch(config-if)# ip flow monitor FLOW-MONITOR-1 input
```

```
Switch(config-if)# ip flow monitor FLOW-MONITOR-2 output
Switch(config-if)# end
```

Related Topics

Configuring a Customized Flow Record , on page 1524
Configuring the Flow Exporters , on page 1526
Configuring a Customized Flow Monitor, on page 1530
IPv6 Netflow, on page 1523
Configuring and Enabling Flow Sampling , on page 1534

Configuring and Enabling Flow Sampling

The following steps are used to configure and enable flow sampling.

SUMMARY STEPS

1. **configure terminal**
2. **sampler** sampler -name
3. **description** description
4. **mode** {deterministic|random} (<1-1>)**out-of** <2-1024>
5. **end**
6. **interface** interface-id
7. **wlan** ssid
8. {ip | ipv6 | datalink] **flow monitor** monitor-name **sampler** sampler-name {input | output}
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**sampler** sampler -name **Example:** `Switch(config)# sampler SampleNameForSAMPLER`	Creates a flow monitor and enters Flexible NetFlow sampler configuration mode. You can also use this command to modify an existing sampler.
Step 3	**description** description **Example:** `Switch(config-sampler)#description SamplerName_1`	(Optional) Configures a description for the sampler.
Step 4	**mode** {deterministic|random} (<1-1>)**out-of** <2-1024>	Specifies the mode and window size from which to select packets. The window size range is from 2 to 1024.

	Command or Action	Purpose
	Example: `Switch(config-sampler)#mode random 1 out-of 2`	**Note** Although visible in the CLI help, the mode deterministic keyword is not supported.
Step 5	**end** **Example:** `Switch(config-sampler)# end`	Returns to global configuration mode.
Step 6	**interface** interface-id **Example:** `Switch(config)# interface tengigabitethernet 1/0/1`	Identifies an interface and enters interface configuration mode.
Step 7	**wlan** ssid **Example:** `Switch(config)# wlan test 1 test`	Configures to apply flow sampler on WLAN.
Step 8	{ip | ipv6 | datalink] **flow monitor** monitor-name **sampler** sampler-name {input | output} **Example:** `Switch(config-if)# ip flow monitor FLOW-MONITOR-1 sampler SAMPLE-1 input`	Activates a previously created IPv4 or IPv6 flow monitor by assigning it to the interface to analyze traffic.
Step 9	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

```
Switch(config)# sampler SAMPLER-1
Switch(config-sampler)# description Sample at 50
Switch(config-sampler)# mode random 1 out-of 2
Switch(config-sampler)# exit
Switch(config)# interface tengigabitethernet 1/0/1
Switch(config)# wlan test 1 test
Switch(config-if)# ip flow monitor FLOW-MONITOR-1 sampler SAMPLE-1 input
```

What to Do Next

How to configure netflow v9 for IPv6.

Related Topics

Configuring a Customized Flow Record , on page 1524
Configuring the Flow Exporters , on page 1526
Configuring a Customized Flow Monitor, on page 1530
Applying a Flow Monitor to an Interface, on page 1532
IPv6 Netflow, on page 1523

Verifying IPv6 Netflow

This section describes the Netflow related **show** commands for IPv6. The following commands can be used to verify Netflow on the switch.

Command	Purpose
show flow record	Displays the status of the flow records.
show flow ssid <ssid_name>	Displays SSID interface information.
show flow monitor {monitor name} {cache|provisioning|statistics}	Displays the flow monitor information.
show flow exporter exporter-name	Displays the status of a flow exporter.
show flow monitor monitor -name	Displays the current status of a flow monitor.
show flow interface interface-id	Verifies that the Flexible NetFlow is configured on the interface.
show flow monitor monitor -name cache format [csv | record | table}	Displays data in the flow monitor cache.
show sampler sampler -name	Displays the current status of a flow sampler.

Monitoring IPv6 Netflow

This section describes the Netflow commands for IPv6.The following commands can be used to monitor Netflow on the switch.

Command	Purpose
show running-config flow record	Displays the configured flow records.
show running-config flow exporter exporter-name	Verifies the configured flow exporter.
show running-config flow monitor monitor -name	Verifies the flow monitor configuration.

Additional References

Related Documents

Related Topic	Document Title
IPv6 command reference	*IPv6 Command Reference (Catalyst 3850 Switches)*

Related Topic	Document Title
Flexible NetFlow command reference	*Cisco Flexible NetFlow Command Reference (Catalyst 3850 Switches)*
Flexible NetFlow configuration	*Cisco Flexible NetFlow Configuration Guide (Catalyst 3850 Switches)*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for IPv6 NetFlow

This table lists the features in this module and provides links to specific configuration information:

Feature	Release	Modification
IPv6 NetFlow Functionality	Cisco IOS XE 3.2SE	This feature was introduced.

PART X

Flexible Netflow

CHAPTER 73

Configuring

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for Flexible NetFlow

The following are prerequisites for your Flexible NetFlow configuration:

- You must configure a source interface. If you do not configure a source interface, the exporter will remain in a disabled state.
- You must configure a valid record name for every flow monitor.

Prerequisites for Wireless Flexible NetFlow

The following are the prerequisites for wireless Flexible NetFlow:

- Ensure that the networking device is running a Cisco release that supports wireless Flexible NetFlow.
- Ensure that the target is connected to a WLAN.
- The networking device must be configured to support protocol types such as IP, IPv6, and datalink.
- Valid flow record and monitor are required before generating the flow.

Restrictions for Flexible NetFlow

The following are restrictions for Flexible NetFlow:

- Traditional NetFlow (TNF) accounting is not supported.
- Flexible NetFlow v5 export format is not supported, only NetFlow v9 export format is supported.
- Both ingress and egress NetFlow accounting is supported.
- Microflow policing feature shares the NetFlow hardware resource with FNF.
- Only one flow monitor per interface and per direction is supported.
- Layer 2, IPv4, and IPv6 traffic types are supported; however, the switch can apply a flow monitor to only one of these types at a time for a given direction and interface.
- Layer 2, VLAN, WLAN and Layer 3 interfaces are supported, but the switch does not support SVI and tunnels.
- The following NetFlow table sizes are supported:

Trim Level	Ingress NetFlow Table	Egress NetFlow Table
LAN Base	Not supported	Not supported
IP Base	8 K	16 K
IP Services	8 K	16 K

- Depending on the switch type, a switch will have one or two forwarding ASICs. The capacities listed in the above table are on a per-ASIC basis.
- The switch can support either one or two ASICs. Each ASIC has 8K ingress and 16 K egress entries.
- The NetFlow tables are on separate compartments and cannot be combined. Depending on which ASIC processed the packet, the flows will be created in the table in the corresponding ASIC.
- Both full flow accounting and sampled NetFlow accounting are supported.
- NetFlow hardware implementation supports four hardware samplers. You can select a sampler rate from 1 out of 2 to 1 out of 1024. Only random sampling mode is supported.

- With the microflow policing feature (which is enabled only for wireless implementation), NetFlow can and should be used only in full flow mode i.e. NetFlow policing cannot be used. For wireless traffic, applying a sampler is not permitted, as it hinders microflow QoS.
- Only full flow accounting is supported for wireless traffic.
- NetFlow hardware uses hash tables internally. Hash collisions can occur in the hardware. Therefore, in spite of the internal overflow Content Addressable Memory (CAM), the actual NetFlow table utilization could be about 80 percent.
- Depending on what fields are used for the flow, a single flow could take two consecutive entries. IPv6 flows also take two entries. In these situations, the effective usage of NetFlow entries is half the table size, which is separate from the above hash collision limitation.
- The switch supports up to 16 flow monitors.
- Microflow policing uses a separate set of flow monitors (limit 3).
- SSID-based NetFlow accounting is supported. SSID is treated in a manner similar to an interface. However, certain fields are not supported (such as AP MAC address and user ID).
- NetFlow v9 format NetFlow export is supported.
- The NetFlow software implementation supports distributed NetFlow export, so the flows are exported from the same switch in which the flow was created.
- Ingress flows are present in the ASIC that first received the packets for the flow. Egress flows are present in the ASIC from which the packets actually left the switch set up.
- The reported value for the bytes count field (called “bytes long”) is Layer-2-packet-size—18 bytes. For classic Ethernet traffic (802.3), this will be accurate. For all other Ethernet types, this field will not be accurate. Use the "bytes layer2” field, which always reports the accurate Layer 2 packet size. For information about supported Flexible NetFlow fields, see Supported Flexible NetFlow Fields, on page 1550.

Information About NetFlow

NetFlow is a Cisco technology that provides statistics on packets flowing through the switch. NetFlow is the standard for acquiring IP operational data from IP networks. NetFlow provides data to enable network and security monitoring, network planning, traffic analysis, and IP accounting. Flexible NetFlow improves on original NetFlow by adding the capability to customize the traffic analysis parameters for your specific requirements. Flexible NetFlow facilitates the creation of more complex configurations for traffic analysis and data export through the use of reusable configuration components.

Overview

uses flows to provide statistics for accounting, network monitoring, and network planning.

A flow is a unidirectional stream of packets that arrives on a source interface and has the same values for the keys. A key is an identified value for a field within the packet. You create a flow using a flow record to define the unique keys for your flow.

The switch supports the feature that enables enhanced network anomalies and security detection. allows you to define an optimal flow record for a particular application by selecting the keys from a large collection of predefined fields.

All key values must match for the packet to count in a given flow. A flow might gather other fields of interest, depending on the export record version that you configure. Flows are stored in the cache.

You can export the data that gathers for your flow by using an exporter and export this data to a remote collector.

You define the size of the data that you want to collect for a flow using a monitor. The monitor combines the flow record and exporter with the cache information.

Wireless Flexible NetFlow Overview

The wireless Flexible NetFlow infrastructure supports the following:

- Flexible NetFlow Version 9.0
- User-based rate limiting
- Microflow policing
- Voice and video flow monitoring
- Reflexive access control list (ACL)

Microflow Policing and User-Based Rate Limiting

Microflow policing associates a 2-color 1-rate policer and related drop statistics to each flow present in the NetFlow table. When the flow mask comprises all packet fields, this functionality is known as microflow policing. When the flow mask comprises either source or destination only, this functionality is known as user-based rate limiting.

Voice and Video Flow Monitoring

Voice and video flows are full flow mask-based entries. The ASIC provides the flexibility to program the policer parameters, share policers across multiple flows and rewrite the IP address and Layer 4 port numbers of these flows.

Note For dynamic entries, the NetFlow engine will use the policer parameters that are derived for the flow based on the policy (ACL/QoS-based policies). Dynamic entries cannot share policer across multiple flows.

Reflexive ACL

Reflexive ACLs allow IP packets to be filtered based on upper-layer session information. The ACLs allow outbound traffic and limit inbound traffic in response to the sessions that originate inside the trusted network. The reflexive ACLs are transparent to the filtering mechanism until a data packet that matches the reflexive entry activates it. At this time, a temporary ACL entry is created and added to the IP-named access lists. The information obtained from the data packet to generate the reflexive ACL entry is permit/deny bit, the source IP address and port, the destination IP address, port, and the protocol type. During reflexive ACL entry evaluation, if the protocol type is either TCP or UDP, then the port information must match exactly. For other protocols, there is no port information to match. After this ACL is installed, the firewall is then opened for the reply packets to pass through. At this time, a potential hacker could have access to the network behind the firewall. To narrow this window, an idle timeout period can be defined. However, in the case of TCP, if two FIN bits or an RST is detected, the ACL entry can be removed.

Related Topics

Configuring WLAN to Apply Flow Monitor in IPV4 and IPv6 Input/Output Direction, on page 1567

Example: Configuring IPv4 Flexible NetFlow in WLAN (Ingress Direction), on page 1570
Example: Configuring IPv6 and Transport Flag Flexible NetFlow in WLAN (Egress Direction), on page 1570
Example: Configuring IPv6 Flexible NetFlow in WLAN (Both Ingress and Egress Directions), on page 1571

Flow Records

A flow record defines the keys that Flexible NetFlow uses to identify packets in the flow, as well as other fields of interest that Flexible NetFlow gathers for the flow. You can define a flow record with any combination of keys and fields of interest. The switch supports a rich set of keys. A flow record also defines the types of counters gathered per flow. You can configure 64-bit packet or byte counters. The switch enables the following match fields as the defaults when you create a flow record:

- match datalink—Layer 2 attributes
- match flow—Flow identifying attributes
- match interface—Interface attributes
- match ipv4—IPv4 attributes
- match ipv6—IPv6 attributes
- match transport—Transport layer fields

Related Topics

Creating a Flow Record, on page 1555

Flexible NetFlow Match Parameters

The following table describes Flexible NetFlow match parameters. You must configure at least one of the following match parameters for the flow records.

Table 158: Match Parameters

Command	Purpose
match datalink {**dot1q** | **ethertype** | **mac** | **vlan** }	Specifies a match to datalink or Layer 2 fields. The following command options are available: • **dot1q**—Matches to the dot1q field. • **ethertype**—Matches to the ethertype of the packet. • **mac**—Matches the source or destination MAC fields. • **vlan**—Matches to the VLAN that the packet is located on (input or output).
match flow direction	Specifies a match to the flow identifying fields.

Command	Purpose
match interface {**input** | **output**}	Specifies a match to the interface fields. The following command options are available: • **input—**Matches to the input interface. • **output—**Matches to the output interface.
match ipv4 {**destination** | **protocol** | **source** | **tos** | **ttl** | **version**}	Specifies a match to the IPv4 fields. The following command options are available: • **destination—**Matches to the IPv4 destination address-based fields. • **protocol—**Matches to the IPv4 protocols. • **source—**Matches to the IPv4 source address based fields. • **tos—**Matches to the IPv4 Type of Service fields. • **ttl—**Matches to the IPv4 Time To Live fields. • **version—**Matches to the IP version from the IPv4 header.
match ipv6 {**destination** | **hop-limit** | **protocol** | **source** | **traffic-class** | **version** }	Specifies a match to the IPv6 fields. The following command options are available: • **destination—**Matches to the IPv6 destination address-based fields. • **hop-limit—**Matches to the IPv6 hop limit fields. • **protocol—**Matches to the IPv6 payload protocol fields. • **source—**Matches to the IPv6 source address based fields. • **traffic-class—**Matches to the IPv6 traffic class. • **version—**Matches to the IP version from the IPv6 header.

Command	Purpose
match transport {**destination-port** | **igmp** | **icmp** | **source-port**}	Specifies a match to the Transport Layer fields. The following command options are available: • **destination-port—**Matches to the transport destination port. • **icmp—**Matches to ICMP fields, including ICMP IPv4 and IPv6 fields. • **igmp—**Matches to IGMP fields. • **source-port—**Matches to the transport source port.

Flexible NetFlow Collect Parameters

The following table describes the Flexible NetFlow collect parameters.

Table 159: Collect Parameters

Command	Purpose
collect counter { **bytes** { **layer2** { **long** } | **long** } | **packets** { **long** } }	Collects the counter fields total bytes and total packets.
collect interface {**input** | **output**}	Collects the fields from the input or output interface.
collect timestamp absolute {**first** | **last**}	Collects the fields for the absolute time the first packet was seen or the absolute time the most recent packet was last seen (in milliseconds).

Command	Purpose
collect transport tcp flags	Collects the following transport TCP flags: • **ack—**TCP acknowledgement flag • **cwr—**TCP congestion window reduced flag • **ece—**TCP ECN echo flag • **fin—**TCP finish flag • **psh—**TCP push flag • **rst—**TCP reset flag • **syn—**TCP synchronize flag • **urg—**TCP urgent flag **Note** On the switch, you cannot specify which TCP flag to collect. You can only specify to collect transport TCP flags. All TCP flags will be collected with this command.

Exporters

An exporter contains network layer and transport layer details for the export packet. The following table lists the configuration options for an exporter.

Table 160: Exporter Configuration Options

Exporter Configuration	Description
default	Sets a command to its default values.
description	Provides a description for the flow exporter.
destination	Export destination.
dscp	Optional DSCP value.
exit	Exits from the flow exporter configuration mode.
export-protocol	Export protocol version.
no	Negates the command or its default.
option	Selects option for exporting.
source	Originating interface for the net flow.
template	Flow exporter template configuration.

Exporter Configuration	Description
transport	Transport protocol.
ttl	Optional TTL or hop limit.

The switch exports data to the collector whenever a timeout occurs or when the flow is terminated (TCP Fin or Rst received, for example). You can configure the following timers to force a flow export:

- Active timeout—The flow continues to have the packets for the past *m* seconds since the flow was created.
- Inactive timeout—The flow does not have any packets for the past *n* seconds.

Related Topics

Creating a Flow Exporter, on page 1557

Export Formats

The switch supports only NetFlow Version 9 export formats. NetFlow Version 9 export format provides the following features and functionality:

- Variable field specification format
- Support for IPv4 destination address export
- More efficient network utilization

Note For information about the Version 9 export format, see RFC 3954.

Monitors

A monitor references the flow record and flow exporter. You apply a monitor to an interface on the switch.

Note the following when applying a flow monitor to an interface:

- If you apply a flow monitor in the input direction:
    - Use the **match** keyword and use the input interface as a key field.
    - Use the **collect** keyword and use the output interface as a collect field. This field will be present in the exported records but with a value of 0.
- If you apply a flow monitor in the output direction:
    - Use the **match** keyword and use the output interface as a key field.
    - Use the **collect** keyword and use the input interface as a collect field. This field will be present in the exported records but with a value of 0.

Related Topics

Creating a Flow Monitor, on page 1559

Samplers

If you are using sampled mode, you use the sampler to specify the rate at which packets are sampled.

Related Topics

Creating a Sampler, on page 1561

Supported Flexible NetFlow Fields

The following tables provide a consolidated list of supported fields in Flexible NetFlow (FNF) for various traffic types and traffic direction.

Note: If the packet has a VLAN field, then that length is not accounted for.

Field	Layer 2 In	Layer 2 Out	IPv4 In	IP v4 Out	IPv6 In	IPv6 Out	Notes
Key or Collect Fields							
Interface input	Yes	—	Yes	—	Yes	—	If you apply a flow monitor in the input direction: • Use the **match** keyword and use the input interface as a key field. • Use the **collect** keyword and use the output interface as a collect field. This field will be present in the exported records but with a value of 0.

Field	Layer 2 In	Layer 2 Out	IPv4 In	IP v4 Out	IPv6 In	IPv6 Out	Notes
Interface output	—	Yes	—	Yes	—	Yes	If you apply a flow monitor in the output direction: • Use the **match** keyword and use the output interface as a key field. • Use the **collect** keyword and use the input interface as a collect field. This field will be present in the exported records but with a value of 0.

Field	Layer 2 In	Layer 2 Out	IPv4 In	IP v4 Out	IPv6 In	IPv6 Out	Notes
Key Fields							
Flow direction	Yes	Yes	Yes	Yes	Yes	Yes	
Ethertype	Yes	Yes	—	—	—	—	
VLAN input	Yes	—	Yes	—	Yes	—	Supported only for a switch port.
VLAN output	—	Yes	—	Yes	—	Yes	Supported only for a switch port.
dot1q VLAN input	Yes	—	Yes	—	Yes	—	Supported only for a switch port.
dot1q VLAN output	—	Yes	—	Yes	—	Yes	Supported only for a switch port.
dot1q priority	Yes	Yes	Yes	Yes	Yes	Yes	Supported only for a switch port.

Field	Layer 2 In	Layer 2 Out	IPv4 In	IP v4 Out	IPv6 In	IPv6 Out	Notes
MAC source address input	Yes	Yes	Yes	Yes	Yes	Yes	
MAC source address output	—	—	—	—	—	—	
MAC destination address input	Yes	—	Yes	—	Yes	—	
MAC destination address output	—	Yes	—	Yes	—	Yes	
IPv4 version	—	—	Yes	Yes	Yes	Yes	
IPv4 TOS	—	—	Yes	Yes	Yes	Yes	
IPv4 protocol	—	—	Yes	Yes	Yes	Yes	Must use if any of src/dest port, ICMP code/type, IGMP type or TCP flags are used.
IPv4 TTL	—	—	Yes	Yes	Yes	Yes	
IPv4 source address	—	—	Yes	Yes	—	—	
IPv4 destination address	—	—	Yes	Yes	—	—	
ICMP IPv4 type	—	—	Yes	Yes	—	—	
ICMP IPv4 code	—	—	Yes	Yes	—	—	

Field	Layer 2 In	Layer 2 Out	IPv4 In	IP v4 Out	IPv6 In	IPv6 Out	Notes
IGMP type	—	—	Yes	Yes	—	—	

Field	Layer 2 In	Layer 2 Out	IPv4 In	IP v4 Out	IPv6 In	IPv6 Out	Notes
Key Fields continued							
IPv6 version	—	—	Yes	Yes	Yes	Yes	Same as IP version.
IPv6 protocol	—	—	Yes	Yes	Yes	Yes	Same as IP protocol. Must use if any of src/dest port, ICMP code/type, IGMP type or TCP flags are used.
IPv6 source address	—	—	—	—	Yes	Yes	
IPv6 destination address	—	—	—	—	Yes	Yes	
IPv6 traffic-class	—	—	Yes	Yes	Yes	Yes	Same as IP TOS.
IPv6 hop-limit	—	—	Yes	Yes	Yes	Yes	Same as IP TTL.
ICMP IPv6 type	—	—	—	—	Yes	Yes	
ICMP IPv6 code	—	—	—	—	Yes	Yes	
source-port	—	—	Yes	Yes	Yes	Yes	
dest-port	—	—	Yes	Yes	Yes	Yes	

Field	Layer 2 In	Layer 2 Out	IPv4 In	IP v4 Out	IPv6 In	IPv6 Out	Notes
Collect Fields							
Bytes long	Yes	Yes	Yes	Yes	Yes	Yes	Packet size = (Ethernet frame size including FCS - 18 bytes) **Recommended:** Avoid this field and use Bytes layer2 long.
Packets long	Yes	Yes	Yes	Yes	Yes	Yes	
Timestamp absolute first	Yes	Yes	Yes	Yes	Yes	Yes	
Timestamp absolute last	Yes	Yes	Yes	Yes	Yes	Yes	
TCP flags	Yes	Yes	Yes	Yes	Yes	Yes	Collects all flags.
Bytes layer2 long	Yes	Yes	Yes	Yes	Yes	Yes	

Default Settings

The following table lists the Flexible NetFlow default settings for the switch.

Table 161: Default Flexible NetFlow Settings

Setting	Default
Flow active timeout	1800 seconds
Flow timeout inactive	15 seconds

How to Configure Flexible NetFlow

To configure Flexible NetFlow, follow these general steps:

1. Create a flow record by specifying keys and non-key fields to the flow.
2. Create an optional flow exporter by specifying the protocol and transport destination port, destination, and other parameters.
3. Create a flow monitor based on the flow record and flow exporter.
4. Create an optional sampler.
5. Apply the flow monitor to a Layer 2 port, Layer 3 port, or VLAN.
6. If applicable to your configuration, configure a WLAN to apply a flow monitor to.

Creating a Flow Record

You can create a flow record and add keys to match on and fields to collect in the flow.

SUMMARY STEPS

1. **configure terminal**
2. **flow record** *name*
3. **description** *string*
4. **match** *type*
5. **collect** *type*
6. **end**
7. **show flow record** [**name** *record-name*]
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**flow record** *name*
	
	Example:
	
	`Switch(config)# flow record test`
	`Switch(config-flow-record)#`
		Creates a flow record and enters flow record configuration mode.

	Command or Action	Purpose
Step 3	**description** *string* **Example:** `Switch(config-flow-record)# description Ipv4Flow`	(Optional) Describes this flow record as a maximum 63-character string.

Step 4	**match** *type*
	
	Example:
	
	`Switch(config-flow-record)# match ipv4 source address`
	`Switch(config-flow-record)# match ipv4 destination address`
	`Switch(config-flow-record)# match flow direction`
		Specifies a match key. For information about possible match key values, see Flexible NetFlow Match Parameters, on page 1545.

Step 5	**collect** *type*
	
	Example:
	
	`Switch(config-flow-record)# collect counter bytes layer2 long`
	`Switch(config-flow-record)# collect counter bytes long`
	`Switch(config-flow-record)# collect timestamp absolute first`
	`Switch(config-flow-record)# collect transport tcp flags`
		Specifies the collection field. For information about possible collection field values, see Flexible NetFlow Collect Parameters, on page 1547.

Step 6	**end** **Example:** `Switch(config-flow-record)#  end`	Returns to privileged EXEC mode.
Step 7	**show flow record** [**name** *record-name*] **Example:** `Switch show flow record test`	(Optional) Displays information about NetFlow flow records.
Step 8	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

What to Do Next

Define an optional flow exporter by specifying the export format, protocol, destination, and other parameters.

Related Topics

Flow Records, on page 1545

Creating a Flow Exporter

You can create a flow export to define the export parameters for a flow.

SUMMARY STEPS

1. **configure terminal**
2. **flow exporter** *name*
3. **description** *string*
4. **dscp** *value*
5. **destination** { *ipv4-address* }
6. **source** { *source type* }
7. **transport udp** *number*
8. **end**
9. **show flow exporter** [**name** *record-name*]
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**flow exporter** *name*
	
	Example:
	
	`Switch(config)# flow exporter ExportTest`
	`Switch (config-flow-exporter)#`
		Creates a flow exporter and enters flow exporter configuration mode.

Step 3	**description** *string* **Example:** `Switch(config-flow-exporter)# description ExportV9`	(Optional) Describes this flow record as a maximum 63-character string.

	Command or Action	Purpose
Step 4	**dscp** *value* **Example:** `Switch(config-flow-exporter)# dscp 0`	(Optional) Specifies the differentiated services codepoint value. The range is from 0 to 63.
Step 5	**destination** { *ipv4-address* } **Example:** `Switch(config-flow-exporter)# destination 192.0.2.1`	Sets the destination IPv4 address or hostname for this exporter.
Step 6	**source** { *source type* } **Example:** `Switch(config-flow-exporter)# source gigabitEthernet1/0/1`	(Optional) Specifies the interface to use to reach the NetFlow collector at the configured destination. The following interfaces can be configured as source: • **Auto Template—**Auto-Template interface • **Capwap—**CAPWAP tunnel interface • **GigabitEthernet—**Gigabit Ethernet IEEE 802 • **GroupVI—**Group virtual interface • **Internal Interface—**Internal interface • **Loopback—**Loopback interface • **Null—**Null interface • **Port-channel—**Ethernet Channel of interface • **TenGigabitEthernet—**10-Gigabit Ethernet • **Tunnel—**Tunnel interface • **Vlan—**Catalyst VLANs
Step 7	**transport udp** *number* **Example:** `Switch(config-flow-exporter)# transport udp 200`	(Optional) Specifies the UDP port to use to reach the NetFlow collector. The range is from 0 to 65535.
Step 8	**end** **Example:** `Switch(config-flow-record)#  end`	Returns to privileged EXEC mode.
Step 9	**show flow exporter** [**name** *record-name*]	(Optional) Displays information about NetFlow flow exporters.

	Command or Action	Purpose
	Example: `Switch show flow exporter ExportTest`	
Step 10	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

What to Do Next

Define a flow monitor based on the flow record and flow exporter.

Related Topics

Exporters, on page 1548

Creating a Flow Monitor

You can create a flow monitor and associate it with a flow record and a flow exporter.

SUMMARY STEPS

1. **configure terminal**
2. **flow monitor** *name*
3. **description** *string*
4. **exporter** *name*
5. **record** *name*
6. **cache** { **timeout** {**active** | **inactive**} *seconds* | **type normal** }
7. **end**
8. **show flow monitor** [**name** *record-name*]
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**flow monitor** *name*
	
	Example:
	
	`Switch(config)# flow monitor MonitorTest`
	`Switch (config-flow-monitor)#`
		Creates a flow monitor and enters flow monitor configuration mode.

Step 3	**description** *string* **Example:** `Switch(config-flow-monitor)# description Ipv4Monitor`	(Optional) Describes this flow record as a maximum 63-character string.
Step 4	**exporter** *name* **Example:** `Switch(config-flow-monitor)# exporter ExportTest`	Associates a flow exporter with this flow monitor.
Step 5	**record** *name* **Example:** `Switch(config-flow-monitor)# record test`	Associates a flow record with the specified flow monitor.
Step 6	**cache** { **timeout** {**active** | **inactive**} *seconds* | **type normal** } **Example:** `Switch(config-flow-monitor)# cache timeout active 15000`	Associates a flow cache with the specified flow monitor.
Step 7	**end** **Example:** `Switch(config-flow-monitor)#  end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 8	**show flow monitor** [**name** *record-name*] **Example:** `Switch` **`show flow monitor name MonitorTest`**	(Optional) Displays information about NetFlow flow monitors.
Step 9	**copy running-config startup-config** **Example:** `Switch#` **`copy running-config startup-config`**	(Optional) Saves your entries in the configuration file.

What to Do Next

Apply the flow monitor to a Layer 2 interface, Layer 3 interface, or VLAN.

Related Topics

Monitors, on page 1549

Creating a Sampler

You can create a sampler to define the NetFlow sampling rate for a flow.

SUMMARY STEPS

1. **configure terminal**
2. **sampler** *name*
3. **description** *string*
4. **mode** {**random**}
5. **end**
6. **show sampler** [*name*]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch#` **`configure terminal`**	Enters the global configuration mode.

	Command or Action	Purpose

Step 2	**sampler** *name*
	
	Example:
	`Switch(config)# sampler SampleTest`
	`Switch(config-flow-sampler)#`
		Creates a sampler and enters flow sampler configuration mode.

Step 3	**description** *string* **Example:** `Switch(config-flow-sampler)# description samples`	(Optional) Describes this flow record as a maximum 63-character string.
Step 4	**mode** {**random**} **Example:** `Switch(config-flow-sampler)# mode random 1 out-of 1024`	Defines the random sample mode.
Step 5	**end** **Example:** `Switch(config-flow-sampler)#  end`	Returns to privileged EXEC mode.
Step 6	**show sampler** [*name*] **Example:** `Switch show sample SampleTest`	(Optional) Displays information about NetFlow samplers.
Step 7	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

What to Do Next

Apply the flow monitor to a source interface, subinterface, VLAN interface, or a VLAN.

Related Topics

Samplers, on page 1550

Applying a Flow to an Interface

You can apply a flow monitor and an optional sampler to an interface.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type*
3. {**ip flow monitor** | **ipv6 flow monitor**}*name* [**sampler** *name*] { **input** |**output** }
4. **end**
5. **show flow interface** [*interface-type number*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**interface** *type*
	
	Example:
	`Switch(config)# interface GigabitEthernet1/0/1`
	`Switch(config-if)#`
		Enters interface configuration mode and configures an interface.
		
		Command parameters for the interface configuration include:
		• **Auto—** Auto-Template interface
		• **Capwap—**CAPWAP tunnel interface
		• **GigabitEthernet—**GigabitEthernet IEEE 802
		• **GroupVI—**Group Virtual interface
		• **Internal Interface—**Internal Interface
		• **Loopback—**Loopback interface
		• **Null—**Null interface
		• **Port-channel—**Ethernet channel of interface
		• **TenGigabitEthernet—**10- Gigabit Ethernet
		• **Tunnel—**Tunnel interface
		• **Vlan—**Catalyst VLANs
		• **Range—**Interface range

Step 3	{**ip flow monitor** | **ipv6 flow monitor**}*name* [**sampler** *name*] { **input** |**output** } **Example:** `Switch(config-if)# ip flow monitor MonitorTest input`	Associate an IPv4 or an IPv6 flow monitor, and an optional sampler to the interface for input or output packets.

	Command or Action	Purpose
Step 4	**end** **Example:** `Switch(config-flow-monitor)#  end`	Returns to privileged EXEC mode.
Step 5	**show flow interface** [*interface-type number*] **Example:** `Switch# show flow interface`	(Optional) Displays information about NetFlow on an interface.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring a Bridged NetFlow on a VLAN

You can apply a flow monitor and an optional sampler to a VLAN.

SUMMARY STEPS

1. **configure terminal**
2. **vlan** [**configuration**] *vlan-id*
3. **ip flow monitor** *name* [**sampler** *name*] {**input** |**output**}
4. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**vlan** [**configuration**] *vlan-id* **Example:** `Switch(config)# vlan configuration 30`	Enters VLAN or VLAN configuration mode.

	Command or Action	Purpose
	`Switch(config-vlan-config)#`	
Step 3	**ip flow monitor** *name* [**sampler** *name*] {**input** |**output**} **Example:** `Switch(config-vlan-config)# ip flow monitor MonitorTest input`	Associates a flow monitor and an optional sampler to the VLAN for input or output packets.
Step 4	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Layer 2 NetFlow

You can define Layer 2 keys in Flexible NetFlow records that you can use to capture flows in Layer 2 interfaces.

SUMMARY STEPS

1. **configure terminal**
2. **flow record** *name*
3. **match datalink** {**dot1q** \|**ethertype** \| **mac** \| **vlan**}
4. **end**
5. **show flow record** [*name*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

Step 2	**flow record** *name*
	
	Example:
	`Switch(config)# flow record L2_record`
	`Switch(config-flow-record)#`
		Enters flow record configuration mode.

	Command or Action	Purpose
Step 3	**match datalink** {**dot1q** |**ethertype** | **mac** | **vlan**} **Example:** `Switch(config-flow-record)# match datalink ethertype`	Specifies the Layer 2 attribute as a key.
Step 4	**end** **Example:** `Switch(config-flow-record)# end`	Returns to privileged EXEC mode.
Step 5	**show flow record** [*name*] **Example:** `Switch# show flow record`	(Optional) Displays information about NetFlow on an interface.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring WLAN to Apply Flow Monitor in Data Link Input/Output Direction

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name*
3. **datalink flow monitor monitor-name {input | output}**
4. **end**
5. **show wlan** *wlan-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**wlan** *wlan-name* **Example:** `Switch (config) # wlan mywlan`	Enters WLAN configuration submode. For *wlan-name*, enter the profile name. The range is 1 to 32 characters.
Step 3	**datalink flow monitor monitor-name {input | output}** **Example:** `Switch (config-wlan) # datalink flow monitor flow-monitor-1 {input | output}`	Applies flow monitor to Layer 2 traffic in the direction of interest.
Step 4	**end** **Example:** `Switch (config) # end`	Returns to privileged EXEC mode.
Step 5	**show wlan** *wlan-name* **Example:** `Switch # show wlan mywlan`	(Optional) Verifies your configuration.

Configuring WLAN to Apply Flow Monitor in IPV4 and IPv6 Input/Output Direction

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-id*
3. **{ip | ipv6} flow monitor monitor-name {input | output}**
4. **end**
5. **show wlan** *wlan-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**wlan** *wlan-id* **Example:** `Switch (config) # wlan 1`	Enters WLAN configuration submode. For *wlan-id*, enter the WLAN ID. The range is 1 to 64.
Step 3	**{ip | ipv6} flow monitor monitor-name {input | output}** **Example:** `Switch (config-wlan) # ip flow monitor flow-monitor-1 input`	Associates a flow monitor to the WLAN for input or output packets.
Step 4	**end** **Example:** `Switch (config) # end`	Returns to privileged EXEC mode.
Step 5	**show wlan** *wlan-name* **Example:** `Switch # show wlan mywlan`	(Optional) Verifies your configuration.

Related Topics

Wireless Flexible NetFlow Overview, on page 1544

Example: Configuring IPv4 Flexible NetFlow in WLAN (Ingress Direction), on page 1570

Example: Configuring IPv6 and Transport Flag Flexible NetFlow in WLAN (Egress Direction), on page 1570

Example: Configuring IPv6 Flexible NetFlow in WLAN (Both Ingress and Egress Directions), on page 1571

Monitoring Flexible NetFlow

The commands in the following table can be used to monitor Flexible NetFlow.

Table 162: Flexible NetFlow Monitoring Commands

Command	Purpose
show flow exporter [**broker** | **export-ids** | **name** | *name* | **statistics** | **templates**]	Displays information about NetFlow flow exporters and statistics.
show flow exporter [**name** *exporter-name*]	Displays information about NetFlow flow exporters and statistics.

Command	Purpose
show flow interface	Displays information about NetFlow interfaces.
show flow monitor [**name** *exporter-name*]	Displays information about NetFlow flow monitors and statistics.
show flow monitor statistics	Displays the statistics for the flow monitor
show flow monitor cache format {**table** | **record** | **csv**}	Displays the contents of the cache for the flow monitor, in the format specified.
show flow record [**name** *record-name*]	Displays information about NetFlow flow records.
show flow ssid	Displays NetFlow monitor installation status for a WLAN.
show sampler [**broker** | **name** | *name*]	Displays information about NetFlow samplers.
show wlan *wlan-name*	Displays the WLAN configured on the device.

Configuration Examples for

Example: Configuring a Flow

This example shows how to create a flow and apply it to an interface:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.

Switch(config)# flow export export1
Switch(config-flow-exporter)# destination 10.0.101.254
Switch(config-flow-exporter)# transport udp 2055
Switch(config-flow-exporter)# exit
Switch(config)# flow record record1
Switch(config-flow-record)# match ipv4 source address
Switch(config-flow-record)# match ipv4 destination address
Switch(config-flow-record)# match ipv4 protocol
Switch(config-flow-record)# match transport source-port
Switch(config-flow-record)# match transport destination-port
Switch(config-flow-record)# collect counter byte long
Switch(config-flow-record)# collect counter packet long
Switch(config-flow-record)# collect timestamp absolute first
Switch(config-flow-record)# collect timestamp absolute last
Switch(config-flow-record)# exit
Switch(config)# flow monitor monitor1
Switch(config-flow-monitor)# record record1
Switch(config-flow-monitor)# exporter export1
Switch(config-flow-monitor)# exit
Switch(config)# interface tenGigabitEthernet 1/0/1
Switch(config-if)# ip flow monitor monitor1 input
Switch(config-if)# end
```

Example: Configuring IPv4 Flexible NetFlow in WLAN (Ingress Direction)

The following example shows how to configure IPv4 Flexible NetFlow on WLAN ingress direction:

```
Switch# configure terminal
Switch(config)# flow record fr_v4
Switch(config-flow-record)# match ipv4 destination address
Switch(config-flow-record)# match ipv4 source address
Switch(config-flow-record)# match ipv4 protocol
Switch(config-flow-record)# match ipv4 tos
Switch(config-flow-record)# match ipv4 ttl
Switch(config-flow-record)# match ipv4 version
Switch(config-flow-record)# match wireless ssid
Switch(config-flow-record)# collect wireless ap mac address
Switch(config-flow-record)# collect counter packets long
Switch(config-flow-record)# collect counter bytes long
Switch(config-flow-record)# collect timestamp absolute first
Switch(config-flow-record)# collect timestamp absolute last
Switch(config-flow-record)# exit

Switch(config)# flow monitor fm_v4
Switch(config-flow-monitor)# record fr_v4
Switch(config-flow-record)# exit

Switch(config)# wlan 1
Switch(config-wlan)# ip flow monitor fm_v4 in
Switch(config-wlan)# end

Switch# show flow monitor fm_v4 cache
```

Related Topics

Configuring WLAN to Apply Flow Monitor in IPV4 and IPv6 Input/Output Direction, on page 1567
Wireless Flexible NetFlow Overview, on page 1544

Example: Configuring IPv6 and Transport Flag Flexible NetFlow in WLAN (Egress Direction)

The following example shows how to configure IPv6 and transport flag Flexible NetFlow on WLAN egress direction:

```
Switch# configure terminal
Switch(config)# flow record fr_v6
Switch(config-flow-record)# match ipv6 destination address
Switch(config-flow-record)# match ipv6 source address
Switch(config-flow-record)# match ipv6 hop-limit
Switch(config-flow-record)# match ipv6 protocol
Switch(config-flow-record)# match ipv6 traffic
Switch(config-flow-record)# match ipv6 version
Switch(config-flow-record)# match wireless ssid
Switch(config-flow-record)# collect wireless ap mac address
Switch(config-flow-record)# collect counter bytes long
Switch(config-flow-record)# collect transport tcp flags
Switch(config-flow-record)# exit

Switch(config)# flow monitor fm_v6
Switch(config-flow-monitor)# record fr_v6
Switch(config-flow-monitor)# exit

Switch(config)# wlan 1
Switch(config-wlan)# ipv6 flow monitor fm_v6 out
```

```
Switch(config-wlan)# end

Switch# show flow monitor fm_v6 cache
```

Note On the switch, you cannot specify which TCP flag to collect. You can only specify to collect transport TCP flags.

Related Topics

Configuring WLAN to Apply Flow Monitor in IPV4 and IPv6 Input/Output Direction, on page 1567
Wireless Flexible NetFlow Overview, on page 1544

Example: Configuring IPv6 Flexible NetFlow in WLAN (Both Ingress and Egress Directions)

The following example shows how to configure IPv6 Flexible NetFlow on WLAN in both directions:

```
Switch# configure terminal
Switch (config)# flow record fr_v6
Switch (config-flow-record)# match ipv6 destination address
Switch (config-flow-record)# match ipv6 source address
Switch (config-flow-record)# match ipv6 hop-limit
Switch (config-flow-record)# match ipv6 protocol
Switch (config-flow-record)# match ipv6 traffic
Switch (config-flow-record)# match ipv6 version
Switch (config-flow-record)# match wireless ssid
Switch (config-flow-record)# collect wireless ap mac address
Switch (config-flow-record)# collect counter packets long
Switch (config-flow-record)# exit

Switch (config)# flow monitor fm_v6
Switch (config-flow-monitor)# record fr_v6
Switch (config-flow-monitor)# exit

Switch (config)# wlan 1
Switch (config-wlan)# ipv6 flow monitor fm_v6 in
Switch (config-wlan)# ipv6 flow monitor fm_v6 out
Switch (config-wlan)# end

Switch# show flow monitor fm_v6 cache
```

Related Topics

Configuring WLAN to Apply Flow Monitor in IPV4 and IPv6 Input/Output Direction, on page 1567
Wireless Flexible NetFlow Overview, on page 1544

Additional References

Related Documents

Related Topic	Document Title
Flexible NetFlow CLI Commands	*Cisco Flexible NetFlow Command Reference (Catalyst 3850 Switches)* *Flexible NetFlow Command Reference, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
RFC 3954	Cisco Systems NetFlow Services Export Version 9

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for Flexible NetFlow

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.
Cisco IOS XE 3.3SE	The following new commands were added: • **match wireless ssid** • **collect wireless ap mac address**

PART XI

Stack Manager

CHAPTER 74

Managing Switch Stacks

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for Switch Stacks

All the switches in the switch stack need to be running the same license level as the active switch. For information about license levels, see the *System Management Configuration Guide (Catalyst 3850 Switches)*.

All switches in the switch stack need to be running compatible software versions.

Restrictions for Switch Stacks

The following are restrictions for your switch stack configuration:

- Switch stacks running the LAN Base license level do not support Layer 3 features.
- A switch stack can have up to nine stacking-capable switches connected through their ports.
- You cannot have a switch stack containing a mix of Catalyst 3850 and Catalyst 3650 switches.

Information About Switch Stacks

Switch Stack Overview

A switch stack can have up to nine stacking-capable switches connected through their StackWise-480 ports. The stack members work together as a unified system. Layer 2 and Layer 3 protocols present the entire switch stack as a single entity to the network.

A switch stack always has one active switch and one standby switch. If the active switch becomes unavailable, the standby switch assumes the role of the active switch, and continues to the keep the stack operational.

The active switch controls the operation of the switch stack, and is the single point of stack-wide management. From the active switch, you configure:

- System-level (global) features that apply to all stack members
- Interface-level features for each stack member

The active switch contains the saved and running configuration files for the switch stack. The configuration files include the system-level settings for the switch stack and the interface-level settings for each stack member. Each stack member has a current copy of these files for back-up purposes.

Supported Features in a Switch Stack

The system-level features supported on the are supported on the entire switch stack.

Switch Stack Membership

A standalone switch is a switch stack with one stack member that also operates as the . You can connect one standalone switch to another to create a switch stack containing two stack members, with one of them as the . You can connect standalone switches to an existing switch stack to increase the stack membership.

Stack Member Numbers

A new, out-of-the-box switch (one that has not joined a switch stack or has not been manually assigned a stack member number) ships with a default stack member number of 1. When it joins a switch stack, its default stack member number changes to the lowest available member number in the stack.

Stack members in the same switch stack cannot have the same stack member number. Every stack member, including a standalone switch, retains its member number until you manually change the number or unless the number is already being used by another member in the stack.

- If you manually change the stack member number by using the **switch** *current-stack-member-number* **renumber** *new-stack-member-number* EXEC command, the new number goes into effect after that stack member resets (or after you use the **reload slot** *stack-member-number* privileged EXEC command) and only if that number is not already assigned to any other members in the stack. Another way to change the stack member number is by changing the SWITCH_NUMBER environment variable.

  If the number is being used by another member in the stack, the switch selects the lowest available number in the stack.

  If you manually change the number of a stack member and no interface-level configuration is associated with that new member number, that stack member resets to its default configuration.

  You cannot use the **switch** *current-stack-member-number* **renumber** *new-stack-member-number* EXEC command on a provisioned switch. If you do, the command is rejected.

- If you move a stack member to a different switch stack, the stack member retains its number only if the number is not being used by another member in the stack. If it is being used, the switch selects the lowest available number in the stack.

- If you merge switch stacks, the switches that join the switch stack of a new select the lowest available numbers in the stack.

As described in the hardware installation guide, you can use the switch port LEDs in Stack mode to visually determine the stack member number of each stack member.

Stack Member Priority Values

A higher priority value for a stack member increases the probability of it being elected and retaining its stack member number. The priority value can be 1 to 15. The default priority value is 1. You can display the stack member priority value by using the **show switch** EXEC command.

Note We recommend assigning the highest priority value to the switch that you prefer to be the . This ensures that the switch is reelected as the if a reelection occurs.

To change the priority value for a stack member, use the **switch** *stack-member-number* **priority** *new priority-value* EXEC command.

The new priority value takes effect immediately but does not affect the current . The new priority value helps determine which stack member is elected as the new when the current or the switch stack resets.

Switch Stack Bridge ID and MAC Address

The MAC address of the determines the stack MAC address.

When the stack initializes, the MAC address of the determines the bridge ID that identifies the stack in the network.

If the changes, the MAC address of the new determines the new bridge ID and stack MAC address.

If the entire switch stack reloads, the switch stack uses the MAC address of the .

Persistent MAC Address on the Switch Stack

You can also configure stack MAC persistency so that the stack MAC address never changes to the new MAC address.

Election and Reelection

The is elected or reelected based on one of these factors and in the order listed:

1. The switch that is currently the .
2. The switch with the highest stack member priority value.

Note We recommend assigning the highest priority value to the switch that you prefer to be the . This ensures that the switch is reelected as if a reelection occurs.

3. The switch with the lowest MAC address.

Switch Stack Configuration Files

The configuration files record these settings:

- System-level (global) configuration settings such as IP, STP, VLAN, and SNMP settings that apply to all stack members
- Stack member interface-specific configuration settings that are specific for each stack member

Note The interface-specific settings of the are saved if the is replaced without saving the running configuration to the startup configuration.

A new, out-of-box switch joining a switch stack uses the system-level settings of that switch stack. If a switch is moved to a different switch stack before it is powered on, that switch loses its saved configuration file and uses the system-level configuration of the new switch stack. If the switch is powered on as a standalone switch before it joins the new switch stack, the stack will reload. When the stack reloads, the new switch may become the active switch, retain its configuration and overwrite the configuration files of the other stack members.

The interface-specific configuration of each stack member is associated with the stack member number. Stack members retain their numbers unless they are manually changed or they are already used by another member in the same switch stack. If the stack member number changes, the new number goes into effect after that stack member resets.

- If an interface-specific configuration does not exist for that member number, the stack member uses its default interface-specific configuration.
- If an interface-specific configuration exists for that member number, the stack member uses the interface-specific configuration associated with that member number.

If you replace a failed member with an identical model, the replacement member automatically uses the same interface-specific configuration as the failed switch. You do not need to reconfigure the interface settings.

The replacement switch (referred to as the provisioned switch) must have the same stack member number as the failed switch.

You back up and restore the stack configuration in the same way as you would for a standalone switch configuration.

Offline Configuration to Provision a Stack Member

You can use the offline configuration feature to *provision* (to supply a configuration to) a new switch before it joins the switch stack. You can configure the stack member number, the switch type, and the interfaces associated with a switch that is not currently part of the stack. The configuration that you create on the switch stack is called the *provisioned configuration*. The switch that is added to the switch stack and that receives this configuration is called the *provisioned switch*.

You manually create the provisioned configuration through the **switch** *stack-member-number* **provision** *type* global configuration command. You must change the *stack-member-number* on the provisioned switch before you add it to the stack, and it must match the stack member number that you created for the new switch on the switch stack. The switch type in the provisioned configuration must match the switch type of the newly added switch. The provisioned configuration is automatically created when a switch is added to a switch stack and when no provisioned configuration exists.

When you configure the interfaces associated with a provisioned switch, the switch stack accepts the configuration, and the information appears in the running configuration. However, as the switch is not active, any configuration on the interface is not operational and the interface associated with the provisioned switch does not appear in the display of the specific feature. For example, VLAN configuration information associated with a provisioned switch does not appear in the **show vlan** user EXEC command output on the switch stack.

The switch stack retains the provisioned configuration in the running configuration whether or not the provisioned switch is part of the stack. You can save the provisioned configuration to the startup configuration file by entering the **copy running-config startup-config** privileged EXEC command. The startup configuration file ensures that the switch stack can reload and can use the saved information whether or not the provisioned switch is part of the switch stack.

Effects of Adding a Provisioned Switch to a Switch Stack

When you add a provisioned switch to the switch stack, the stack applies either the provisioned configuration or the default configuration. This table lists the events that occur when the switch stack compares the provisioned configuration with the provisioned switch.

Table 163: Results of Comparing the Provisioned Configuration with the Provisioned Switch

Scenario		Result
The stack member numbers and the switch types match.	**1** If the stack member number of the provisioned switch matches the stack member number in the provisioned configuration on the stack, and **2** If the switch type of the provisioned switch matches the switch type in the provisioned configuration on the stack.	The switch stack applies the provisioned configuration to the provisioned switch and adds it to the stack.

Scenario		Result
The stack member numbers match but the switch types do not match.	**1** If the stack member number of the provisioned switch matches the stack member number in the provisioned configuration on the stack, but **2** The switch type of the provisioned switch does not match the switch type in the provisioned configuration on the stack.	The switch stack applies the default configuration to the provisioned switch and adds it to the stack. The provisioned configuration is changed to reflect the new information.
The stack member number is not found in the provisioned configuration.		The switch stack applies the default configuration to the provisioned switch and adds it to the stack. The provisioned configuration is changed to reflect the new information.
The stack member number of the provisioned switch is not found in the provisioned configuration.		The switch stack applies the default configuration to the provisioned switch and adds it to the stack.

If you add a provisioned switch that is a different type than specified in the provisioned configuration to a powered-down switch stack and then apply power, the switch stack rejects the (now incorrect) **switch** *stack-member-number* **provision** *type* global configuration command in the startup configuration file. However, during stack initialization, the nondefault interface configuration information in the startup configuration file for the provisioned interfaces (potentially of the wrong type) is executed. Depending on the differences between the actual switch type and the previously provisioned switch type, some commands are rejected, and some commands are accepted.

Note If the switch stack does not contain a provisioned configuration for a new switch, the switch joins the stack with the default interface configuration. The switch stack then adds to its running configuration with a **switch** *stack-member-number* **provision** *type* global configuration command that matches the new switch.

Effects of Replacing a Provisioned Switch in a Switch Stack

When a provisioned switch in a switch stack fails, it is removed from the stack, and is replaced with another switch, the stack applies either the provisioned configuration or the default configuration to it. The events that occur when the switch stack compares the provisioned configuration with the provisioned switch are the same as those when you add a provisioned switch to a stack.

Effects of Removing a Provisioned Switch from a Switch Stack

If you remove a provisioned switch from the switch stack, the configuration associated with the removed stack member remains in the running configuration as provisioned information. To completely remove the configuration, use the **no switch** *stack-member-number* **provision** global configuration command.

Stack Protocol Version

Each software image includes a *stack protocol version*. The stack protocol version has a *major* version number and a *minor* version number (for example 1.4, where 1 is the major version number and 4 is the minor version number). Both version numbers determine the level of compatibility among the stack members.

Switches with the same Cisco IOS software version have the same stack protocol version. Such switches are fully compatible, and all features function properly across the switch stack. A switch with the same Cisco IOS software version as the can immediately join the switch stack.

If an incompatibility exists, the fully functional stack members generate a system message that describes the cause of the incompatibility on the specific stack members. The sends the message to all stack members.

Major Stack Protocol Version Number Incompatibility Among Stack-Capable Switches

Switches with different major Cisco IOS software versions usually have different stack protocol versions. Switches with different major version numbers are incompatible and cannot exist in the same switch stack.

Minor Stack Protocol Version Number Incompatibility Among Stack-Capable Switches

Switches with the same major version number but with a different minor version number are considered partially compatible. When connected to a switch stack, a partially compatible switch enters version-mismatch (VM) mode and cannot join the stack as a fully functioning member. The software detects the mismatched software and tries to upgrade (or downgrade) the switch in VM mode with the switch stack image or with a tar file image from the switch stack flash memory. The software uses the automatic upgrade (auto-upgrade) and the automatic advise (auto-advise) features.

SDM Template Mismatch in Switch Stacks

All stack members use the Switch Database Management (SDM) template configured on the . When a new switch is added to a stack, the SDM configuration that is stored on the overrides the template configured on an individual switch.

You can use the **show switch** privileged EXEC command to see if any stack members are in SDM-mismatch mode.

Version-mismatch (VM) mode has priority over SDM-mismatch mode. If a VM-mode condition and an SDM-mismatch mode exist, the switch stack first attempts to resolve the VM-mode condition.

Switch Stack Management Connectivity

You manage the switch stack and the stack member interfaces through the . You can use the CLI, SNMP, and supported network management applications such as CiscoWorks. You cannot manage stack members on an individual switch basis.

Note Use SNMP to manage network features across the stack that are defined by supported MIBs. The switch does not support MIBs to manage stacking-specific features such as stack membership and election.

Connectivity to Specific Stack Members

If you want to configure a specific stack member port, you must include the stack member number in the CLI command interface notation.

Connectivity to the Switch Stack Through an IP Address

The switch stack is managed through a single IP address. The IP address is a system-level setting and is not specific to the or to any other stack member. You can still manage the stack through the same IP address even if you remove the or any other stack member from the stack, provided there is IP connectivity.

Note Stack members retain their IP addresses when you remove them from a switch stack. To avoid a conflict by having two devices with the same IP address in your network, change the IP addresses of any switches that you remove from the switch stack.

Connectivity to the Switch Stack Through an SSH Session

The Secure Shell (SSH) connectivity to the stack can be lost if a running the cryptographic version fails and is replaced by a switch that is running a noncryptographic version. We recommend that a switch running the cryptographic version of the software be the . Encryption features are unavailable if the is running the noncryptographic software image.

Connectivity to the Switch Stack Through Console Ports or Ethernet Management Ports

You can connect to the by using one of these methods:

- You can connect a terminal or a PC to the through the console port of one or more stack members.
- You can connect a PC to the through the Ethernet management ports of one or more stack members.

Be careful when using multiple CLI sessions to the . Commands that you enter in one session are not displayed in the other sessions. Therefore, it is possible that you might not be able to identify the session from which you entered a command.

We recommend using only one CLI session when managing the switch stack.

Switch Stack Upgrade

Before you start the process, configure a redundant uplink to the network to ensure that the stack has network connectivity. You can also set the stack boot time.

When upgrading the whole switch stack, or a subset of switches in the stack, the communicates with each stack member to coordinate the upgrade. When the upgrade completes, all switches in the stack reload to complete the installation.

The copies the source bundle to each member in the stack. Each stack member then performs various installation steps. If any step fails on a stack member, the instructs all stack members to abort the installation.

If the becomes unavailable during the installation, all stack members abort the installation.

Auto-Upgrade

The purpose of the auto-upgrade feature is to allow a switch to be upgraded to a compatible software image, so that the switch can join the switch stack.

When a new switch attempts to join a switch stack, each stack member performs compatibility checks with itself and the new switch. Each stack member sends the results of the compatibility checks to the , which uses the results to determine whether the switch can join the switch stack. If the software on the new switch is incompatible with the switch stack, the new switch enters version-mismatch (VM) mode.

If the auto-upgrade feature is enabled on the new switch, the automatically upgrades the new switch with the same software image running on a compatible stack member. Auto-upgrade starts a few minutes after the mismatched software is detected before starting.

Auto-upgrade is disabled by default.

Auto-upgrade includes an auto-copy process and an auto-extract process.

- Auto-copy automatically copies the software image running on any stack member to the new switch to automatically upgrade it. Auto-copy occurs if auto-upgrade is enabled, if there is enough flash memory in the new switch, and if the software image running on the switch stack is suitable for the new switch.

  Note A switch in VM mode might not run all released software. For example, new switch hardware is not recognized in earlier versions of software.

- Automatic extraction (auto-extract) occurs when the auto-upgrade process cannot find the appropriate software in the stack to copy to the new switch. In that case, the auto-extract process searches all switches in the stack for the bin file needed to upgrade the switch stack or the new switch. The bin file can be in any flash file system in the switch stack or in the new switch. If a bin file suitable for the new switch is found on a stack member, the process extracts the file and automatically upgrades the new switch.

The auto-upgrade feature is not available in bundle mode. The switch stack must be running in installed mode. If the switch stack is in bundle mode, use the **software expand** privileged EXEC command to change to installed mode.

You can enable auto-upgrade by using the **software auto-upgrade enable** global configuration command on the new switch. You can check the status of auto-upgrade by using the **show running-config** privileged EXEC command and by checking the *Auto upgrade* line in the display.

You can configure auto-upgrade to upgrade the new switch with a specific software bundle by using the **software auto-upgrade source url** global configuration command. If the software bundle is invalid, the new switch is upgraded with the same software image running on a compatible stack member.

When the auto-upgrade process is complete, the new switch reloads and joins the stack as a fully functioning member. If you have both stack cables connected during the reload, network downtime does not occur because the switch stack operates on two rings.

For more information about upgrading a switch running incompatible software see the *Cisco IOS File System, Configuration Files, and Bundle Files Appendix, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*.

Auto-Advise

Examples of Auto-Advise Messages

Auto-Upgrade Is Disabled and Incompatible Switch Attempting to Join: Example

This sample auto-advise output shows the system messages displayed when the auto-upgrade feature is disabled and an incompatible switch 1 tries to join the switch stack:

```
*Oct 18 08:36:19.379: %INSTALLER-6-AUTO_ADVISE_SW_INITIATED: 2 installer:  Auto advise
initiated for switch 1
*Oct 18 08:36:19.380: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer:  Searching stack for software
 to upgrade switch 1
*Oct 18 08:36:19.382: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer:  Switch 1 with incompatible
 software has been
*Oct 18 08:36:19.382: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer:  added to the stack. The
software running on
*Oct 18 08:36:19.382: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer:  all stack members was
scanned and it has been
*Oct 18 08:36:19.382: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer:  determined that the 'software
 auto-upgrade'
*Oct 18 08:36:19.382: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer:  command can be used to
install compatible
*Oct 18 08:36:19.382: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer:  software on switch 1.
```

Auto-Upgrade is Disabled and New Switch is in Bundle Mode: Example

This sample auto-advise output shows the system messages displayed when auto-upgrade is disabled and a switch running in bundle mode tries to join the stack that is running in installed mode:

```
*Oct 18 11:09:47.005: %INSTALLER-6-AUTO_ADVISE_SW_INITIATED: 2 installer:  Auto advise
initiated for switch 1
*Oct 18 11:09:47.005: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer:  Switch 1 running bundled
software has been added
*Oct 18 11:09:47.005: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer:  to the stack that is running
 installed software.
*Oct 18 11:09:47.005: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer:  The 'software auto-upgrade'
 command can be used to
*Oct 18 11:09:47.005: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer:  convert switch 1 to the
installed running mode by
*Oct 18 11:09:47.005: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer:  installing its running
software.
```

How to Configure a Switch Stack

Enabling the Persistent MAC Address Feature

This procedure is optional.

Note When you enter the command to configure this feature, a warning message appears with the consequences of your configuration. You should use this feature cautiously. Using the old MAC address elsewhere in the same domain could result in lost traffic.

SUMMARY STEPS

1. **configure terminal**
2. **stack-mac persistent timer** [**0** | *time-value*]
3. **end**
4. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**stack-mac persistent timer** [**0** | *time-value*] **Example:** `Switch(config)# stack-mac persistent timer 7`	Enables a time delay after a stack-master change before the stack MAC address changes to that of the new stack master. If the previous stack master rejoins the stack during this period, the stack uses that MAC address as the stack MAC address. You can configure the time period as 0 to 60 minutes. • Enter the command with no value to set the default delay of approximately 4 minutes. We recommend that you always enter a value. If the command is entered without a value, the time delay appears in the running-config file with an explicit timer value of 4 minutes. • Enter **0** to continue using the MAC address of the current stack master indefinitely. The stack MAC address of the previous stack master is used until you enter the **no stack-mac persistent timer** command, which immediately changes the stack MAC address to that of the current stack master. • Enter a *time-value* from 1 to 60 minutes to configure the time period before the stack MAC address changes to the new stack master. The stack MAC address of the previous stack master is used until the configured time period expires or until you enter the **no stack-mac persistent timer** command. **Note** If you enter the **no stack-mac persistent timer** command after a new stack master takes over, before the time expires, the switch stack moves to the current stack master MAC address.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Example: `Switch# copy running-config startup-config`	

Assigning a Stack Member Number

This optional task is available only from the .

SUMMARY STEPS

1. **switch** *current-stack-member-number* **renumber** *new-stack-member-number*
2. **reload slot** *stack-member-number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**switch** *current-stack-member-number* **renumber** *new-stack-member-number* **Example:** `Switch# switch 3 renumber 4`	You can display the current stack member number by using the **show switch** user EXEC command.
Step 2	**reload slot** *stack-member-number* **Example:** `Switch# reload slot 4`	Resets the stack member.

Setting the Stack Member Priority Value

This optional task is available only from the .

SUMMARY STEPS

1. **switch** *stack-member-number* **priority** *new-priority-number*
2. **reload slot** *stack-member-number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**switch** *stack-member-number* **priority** *new-priority-number* **Example:** `Switch# switch 3 priority 2`	You can display the current priority value by using the **show switch** user EXEC command. The new priority value takes effect immediately but does not affect the current . The new priority value helps determine which stack member is elected as the new when the current or switch stack resets.
Step 2	**reload slot** *stack-member-number* **Example:** `Switch# reload slot 3`	Specifies the stack member number and the new priority for the stack member. The stack member number range is 1 to 9. The priority value range is 1 to 15. You can display the current priority value by using the **show switch** user EXEC command. The new priority value takes effect immediately but does not affect the current active switch. The new priority value helps determine which stack member is elected as the new active switch when the current active switch or switch stack resets.

Removing Provisioned Switch Information

Before you begin, you must remove the provisioned switch from the stack. This optional task is available only from the .

SUMMARY STEPS

1. **configure terminal**
2. **no switch** *stack-member-number* **provision**
3. **end**
4. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**no switch** *stack-member-number* **provision** **Example:** `Switch(config)# no switch 3 provision`	Removes the provisioning information for the specified member.

	Command or Action	Purpose
Step 3	**end** **Example:** `Switch(config)#` **`end`**	Returns to privileged EXEC mode.
Step 4	**copy running-config startup-config** **Example:** `Switch#` **`copy running-config startup-config`**	(Optional) Saves your entries in the configuration file.

If you are removing a provisioned switch in a stack with this configuration:

- The stack has four members
- Stack member 1 is the
- Stack member 3 is a provisioned switch

and want to remove the provisioned information and to avoid receiving an error message, you can remove power from stack member 3, disconnect the cables between the stack member 3 and switches to which it is connected, reconnect the cables between the remaining stack members, and enter the **no switch** *stack-member-number* **provision** global configuration command.

Displaying Incompatible Switches in the Switch Stack

SUMMARY STEPS

1. **show switch**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show switch** **Example:** `Switch#` **`show switch`**	Displays any incompatible switches in the switch stack (indicated by a 'Current State' of 'V-Mismatch'). The V-Mismatch state identifies the switches with incompatible software. The output displays Lic-Mismatch for switches that are not running the same license level as the . For information about managing license levels, see the *System Management Configuration Guide (Catalyst 3850 Switches)*.

Upgrading an Incompatible Switch in the Switch Stack

SUMMARY STEPS

1. **software auto-upgrade**
2. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**software auto-upgrade** **Example:** `Switch# software auto-upgrade`	Upgrades incompatible switches in the switch stack, or changes switches in bundle mode to installed mode.
Step 2	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Troubleshooting the Switch Stack

Accessing the Diagnostic Console of a Stack Member

Before You Begin

This optional task is available only from the active switch.

SUMMARY STEPS

1. **session switch** *stack-member-number*
2. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**session switch** *stack-member-number* **Example:** `Switch# session switch 2`	Accesses the diagnostic shell of the stack member from the active switch.

	Command or Action	Purpose
Step 2	**exit** **Example:** `Switch(diag)> exit`	Returns to the CLI session on the .

Temporarily Disabling a Stack Port

If a stack port is flapping and causing instability in the stack ring, to disable the port, enter the **switch** *stack-member-number* **stack port** *port-number* **disable** privileged EXEC command. To reenable the port, enter the **switch** *stack-member-number* **stack port** *port-number* **enable** command.

Note Be careful when using the **switch** *stack-member-number* **stack port** *port-number* **disable** command. When you disable the stack port, the stack operates at half bandwidth.

A stack is in the full-ring state when all members are connected through the stack ports and are in the ready state.

The stack is in the partial-ring state when the following occurs:

- All members are connected through their stack ports but some are not in the ready state.
- Some members are not connected through the stack ports.

SUMMARY STEPS

1. **switch** *stack-member-number* **stack port** *port-number* **disable**
2. **switch** *stack-member-number* **stack port** *port-number* **enable**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**switch** *stack-member-number* **stack port** *port-number* **disable** **Example:** `Switch# switch 2 stack port 1 disable`	Disables the specified stack port.
Step 2	**switch** *stack-member-number* **stack port** *port-number* **enable** **Example:** `Switch# switch 2 stack port 1 enable`	Reenables the stack port.

When you disable a stack port and the stack is in the full-ring state, you can disable only one stack port. This message appears:

```
Enabling/disabling a stack port may cause undesired stack changes. Continue?[confirm]
```

When you disable a stack port and the stack is in the partial-ring state, you cannot disable the port. This message appears:

```
Disabling stack port not allowed with current stack configuration.
```

Reenabling a Stack Port While Another Member Starts

Stack Port 1 on Switch 1 is connected to Port 2 on Switch 4. If Port 1 is flapping, you can disable Port 1 with the **switch 1 stack port 1 disable** privileged EXEC command. While Port 1 on Switch 1 is disabled and Switch 1 is still powered on, follow these steps to reenable a stack port:

Step 1 Disconnect the stack cable between Port 1 on Switch 1 and Port 2 on Switch 4.

Step 2 Remove Switch 4 from the stack.

Step 3 Add a switch to replace Switch 4 and assign it switch-number 4.

Step 4 Reconnect the cable between Port 1 on Switch 1 and Port 2 on Switch 4 (the replacement switch).

Step 5 Reenable the link between the switches. Enter the **switch 1 stack port 1 enable** privileged EXEC command to enable Port 1 on Switch 1.

Step 6 Power on Switch 4.

Caution Powering on Switch 4 before enabling the Port 1 on Switch 1 might cause one of the switches to reload.

If Switch 4 is powered on first, you might need to enter the **switch 1 stack port 1 enable** and the **switch 4 stack port 2 enable** privileged EXEC commands to bring up the link.

Monitoring the Switch Stack

Table 164: Commands for Displaying Stack Information

Command	Description
show switch	Displays summary information about the stack, including the status of provisioned switches and switches in version-mismatch mode.
show switch *stack-member-number*	Displays information about a specific member.
show switch detail	Displays detailed information about the stack.
show switch neighbors	Displays the stack neighbors.

Command	Description
show switch stack-ports [**summary**]	Displays port information for the stack. Use the **summary** keyword to display the stack cable length, the stack link status, and the loopback status.
show redundancy	Displays the redundant system and the current processor information. The redundant system information includes the system uptime, standby failures, switchover reason, hardware, configured and operating redundancy mode. The current processor information displayed includes the active location, the software state, the uptime in the current state and so on.
show redundancy state	Displays all the redundancy states of the active and standby switches.

Configuration Examples for Switch Stacks

Switch Stack Configuration Scenarios

Most of these switch stack configuration scenarios assume that at least two switches are connected through their ports.

Table 165: Configuration Scenarios

Scenario		Result
election specifically determined by existing	Connect two powered-on switch stacks through the ports.	Only one of the two becomes the new .
election specifically determined by the stack member priority value	**1** Connect two switches through their ports. **2** Use the **switch** *stack-member-number* **priority** *new-priority-number* EXEC command to set one stack member with a higher member priority value. **3** Restart both stack members at the same time.	The stack member with the higher priority value is elected .

Scenario		Result
election specifically determined by the configuration file	Assuming that both stack members have the same priority value: **1** Make sure that one stack member has a default configuration and that the other stack member has a saved (nondefault) configuration file. **2** Restart both stack members at the same time.	The stack member with the saved configuration file is elected .
election specifically determined by the MAC address	Assuming that both stack members have the same priority value, configuration file, and license level, restart both stack members at the same time.	The stack member with the lower MAC address is elected .
Stack member number conflict	Assuming that one stack member has a higher priority value than the other stack member: **1** Ensure that both stack members have the same stack member number. If necessary, use the **switch** *current-stack-member-number* **renumber** *new-stack-member-number* EXEC command. **2** Restart both stack members at the same time.	The stack member with the higher priority value retains its stack member number. The other stack member has a new stack member number.
Add a stack member	**1** Power off the new switch. **2** Through their ports, connect the new switch to a powered-on switch stack. **3** Power on the new switch.	The is retained. The new switch is added to the switch stack.
failure	Remove (or power off) the .	One of the remaining stack members becomes the new stack master. All other stack members in the stack remain as stack members and do not reboot.

Scenario		Result
	1. Through their ports, connect switches. 2. Power on all switches.	Two switches become . One has stack members. The other remains as a standalone switch. Use the Mode button and port LEDs on the switches to identify which switches are and which switches belong to each .

Enabling the Persistent MAC Address Feature: Example

This example shows how to configure the persistent MAC address feature for a 7-minute time delay and to verify the configuration:

```
Switch(config)# stack-mac persistent timer 7
  WARNING: The stack continues to use the base MAC of the old Master
  WARNING: as the stack MAC after a master switchover until the MAC
  WARNING: persistency timer expires. During this time the Network
  WARNING: Administrators must make sure that the old stack-mac does
  WARNING: not appear elsewhere in this network domain. If it does,
  WARNING: user traffic may be blackholed.
  Switch(config)# end
  Switch# show switch
  Switch/Stack Mac Address : 0016.4727.a900
  Mac persistency wait time: 7 mins
                                           H/W   Current
  Switch#  Role   Mac Address     Priority Version  State
  ----------------------------------------------------------
  *1        0016.4727.a900     1      P2B     Ready
```

Provisioning a New Member for a Switch Stack: Example

This example shows how to provision a switch with a stack member number of 2 for the switch stack. The **show running-config** command output shows the interfaces associated with the provisioned switch:

```
Switch(config)# switch 2 provision switch_PID
  Switch(config)# end
  Switch# show running-config | include switch 2
  !
  interface GigabitEthernet2/0/1
  !
  interface GigabitEthernet2/0/2
  !
  interface GigabitEthernet2/0/3
  <output truncated>
```

show switch stack-ports summary Command Output: Example

Only Port 1 on stack member 2 is disabled.

```
Switch# show switch stack-ports summary
  Switch#/  Stack   Neighbor   Cable    Link   Link   Sync      #         In
   Port#     Port              Length    OK   Active   OK    Changes   Loopback
            Status                                          To LinkOK
```

```
--------  ------  --------  ---------  ----  ------  ----  ---------  --------
  1/1     OK         3      50 cm      Yes    Yes    Yes        1        No
  1/2     Down      None    3 m        Yes    No     Yes        1        No
  2/1     Down      None    3 m        Yes    No     Yes        1        No
  2/2     OK         3      50 cm      Yes    Yes    Yes        1        No
  3/1     OK         2      50 cm      Yes    Yes    Yes        1        No
  3/2     OK         1      50 cm      Yes    Yes    Yes        1        No
```

Table 166: show switch stack-ports summary Command Output

Field	Description
Switch#/Port#	Member number and its stack port number.
Stack Port Status	Status of the stack port. • Absent—No cable is detected on the stack port. • Down—A cable is detected, but either no connected neighbor is up, or the stack port is disabled. • OK—A cable is detected, and the connected neighbor is up.
Neighbor	Switch number of the active member at the other end of the stack cable.
Cable Length	Valid lengths are 50 cm, 1 m, or 3 m. If the switch cannot detect the cable length, the value is *no cable*. The cable might not be connected, or the link might be unreliable.
Link OK	Whether the stack cable is connected and functional. There may or may not be a neighbor connected on the other end. The *link partner* is a stack port on a neighbor switch. • No—There is no stack cable connected to this port or the stack cable is not functional. • Yes—There is a functional stack cable connected to this port.
Link Active	Whether a neighbor is connected on the other end of the stack cable. • No—No neighbor is detected on the other end. The port cannot send traffic over this link. • Yes—A neighbor is detected on the other end. The port can send traffic over this link.

Field	Description
Sync OK	Whether the link partner sends valid protocol messages to the stack port. • No—The link partner does not send valid protocol messages to the stack port. • Yes—The link partner sends valid protocol messages to the port.
# Changes to LinkOK	The relative stability of the link. If a large number of changes occur in a short period of time, link flapping can occur.
In Loopback	Whether a stack cable is attached to a stack port on the member. • No—At least one stack port on the member has an attached stack cable. • Yes—None of the stack ports on the member has an attached stack cable.

Software Loopback: Examples

In a stack with three members, stack cables connect all the members:

```
Switch# show switch stack-ports summary
  Switch#
 Sw#/Port#  Port    Neighbor  Cable     Link  Link    Sync   #Changes   In
            Status            Length    OK    Active  OK     To LinkOK  Loopback
  --------  ------  --------  --------  ----  ------  ----  ---------   --------
    1/1     OK         3      50 cm     Yes    Yes    Yes        1        No
    1/2     OK         2      3 m       Yes    Yes    Yes        1        No
    2/1     OK         1      3 m       Yes    Yes    Yes        1        No
    2/2     OK         3      50 cm     Yes    Yes    Yes        1        No
    3/1     OK         2      50 cm     Yes    Yes    Yes        1        No
    3/2     OK         1      50 cm     Yes    Yes    Yes        1        No
```

If you disconnect the stack cable from Port 1 on Switch 1, these messages appear:

```
01:09:55: %STACKMGR-4-STACK_LINK_CHANGE: Stack Port 2 Switch 3 has changed to state DOWN
  01:09:56: %STACKMGR-4-STACK_LINK_CHANGE: Stack Port 1 Switch 1 has changed to state DOWN

  Switch# show switch stack-ports summary
  Switch#
 Sw#/Port#  Port    Neighbor  Cable     Link  Link    Sync  #Changes   In
            Status            Length    OK    Active  OK    To LinkOK  Loopback

  --------  ------  --------  --------  ----  ------  ----  ---------  --------
    1/1     Absent    None    No cable  No     No     No         1        No
    1/2     OK         2      3 m       Yes    Yes    Yes        1        No
    2/1     OK         1      3 m       Yes    Yes    Yes        1        No
    2/2     OK         3      50 cm     Yes    Yes    Yes        1        No
    3/1     OK         2      50 cm     Yes    Yes    Yes        1        No
    3/2     Down      None    50 cm     No     No     No         1        No
```

If you disconnect the stack cable from Port 2 on Switch 1, the stack splits.

Switch 2 and Switch 3 are now in a two-member stack connected through stack cables:

```
Switch# show sw stack-ports summary
  Switch#
 Sw#/Port#  Port    Neighbor  Cable     Link  Link    Sync  #Changes   In
            Status            Length    OK    Active  OK    To LinkOK  Loopback
  --------  ------  --------  --------  ----  ------  ----  ---------  --------
    2/1     Down      None    3 m       No     No     No        1        No
    2/2     OK         3      50 cm     Yes    Yes    Yes       1        No
    3/1     OK         2      50 cm     Yes    Yes    Yes       1        No
    3/2     Down      None    50 cm     No     No     No        1        No
```

Switch 1 is a standalone switch:

```
Switch# show switch stack-ports summary
  Switch#
 Sw#/Port#  Port    Neighbor  Cable     Link  Link    Sync  #Changes   In
            Status            Length    OK    Active  OK    To LinkOK  Loopback
  --------  ------  --------  --------  ----  ------  ----  ---------  --------
    1/1     Absent    None    No cable  No     No     No        1        Yes
    1/2     Absent    None    No cable  No     No     No        1        Yes
```

Software Loopback with no Connected Stack Cable: Example

```
Switch# show switch stack-ports summary
  Switch#
 Sw#/Port#  Port    Neighbor  Cable     Link  Link    Sync  #Changes   In
            Status            Length    OK    Active  OK    To LinkOK  Loopback
  --------  ------  --------  --------  ----  ------  ----  ---------  --------
    1/1     Absent    None    No cable   No     No     No      1         Yes
    1/2     Absent    None    No cable   No     No     No      1         Yes
```

Software Loopback with Connected Stack Cables: Examples

- On Port 1 on Switch 1, the port status is *Down*, and a cable is connected.

  On Port 2 on Switch 1, the port status is *Absent*, and no cable is connected.

```
Switch# show switch stack-ports summary
  Switch#
 Sw#/Port#  Port    Neighbor  Cable     Link  Link    Sync  #Changes   In
            Status            Length    OK    Active  OK    To LinkOK  Loopback
  --------  ------  --------  --------  ---- ------  ----  ---------  --------
    1/1     Down      None    50 Cm      No     No     No       1          No
    1/2     Absent    None    No cable   No     No     No       1          No
```

- In a *physical loopback*, a cable connects both stack ports on a switch. You can use this configuration to test
  - Cables on a switch that is running properly
  - Stack ports with a cable that works properly

```
Switch# show switch stack-ports summary
  Switch#
 Sw#/Port#  Port    Neighbor  Cable     Link  Link    Sync  #Changes   In
            Status            Length    OK    Active  OK    To LinkOK  Loopback
  --------  ------  --------  --------  ----  ------  ----  ---------  --------
```

```
2/1       OK       2      50 cm     Yes    Yes    Yes       1          No
2/2       OK       2      50 cm     Yes    Yes    Yes       1          No
```

The port status shows that

- ◦ Switch 2 is a standalone switch.
- ◦ The ports can send and receive traffic.

Finding a Disconnected Stack Cable: Example

Stack cables connect all stack members. Port 2 on Switch 1 connects to Port 1 on Switch 2.

This is the port status for the members:

```
Switch# show switch stack-ports summary
  Switch#
 Sw#/Port#  Port    Neighbor  Cable     Link  Link    Sync  #Changes   In
            Status            Length    OK    Active  OK    To LinkOK  Loopback
  --------  ------  --------  --------  ----  ------  ----  ---------  --------
    1/1     OK         2      50 cm     Yes    Yes    Yes       0         No
    1/2     OK         2      50 cm     Yes    Yes    Yes       0         No
    2/1     OK         1      50 cm     Yes    Yes    Yes       0         No
    2/2     OK         1      50 cm     Yes    Yes    Yes       0         No
```

If you disconnect the cable from Port 2 on Switch 1, these messages appear:

```
%STACKMGR-4-STACK_LINK_CHANGE: Stack Port 1 Switch 2 has changed to state DOWN

  %STACKMGR-4-STACK_LINK_CHANGE: Stack Port 2 Switch 1 has changed to state DOWN
```

This is now the port status:

```
  Switch# show switch stack-ports summary
  Switch#
 Sw#/Port#  Port    Neighbor  Cable     Link  Link    Sync  #Changes   In
            Status            Length    OK    Active  OK    To LinkOK  Loopback
  --------  ------  --------  --------  ----  ------  ----  ---------  --------
    1/1       OK       2      50 cm     Yes    Yes    Yes       1         No
    1/2     Absent    None    No cable   No     No     No       2         No
    2/1      Down     None    50 cm      No     No     No       2         No
    2/2       OK       1      50 cm     Yes    Yes    Yes       1         No
```

Only one end of the cable connects to a stack port, Port 1 on Switch 2.

- The *Stack Port Status* value for Port 2 on Switch 1 is *Absent*, and the value for Port 1 on Switch 2 is *Down*.
- The *Cable Length* value is *No cable*.

Diagnosing the problem:

- Verify the cable connection for Port 2 on Switch 1.
- Port 2 on Switch 1 has a port or cable problem if
  - ◦ The *In Loopback* value is *Yes*.

  or

◦ The *Link OK*, *Link Active*, or *Sync OK* value is *No*.

Fixing a Bad Connection Between Stack Ports: Example

Stack cables connect all members. Port 2 on Switch 1 connects to Port 1 on Switch 2.

This is the port status:

```
Switch# show switch stack-ports summary
  Switch#
 Sw#/Port#   Port     Neighbor   Cable      Link   Link     Sync   #Changes    In
             Status              Length     OK     Active   OK     To LinkOK   Loopback
  --------   ------   --------   --------   ----   ------   ----   ---------   --------
    1/1        OK        2       50 cm      Yes     Yes     Yes        1          No
    1/2       Down     None      50 cm       No      No      No        2          No
    2/1       Down     None      50 cm       No      No      No        2          No
    2/2        OK        1       50 cm      Yes     Yes     Yes        1          No
```

Diagnosing the problem:

- The Stack Port Status value is *Down*.
- Link OK, Link Active, and Sync OK values are *No*.
- The Cable Length value is *50 cm*. The switch detects and correctly identifies the cable.

The connection between Port 2 on Switch 1 and Port 1 on Switch 2 is unreliable on at least one of the connector pins.

Additional References for Switch Stacks

Related Documents

Related Topic	Document Title
Cabling and powering on a switch stack.	*Catalyst 3850 Switch Hardware Installation Guide*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and licensed feature sets,, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Switch Stacks

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 75

Configuring Cisco NSF with SSO

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for NSF with SSO

The following are prerequisites and considerations for configuring NSF with SSO.

- Use of the routing protocols requires the IP Services license level. EIGRP-stub and OSPF for routed access are supported on IP Base license level.
- BGP support in NSF requires that neighbor networking devices be NSF-aware; that is, the devices must have the graceful restart capability and advertise that capability in their OPEN message during session establishment. If an NSF-capable router discovers that a particular BGP neighbor does not have graceful restart capability, it does not establish an NSF-capable session with that neighbor. All other neighbors

that have graceful restart capability continue to have NSF-capable sessions with this NSF-capable networking device.

- OSPF support in NSF requires that all neighbor networking devices be NSF-aware. If an NSF-capable router discovers that it has non-NSF -aware neighbors on a particular network segment, it disables NSF capabilities for that segment. Other network segments composed entirely of NSF-capable or NSF-aware routers continue to provide NSF capabilities.

Restrictions for NSF with SSO

The following are restrictions for configuring NSF with SSO:

- NSF capability is supported for IPv4 routing protocols only. NSF capability is not supported for IPv6 routing protocols.
- NSF does not support IP Multicast Routing, as it is not SSO-aware.
- NSF is not supported if the IOS-XE software is running in the LAN Base mode.
- For NSF operation, you must have SSO configured on the device.
- NSF with SSO supports IP Version 4 traffic and protocols only; NSF with SSO does not support IPv6 traffic.
- All Layer 3 neighboring devices must be NSF Helper or NSF-capable to support graceful restart capability.
- For IETF, all neighboring devices must be running an NSF-aware software image.

Information About NSF with SSO

Overview of NSF with SSO

The switch supports fault resistance by allowing a standby switch to take over if the active switch becomes unavailable. Cisco nonstop forwarding (NSF) works with stateful switchover (SSO) to minimize the amount of time a network is unavailable.

NSF provides these benefits:

- Improved network availability—NSF continues forwarding network traffic and application state information so that user session information is maintained after a switchover.
- Overall network stability—Network stability may be improved with the reduction in the number of route flaps, which were created when routers in the network failed and lost their routing tables.
- Neighboring routers do not detect a link flap—Because the interfaces remain up during a switchover, neighboring routers do not detect a link flap (the link does not go down and come back up).
- Prevents routing flaps—Because SSO continues forwarding network traffic during a switchover, routing flaps are avoided.
- Maintains user sessions established prior to the switchover.

SSO Operation

When a standby switch runs in SSO mode, the standby switch starts up in a fully-initialized state and synchronizes with the persistent configuration and the running configuration of the active switch. It subsequently maintains the state on the protocols listed below, and all changes in hardware and software states for features that support stateful switchover are kept in synchronization. Consequently, it offers minimum interruption to Layer 2 sessions in a redundant active switch configuration.

If the active switch fails, the standby switch becomes the active switch. This new active switch uses existing Layer 2 switching information to continue forwarding traffic. Layer 3 forwarding will be delayed until the routing tables have been repopulated in the newly active switch.

Note SSO is not supported if the IOS-XE software is running the LAN Base license level.

The state of these features is preserved between both the active and standby switches:

- 802.3
- 802.3u
- 802.3x (Flow Control)
- 802.3ab (GE)
- 802.3z (Gigabit Ethernet including CWDM)
- 802.3ad (LACP)
- 802.1p (Layer 2 QoS)
- 802.1q
- 802.1X (Authentication)
- 802.1D (Spanning Tree Protocol)
- 802.3af (Inline power)
- PAgP
- VTP
- Dynamic ARP Inspection
- DHCP snooping
- IP source guard
- IGMP snooping (versions 1 and 2)
- DTP (802.1q and ISL)
- MST
- PVST+
- Rapid-PVST
- PortFast/UplinkFast/BackboneFast
- BPDU guard and filtering

- Voice VLAN
- Port security
- Unicast MAC filtering
- ACL (VACLS, PACLS, RACLS)
- QOS (DBL)
- Multicast storm control/broadcast storm control

SSO is compatible with the following list of features. However, the protocol database for these features is not synchronized between the standby and active switches:

- 802.1Q tunneling with Layer 2 Protocol Tunneling (L2PT)
- Baby giants
- Jumbo frame support
- CDP
- Flood blocking
- UDLD
- SPAN/RSPAN
- NetFlow

All Layer 3 protocols on a switch are learned on the standby switch if SSO is enabled.

NSF Operation

Cisco IOS Nonstop Forwarding (NSF) always runs with stateful switchover (SSO) and provides redundancy for Layer 3 traffic. NSF is supported by the BGP, OSPF, and EIGRP routing protocols and is supported by Cisco Express Forwarding (CEF) for forwarding. The routing protocols have been enhanced with NSF-capability and awareness, which means that routers running these protocols can detect a switchover and take the necessary actions to continue forwarding network traffic and to recover route information from the peer devices.

Each protocol depends on CEF to continue forwarding packets during switchover while the routing protocols rebuild the Routing Information Base (RIB) tables. After the routing protocols have converged, CEF updates the FIB table and removes stale route entries. CEF then updates the hardware with the new FIB information.

If the active switch is configured for BGP (with the **graceful-restart** command), OSPF, or EIGRP routing protocols, routing updates are automatically sent during the active switch election.

The switch supports NSF-awareness and NSF-capability for the BGP, OSPF, and EIGRP protocols in IP Services license level and NSF-awareness for the EIGRP-stub in IP Base license level.

NSF has two primary components:

- NSF-awareness

  A networking device is NSF-aware if it is running NSF-compatible software. If neighboring router devices detect that an NSF router can still forward packets when an active switch election happens, this capability is referred to as NSF-awareness. Cisco IOS enhancements to the Layer 3 routing protocols (BGP, OSPF, and EIGRP) are designed to prevent route-flapping so that the CEF routing table does not time out or the NSF router does not drop routes. An NSF-aware router helps to send routing protocol information to the neighboring NSF router. NSF-awareness is enabled by default for EIGRP-stub, EIGRP, and OSPF protocols. NSF-awareness is disabled by default for BGP.

- NSF-capability

  A device is NSF-capable if it has been configured to support NSF; it rebuilds routing information from NSF-aware or NSF-capable neighbors. NSF works with SSO to minimize the amount of time that a Layer 3 network is unavailable following an active switch election by continuing to forward IP packets. Reconvergence of Layer 3 routing protocols (BGP, OSPFv2, and EIGRP) is transparent to the user and happens automatically in the background. The routing protocols recover routing information from neighbor devices and rebuild the Cisco Express Forwarding (CEF) table.

Note NSF does not support IPv6 and is IPv4 Unicast only.

Cisco Express Forwarding

A key element of Cisco IOS Nonstop Forwarding (NSF) is packet forwarding. In a Cisco networking device, packet forwarding is provided by Cisco Express Forwarding (CEF). CEF maintains the FIB and uses the FIB information that was current at the time of the switchover to continue forwarding packets during a switchover. This feature reduces traffic interruption during the switchover.

During normal NSF operation, CEF on the active supervisor switch synchronizes its current FIB and adjacency databases with the FIB and adjacency databases on the standby switch. Upon switchover, the standby switch initially has FIB and adjacency databases that are mirror images of those that were current on the active switch. CEF keeps the forwarding engine on the standby switch current with changes that are sent to it by CEF on the active switch. The forwarding engine can continue forwarding after a switchover as soon as the interfaces and a data path are available.

As the routing protocols start to repopulate the RIB on a prefix-by-prefix basis, the updates cause prefix-by-prefix updates to CEF, which it uses to update the FIB and adjacency databases. Existing and new entries receive the new version ("epoch") number, indicating that they have been refreshed. The forwarding information is updated on the forwarding engine during convergence. The switch signals when the RIB has converged. The software removes all FIB and adjacency entries that have an epoch older than the current switchover epoch. The FIB now represents the newest routing protocol forwarding information.

BGP Operation

When an NSF-capable router begins a BGP session with a BGP peer, it sends an OPEN message to the peer. Included in the message is a statement that the NSF-capable device has "graceful" restart capability. Graceful restart is the mechanism by which BGP routing peers avoid a routing flap following a switchover. If the BGP peer has received this capability, it is aware that the device sending the message is NSF-capable. Both the NSF-capable router and its BGP peers need to exchange the graceful restart capability in their OPEN messages at the time of session establishment. If both the peers do not exchange the graceful restart capability, the session will not be capable of a graceful restart.

If the BGP session is lost during the active switch switchover, the NSF-aware BGP peer marks all the routes associated with the NSF-capable router as stale; however, it continues to use these routes to make forwarding decisions for a set period of time. This functionality prevents packets from being lost while the newly active switch is waiting for convergence of the routing information with the BGP peers.

After an active switch switchover occurs, the NSF-capable router reestablishes the session with the BGP peer. In establishing the new session, it sends a new graceful restart message that identifies the NSF-capable router as having restarted.

At this point, the routing information is exchanged between the two BGP peers. After this exchange is complete, the NSF-capable device uses the routing information to update the RIB and the FIB with the new forwarding information. The NSF-aware device uses the network information to remove stale routes from its BGP table; the BGP protocol then is fully converged.

If a BGP peer does not support the graceful restart capability, it ignores the graceful restart capability in an OPEN message but establishes a BGP session with the NSF-capable device. This function allows interoperability with non-NSF-aware BGP peers (and without NSF functionality), but the BGP session with non-NSF-aware BGP peers is not capable of a graceful restart.

Note BGP support in NSF requires that neighbor networking devices be NSF-aware; that is, the devices must have the graceful restart capability and advertise that capability in their OPEN message during session establishment. If an NSF-capable router discovers that a particular BGP neighbor does not have graceful restart capability, it does not establish an NSF-capable session with that neighbor. All other neighbors that have graceful restart capability continue to have NSF-capable sessions with this NSF-capable networking device.

OSPF Operation

When an OSPF NSF-capable router performs an active switch switchover, it must perform the following tasks in order to resynchronize its link state database with its OSPF neighbors:

- Relearn the available OSPF neighbors on the network without causing a reset of the neighbor relationship
- Reacquire the contents of the link state database for the network

As quickly as possible after an active switch switchover, the NSF-capable router sends an OSPF NSF signal to neighboring NSF-aware devices. Neighbor networking devices recognize this signal as an indicator that the neighbor relationship with this router should not be reset. As the NSF-capable router receives signals from other routers on the network, it can begin to rebuild its neighbor list.

After neighbor relationships are reestablished, the NSF-capable router begins to resynchronize its database with all of its NSF-aware neighbors. At this point, the routing information is exchanged between the OSPF neighbors. Once this exchange is complete, the NSF-capable device uses the routing information to remove stale routes, update the RIB, and update the FIB with the new forwarding information. The OSPF protocols are then fully converged.

Note OSPF support in NSF requires that all neighbor networking devices be NSF-aware. If an NSF-capable router discovers that it has non-NSF -aware neighbors on a particular network segment, it disables NSF capabilities for that segment. Other network segments composed entirely of NSF-capable or NSF-aware routers continue to provide NSF capabilities.

EIGRP Operation

When an EIGRP NSF-capable router initially re-boots after an NSF restart, it has no neighbor and its topology table is empty. The router is notified by the standby (now active) switch when it needs to bring up the interfaces, reacquire neighbors, and rebuild the topology and routing tables. The restarting router and its peers must accomplish these tasks without interrupting the data traffic directed toward the restarting router. EIGRP peer routers maintain the routes learned from the restarting router and continue forwarding traffic through the NSF restart process.

To prevent an adjacency reset by the neighbors, the restarting router uses a new Restart (RS) bit in the EIGRP packet header to indicate a restart. The RS bit is set in the hello packets and in the initial INIT update packets during the NSF restart period. The RS bit in the hello packets allows the neighbors to be quickly notified of the NSF restart. Without seeing the RS bit, the neighbor can only detect an adjacency reset by receiving an INIT update or by the expiration of the hello hold timer. Without the RS bit, a neighbor does not know if the adjacency reset should be handled using NSF or the normal startup method.

When the neighbor receives the restart indication, either by receiving the hello packet or the INIT packet, it recognizes the restarting peer in its peer list and maintains the adjacency with the restarting router. The neighbor then sends it topology table to the restarting router with the RS bit set in the first update packet indicating that it is NSF-aware and is helping out the restarting router. The neighbor does not set the RS bit in their hello packets, unless it is also a NSF restarting neighbor.

Note A router may be NSF-aware but may not be helping the NSF restarting neighbor because booting from a cold start.

If at least one of the peer routers is NSF-aware, the restarting router would then receive updates and rebuild its database. The restarting router must then find out if it had converged so that it can notify the routing information base (RIB). Each NSF-aware router is required to send an end of table (EOT) marker in the last update packet to indicate the end of the table content. The restarting router knows it has converged when it receives the EOT marker. The restarting router can then begin sending updates.

An NSF-aware peer would know when the restarting router had converged when it receives an EOT indication from the restarting router. The peer then scans its topology table to search for the routes with the restarted neighbor as the source. The peer compares the route timestamp with the restart event timestamp to determine if the route is still available. The peer then goes active to find alternate paths for the routes that are no longer available through the restarted router.

When the restarting router has received all EOT indications from its neighbors or when the NSF converge timer expires, EIGRP notifies the RIB of convergence. EIGRP waits for the RIB convergence signal and then floods its topology table to all awaiting NSF-aware peers.

How to Configure Cisco NSF with SSO

Configuring SSO

You must configure SSO in order to use NSF with any supported protocol.

SUMMARY STEPS

1. **redundancy**
2. **mode sso**
3. **end**
4. **show running-config**
5. **show redundancy states**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**redundancy** **Example:** `Switch(config)# redundancy`	Enters redundancy configuration mode.
Step 2	**mode sso** **Example:** `Switch(config-red)# mode sso`	Configures SSO. When this command is entered, the standby switch is reloaded and begins to work in SSO mode.
Step 3	**end** **Example:** `Switch(config-red)# end`	Returns to EXEC mode.
Step 4	**show running-config** **Example:** `Switch# show running-config`	Verifies that SSO is enabled.
Step 5	**show redundancy states** **Example:** `Switch# show redundancy states`	Displays the operating redundancy mode.

Configuring SSO Example

This example shows how to configure the system for SSO and display the redundancy state:

```
Switch(config)# redundancy
  Switch(config)# mode sso
  Switch(config)# end
  Switch# show redundancy states
  my state = 13 -ACTIVE
  peer state = 8 -STANDBY HOT
  Mode = Duplex
  Unit = Primary
  Unit ID = 5
  Redundancy Mode (Operational) = sso
  Redundancy Mode (Configured) = sso
  Split Mode = Disabled
```

```
Manual Swact = Enabled
Communications = Up
client count = 29
client_notification_TMR = 30000 milliseconds
keep_alive TMR = 9000 milliseconds
keep_alive count = 1
keep_alive threshold = 18
RF debug mask = 0x0
```

Configuring CEF NSF

The CEF NSF feature operates by default while the networking device is running in SSO mode. No configuration is necessary.

Verifying CEF NSF

To verify CEF NSF, use the **show cef state** privileged EXEX command.

```
Switch# show cef state
CEF Status:
RP instance
common CEF enabled
IPv4 CEF Status:
CEF enabled/running
dCEF enabled/running
CEF switching enabled/running
universal per-destination load sharing algorithm, id DEA83012
IPv6 CEF Status:
CEF disabled/not running
dCEF disabled/not running
universal per-destination load sharing algorithm, id DEA83012
RRP state:
I am standby RRP: no
RF Peer Presence: yes
RF PeerComm reached: yes
RF Progression blocked: never
Redundancy mode: rpr(1)
CEF NSF sync: disabled/not running
CEF ISSU Status:
FIBHWIDB broker
No slots are ISSU capable.
FIBIDB broker
No slots are ISSU capable.
FIBHWIDB Subblock broker
No slots are ISSU capable.
FIBIDB Subblock broker
No slots are ISSU capable.
Adjacency update
No slots are ISSU capable.
IPv4 table broker
No slots are ISSU capable.
CEF push
No slots are ISSU capable.
```

Configuring BGP for NSF

You must configure BGP graceful restart on all peer devices participating in BGP NSF.

SUMMARY STEPS

1. **configure terminal**
2. **router bgp as-number**
3. **bgp graceful-restart**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch(config)# configure terminal`	Enters global configuration mode.
Step 2	**router bgp as-number** **Example:** `Switch(config)# router bgp 300`	Enables a BGP routing process, which places the switch in switch configuration mode.
Step 3	**bgp graceful-restart** **Example:** `Switch(config)# bgp graceful-restart`	Enables the BGP graceful restart capability, starting BGP NSF. If you enter this command after the BGP session has been established, you must restart the session for the capability to be exchanged with the BGP neighbor. Use this command on the restarting switch and all of its peers.

Verifying BGP NSF

To verify BGP NSF, you must check that BGP graceful restart is configured on the SSO-enabled networking device and on the neighbor devices. To verify, follow these steps:

Step 1 Verify that “bgp graceful-restart” appears in the BGP configuration of the SSO-enabled switch by entering the **show running-config** command:

Example:

```
Switch# show running-config
.
.
.
router bgp 120
.
.
.
bgp graceful-restart
neighbor 192.0.2.0 remote-as 300
.
```

```
.
.
```

Step 2 Repeat Step 1 on each of the BGP neighbors.

Step 3 On the SSO device and the neighbor device, verify that the graceful restart function is shown as both advertised and received, and confirm the address families that have the graceful restart capability. If no address families are listed, BGP NSF does not occur either:

Example:

```
Switch# show ip bgp neighbors
BGP neighbor is 192.0.2.3, remote AS 1, internal link
BGP version 4, remote router ID 192.0.2.4
BGP state = Established, up for 00:02:38
Last read 00:00:38, last write 00:00:35, hold time is 180, keepalive interval is 60
seconds
Neighbor capabilities:
Route refresh: advertised and received(new)
Address family IPv4 Unicast: advertised and received
Message statistics:
InQ depth is 0
OutQ depth is 0
Sent Rcvd
Opens: 1 1
Notifications: 0 0
Updates: 0 0
Keepalives: 4 4
Route Refresh: 0 0
Total: 5 5
Default minimum time between advertisement runs is 0 seconds
.............................................................
(Remaining output deleted)
```

Configuring OSPF NSF

All peer devices participating in OSPF NSF must be made OSPF NSF-aware, which happens automatically when you install an NSF software image on the device.

SUMMARY STEPS

1. **configure terminal**
2. **router ospf** *processID*
3. **nsf**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch(config)# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**router ospf** *processID* **Example:** `Switch(config)# router ospf processID`	Enables an OSPF routing process, which places the switch in router configuration mode.
Step 3	**nsf** **Example:** `Switch(config)# nsf`	Enables NSF operations for OSPF.

Verifying OSPF NSF

Step 1 Verify that 'nsf' appears in the OSPF configuration of the SSO-enabled device by entering the show running-config command:

Example:

```
Switch(config)#show running-config
route ospf 120
log-adjacency-changes
nsf
network 192.0.2.0 192.0.2.255 area 0
network 192.0.2.1 192.0.2.255 area 1
network 192.0.2.2 192.0.2.255 area 2
.
.
.
```

Step 2 Enter the **show ip ospf** command to verify that NSF is enabled on the device:

Example:

```
Switch show ip ospf
Routing Process "ospf 1" with ID 192.0.2.1
Start time: 00:02:07.532, Time elapsed: 00:39:05.052
Supports only single TOS(TOS0) routes
Supports opaque LSA
Supports Link-local Signaling (LLS)
transit capable is 0
External flood list length 0
IETF Non-Stop Forwarding enabled
restart-interval limit: 120 sec
IETF NSF helper support enabled
Cisco NSF helper support enabled
Reference bandwidth unit is 100 mbps
Area BACKBONE(0)
Number of interfaces in this area is 3 (1 loopback)
Area has no authentication
SPF algorithm last executed 00:08:53.760 ago
SPF algorithm executed 2 times
Area ranges are
Number of LSA 3. Checksum Sum 0x025BE0
Number of opaque link LSA 0. Checksum Sum 0x000000
Number of DCbitless LSA 0
Number of indication LSA 0
```

```
Number of DoNotAge LSA 0
Flood list length 0
```

Configuring EIGRP NSF

SUMMARY STEPS

1. **configure terminal**
2. **router eigrp** *as-number*
3. **nsf**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch configure terminal`	Enters global configuration mode.
Step 2	**router eigrp** *as-number* **Example:** `Switch(config)# router eigrp  as-number`	Enables an EIGRP routing process, which places the switch in router configuration mode.
Step 3	**nsf** **Example:** `Switch(config-router)# nsf`	Enables EIGRP NSF. Use this command on the “restarting” switch and all of its peers.

Verifying EIGRP NSF

Step 1 Verify that “nsf” appears in the EIGRP configuration of the SSO-enabled device by entering the show **running-config command**:

Example:

```
Switch show running-config
..
.
router eigrp 100
auto-summary
nsf
..
```

.

Step 2 Enter the **show ip protocols** command to verify that NSF is enabled on the device:

Example:

```
Switch show ip protocols
*** IP Routing is NSF aware ***
Routing Protocol is "ospf 1"
Outgoing update filter list for all interfaces is not set
Incoming update filter list for all interfaces is not set
Router ID 192.0.2.3
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
Maximum path: 1
Routing for Networks:
Routing on Interfaces Configured Explicitly (Area 0):
Loopback0
GigabitEthernet5/3
TenGigabitEthernet3/1
Routing Information Sources:
Gateway Distance Last Update
192.0.2.1 110 00:01:02
Distance: (default is 110)
Routing Protocol is "bgp 601"
Outgoing update filter list for all interfaces is not set
Incoming update filter list for all interfaces is not set
IGP synchronization is disabled
Automatic route summarization is disabled
Neighbor(s):
Address FiltIn FiltOut DistIn DistOut Weight RouteMap
192.0.2.0
Maximum path: 1
Routing Information Sources:
Gateway Distance Last Update
192.0.2.0 20 00:01:03
Distance: external 20 internal 200 local 200
```

Additional References for NSF with SSO

Related Documents

Related Topic	Document Title
IP Routing: BGP	IP Routing: BGP Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)
IP Routing: EIGRP	IP Routing: EIGRP Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)
IP Routing: OSPF	IP Routing: OSPF Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and licensed feature sets,, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for NSF with SSO

Release	Modification

Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 76

High Availability

This chapter contains the following sections:

Information about High Availability

The high availability feature is enabled by default when the switches are connected using the stack cable and the technology is enabled. You cannot disable it; however, you can initiate a manual graceful-switchover using the command line interface to use the high availability feature enabled in the switch.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for Switchover

This section lists the restrictions that you should keep in mind while performing a switchover from the active unit to the standby unit:

- Verify that the unit has been reloaded using the stack manager.
- Verify if the logging console is enabled.
- Verify the progression failures. The progression is bound by a time unit of 30 seconds.
- Verify if no process failures occur.
- Verify if any congestion occurs in the up coming unit.

Post Switchover Tasks

- This section defines the steps that you must perform to ensure that successful switchover from the active to standby switch is performed. On successful switchover of the standby switch as active, all access points connected to the active need to re-join the standby (then active) switch.

SUMMARY STEPS

1. **show ap uptime**
2. **show wireless summary**
3. **show wcdb database all**
4. **show power inline**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show ap uptime**	Verify the uptime of the access point uptime after the switchover is large enough
Step 2	**show wireless summary**	Display the clients connected in the active switch.
Step 3	**show wcdb database all**	Display if the client has reached the uptime.
Step 4	**show power inline**	Display the power over Ethernet power state.

Information on Mobility

On switchover, the mobility states will be removed from the mobility domain. Once the switchover is complete, the MC and MA messages are sent to the peer nodes. These messages aid in identifying the status of the peers in the network. Each switch is part of the stack, and the MA and MC software is available on both the active and standby switches. The mobility control session up and down is identified using the keep alive messages. The details of the mobility client database data synchronization between the active and standby is performed. For more details on mobility, see the Cisco

You must remove the following when client SSO is supported as the switch-over event in this release is AP SSO:

- The switch-over notification from mobility controller to mobility oracle and to all the mobility anchors in the same group.
- Mobility anchors in the sub-domain that deletes all the client.
- The mobility oracle deletes clients with switchover mobility controller as foreign or anchor.
- The other mobility agents of the group cleans up the clients that are associated to the switchover mobility agent.
- Delete client messages to the mobility oracle.

Debugging Mobility before the Switchover

This section provides steps on how you can debug mobility before the switch-over.

SUMMARY STEPS

1. **show etherchannel summary**
2. **show platform etherchannel**
3. **show wireless mobility summary**
4. **show wireless mobility dtls connections**
5. **show mgmt-infra trace messages devshell**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show etherchannel summary**	Display port channel summary.
Step 2	**show platform etherchannel**	Display the states of the platform port channel.
Step 3	**show wireless mobility summary**	Display the mobility link state.
Step 4	**show wireless mobility dtls connections**	Display the mobility DTLS connection table.
Step 5	**show mgmt-infra trace messages devshell**	Display the mobility DTLS lookup table.

Debugging Mobility after the Switchover

This section provides steps on how you can debug mobility after the switchover. You must perform the same steps displayed in the debug mobility before the switchover to debug the mobility after switchover.

Information About Radio Resource Management

This section provides the theory of operations for the Radio Resource Management (RRM) feature in this release. RRM allows Cisco's Unified WLAN Architecture to continuously analyze the existing RF environment, automatically adjusting APs' power levels and channel configurations to help mitigate such things as co-channel interference and signal coverage problems. RRM reduces the need to perform exhaustive site surveys, increases system capacity, and provides automated self-healing functionality to compensate for RF dead zones and AP failures.

During a switchover, the AP's Radio Slot Data, RF measurement, CleanAir, and RF grouping state is synchronized to the standby switch during AP SSO. The standby switch then does not participate in any RF grouping unless it is in active state. The standby unit participates in the RF group as a new member. During a switchover, incremental data synchronization of data from active switch to standby switch is performed.

Information on Security

Wireless security is the prevention of unauthorized access or damage to computers using wireless networks. In Cisco Wireless Controller 5700 series, you can:

- Configure IEEE 802.1x global configurations.
- Configure Locally Significant Certificates.
- Configure strong password enforcement options.
- Enable over-the-air frame padding.
- Configure access point neighbor authentication.
- Enable protection from Denial of Service (DoS) attacks.
- Configure Intrusion Detection System (IDS) sensors for the Wireless Protection System (WPS).
- Configure client exclusion policies.
- Configure Management Frame Protection (MFP).
- Enforce the controller to synchronize with other controllers in the mobility group for the shun list.

On switchover, the following security features continue to be up and running:

- Datagram Transport Layer Security (DTLS) protocol for CAPWAP and mobility.
- Infrastructure Management Frame.
- Rogue access points and client detection.

- The access point list in the controller continues to work by re-learning.
- Synchronization of the Intrusion Detection System (IDS)/ Shun List in the mobility anchor and mobility client.
- The Wireless Intrusion Prevention System (wIPS) continues to work by re-learning and configuration methods.
- Synchronization of pairwise master key to enable faster re-association of the wireless clients.

Information on Location and Certificate Management

The states of the TCP port (16113) that the controller and mobility services engine communicate are not synchronized. So, once a switchover occurs, the client must re-associate to the location server. You must also manually download the LSC certificate and install in both the units.

•

Information on CAPWAP, Multicast, and CDP

In a CAPWAP environment, a lightweight access point discovers a controller by using CAPWAP discovery mechanisms and then sends the controller a CAPWAP join request. The controller sends the access point a CAPWAP join response allowing the access point to join the controller. When the access point joins the controller, the controller manages its configuration, firmware, control transactions, and data transactions. On switch-over, all the clients associated need to de-associate. The following occurs, after a switchover:

- The CAPWAP control message sequence number continues to be functional.
- The multicast protocol functions by re-learning the routes.
- The CDP protocol discovers all network elements in the network, and the access point sends the CDP update.
- Event log of the access point will not be synchronized during the switchover.

Information on Voice and QoS

QoS enables you to provide preferential treatment to specific types of traffic at the expense of other traffic types. Without QoS, the controller offers best-effort service to each packet, regardless of the packet contents r size. On switchover, the voice and QoS statistics values displayed in the controller will be reset. This implies that the Voice and QoS statistics are not synchronized while switchover happens from active unit to standby unit.

PART XII

Network Management

CHAPTER 77

Configuring Cisco IOS Configuration Engine

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for Configuring the Configuration Engine

- Obtain the name of the configuration engine instance to which you are connecting.
- Because the CNS uses both the event bus and the configuration server to provide configurations to devices, you must define both ConfigID and Device ID for each configured switch.
- All switches configured with the **cns config partial** global configuration command must access the event bus. The DeviceID, as originated on the switch, must match the DeviceID of the corresponding

switch definition in the Cisco Configuration Engine. You must know the hostname of the event bus to which you are connecting.

Related Topics

Cisco Networking Services IDs and Device Hostnames, on page 1630
DeviceID, on page 1631

Restrictions for Configuring the Configuration Engine

- Within the scope of a single instance of the configuration server, no two configured switches can share the same value for ConfigID.
- Within the scope of a single instance of the event bus, no two configured switches can share the same value for DeviceID.

Related Topics

Cisco Networking Services IDs and Device Hostnames, on page 1630

Information About Configuring the Configuration Engine

Cisco Configuration Engine Software

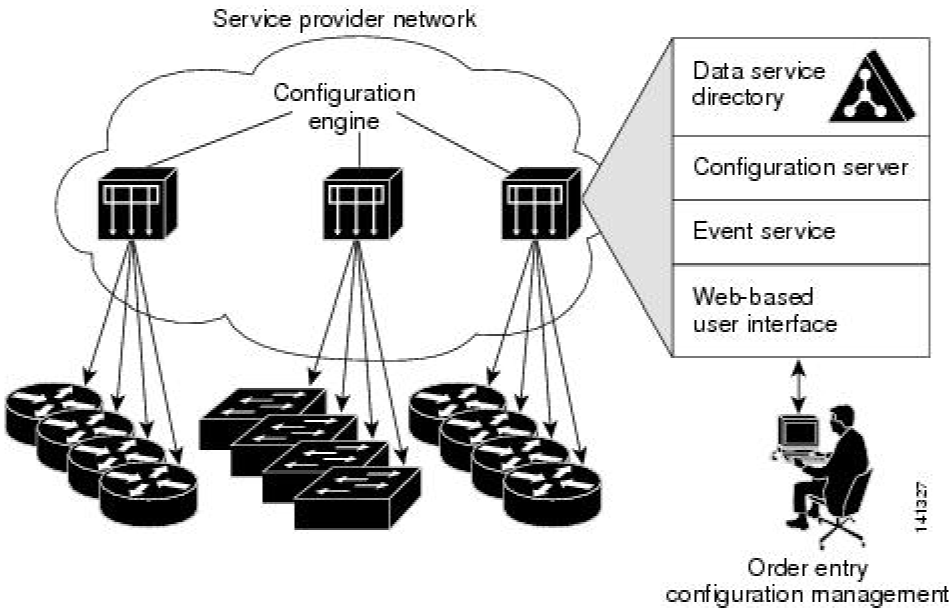
The Cisco Configuration Engine is network management utility software that acts as a configuration service for automating the deployment and management of network devices and services. Each Cisco Configuration Engine manages a group of Cisco devices (switches and routers) and the services that they deliver, storing their configurations and delivering them as needed. The Cisco Configuration Engine automates initial configurations and configuration updates by generating device-specific configuration changes, sending them to the device, executing the configuration change, and logging the results.

The Cisco Configuration Engine supports standalone and server modes and has these Cisco Networking Services (CNS) components:

- Configuration service:
  - Web server
  - File manager
  - Namespace mapping server
- Event service (event gateway)
- Data service directory (data models and schema)

In standalone mode, the Cisco Configuration Engine supports an embedded directory service. In this mode, no external directory or other data store is required. In server mode, the Cisco Configuration Engine supports the use of a user-defined external directory.

Figure 72: Cisco Configuration Engine Architectural Overview



Configuration Service

The Configuration Service is the core component of the Cisco Configuration Engine. It consists of a Configuration Server that works with Cisco IOS CNS agents on the switch. The Configuration Service delivers device and service configurations to the switch for initial configuration and mass reconfiguration by logical groups. Switches receive their initial configuration from the Configuration Service when they start up on the network for the first time.

The Configuration Service uses the CNS Event Service to send and receive configuration change events and to send success and failure notifications.

The Configuration Server is a web server that uses configuration templates and the device-specific configuration information stored in the embedded (standalone mode) or remote (server mode) directory.

Configuration templates are text files containing static configuration information in the form of CLI commands. In the templates, variables are specified by using Lightweight Directory Access Protocol (LDAP) URLs that reference the device-specific configuration information stored in a directory.

The Cisco IOS agent can perform a syntax check on received configuration files and publish events to show the success or failure of the syntax check. The configuration agent can either apply configurations immediately or delay the application until receipt of a synchronization event from the configuration server.

Event Service

The Cisco Configuration Engine uses the Event Service for receipt and generation of configuration events. The Event Service consists of an event agent and an event gateway. The event agent is on the switch and facilitates the communication between the switch and the event gateway on the Cisco Configuration Engine.

The Event Service is a highly capable publish-and-subscribe communication method. The Event Service uses subject-based addressing to send messages to their destinations. Subject-based addressing conventions define a simple, uniform namespace for messages and their destinations.

Related Topics

Enabling the CNS Event Agent, on page 1634

NameSpace Mapper

The Cisco Configuration Engine includes the NameSpace Mapper (NSM) that provides a lookup service for managing logical groups of devices based on application, device or group ID, and event.

Cisco IOS devices recognize only event subject-names that match those configured in Cisco IOS software; for example, cisco.cns.config.load. You can use the namespace mapping service to designate events by using any desired naming convention. When you have populated your data store with your subject names, NSM changes your event subject-name strings to those known by Cisco IOS.

For a subscriber, when given a unique device ID and event, the namespace mapping service returns a set of events to which to subscribe. Similarly, for a publisher, when given a unique group ID, device ID, and event, the mapping service returns a set of events on which to publish.

Cisco Networking Services IDs and Device Hostnames

The Cisco Configuration Engine assumes that a unique identifier is associated with each configured switch. This unique identifier can take on multiple synonyms, where each synonym is unique within a particular namespace. The event service uses namespace content for subject-based addressing of messages.

The Cisco Configuration Engine intersects two namespaces, one for the event bus and the other for the configuration server. Within the scope of the configuration server namespace, the term *ConfigID* is the unique identifier for a device. Within the scope of the event bus namespace, the term *DeviceID* is the CNS unique identifier for a device.

Related Topics

Prerequisites for Configuring the Configuration Engine, on page 1627

Restrictions for Configuring the Configuration Engine, on page 1628

ConfigID

Each configured switch has a unique ConfigID, which serves as the key into the Cisco Configuration Engine directory for the corresponding set of switch CLI attributes. The ConfigID defined on the switch must match the ConfigID for the corresponding switch definition on the Cisco Configuration Engine.

The ConfigID is fixed at startup time and cannot be changed until the device restarts, even if the switch hostname is reconfigured.

DeviceID

Each configured switch participating on the event bus has a unique DeviceID, which is analogous to the switch source address so that the switch can be targeted as a specific destination on the bus.

The origin of the DeviceID is defined by the Cisco IOS hostname of the switch. However, the DeviceID variable and its usage reside within the event gateway adjacent to the switch.

The logical Cisco IOS termination point on the event bus is embedded in the event gateway, which in turn functions as a proxy on behalf of the switch. The event gateway represents the switch and its corresponding DeviceID to the event bus.

The switch declares its hostname to the event gateway immediately after the successful connection to the event gateway. The event gateway couples the DeviceID value to the Cisco IOS hostname each time this connection is established. The event gateway retains this DeviceID value for the duration of its connection to the switch.

Related Topics

Prerequisites for Configuring the Configuration Engine, on page 1627

Hostname and DeviceID

The DeviceID is fixed at the time of the connection to the event gateway and does not change even when the switch hostname is reconfigured.

When changing the switch hostname on the switch, the only way to refresh the DeviceID is to break the connection between the switch and the event gateway. For instructions on refreshing DeviceIDs, see "Related Topics."

When the connection is reestablished, the switch sends its modified hostname to the event gateway. The event gateway redefines the DeviceID to the new value.

Caution When using the Cisco Configuration Engine user interface, you must first set the DeviceID field to the hostname value that the switch acquires *after*, not *before*, and you must reinitialize the configuration for your Cisco IOS CNS agent. Otherwise, subsequent partial configuration command operations may malfunction.

Related Topics

Refreshing DeviceIDs, on page 1642

Hostname, DeviceID, and ConfigID

In standalone mode, when a hostname value is set for a switch, the configuration server uses the hostname as the DeviceID when an event is sent on hostname. If the hostname has not been set, the event is sent on the cn=<*value*> of the device.

In server mode, the hostname is not used. In this mode, the unique DeviceID attribute is always used for sending an event on the bus. If this attribute is not set, you cannot update the switch.

These and other associated attributes (tag value pairs) are set when you run **Setup** on the Cisco Configuration Engine.

Cisco IOS CNS Agents

The CNS event agent feature allows the switch to publish and subscribe to events on the event bus and works with the Cisco IOS CNS agent. These agents, embedded in the switch Cisco IOS software, allow the switch to be connected and automatically configured.

Initial Configuration

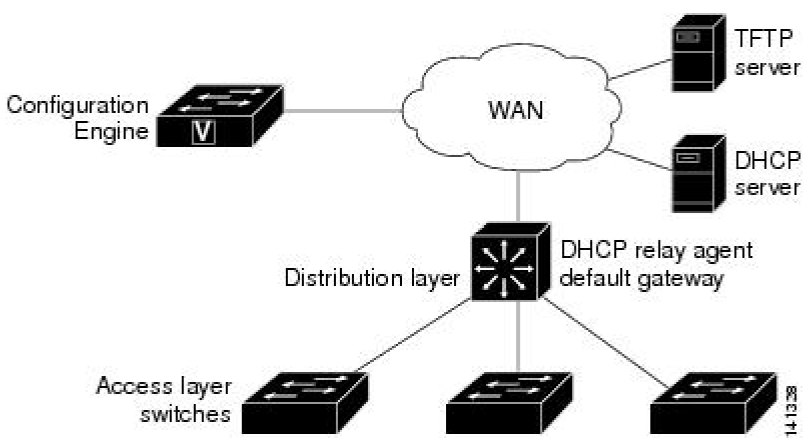
When the switch first comes up, it attempts to get an IP address by broadcasting a Dynamic Host Configuration Protocol (DHCP) request on the network. Assuming there is no DHCP server on the subnet, the distribution switch acts as a DHCP relay agent and forwards the request to the DHCP server. Upon receiving the request, the DHCP server assigns an IP address to the new switch and includes the Trivial File Transfer Protocol (TFTP) server Internet Protocol (IP) address, the path to the bootstrap configuration file, and the default gateway IP address in a unicast reply to the DHCP relay agent. The DHCP relay agent forwards the reply to the switch.

The switch automatically configures the assigned IP address on interface VLAN 1 (the default) and downloads the bootstrap configuration file from the TFTP server. Upon successful download of the bootstrap configuration file, the switch loads the file in its running configuration.

The Cisco IOS CNS agents initiate communication with the Configuration Engine by using the appropriate ConfigID and EventID. The Configuration Engine maps the Config ID to a template and downloads the full configuration file to the switch.

The following figure shows a sample network configuration for retrieving the initial bootstrap configuration file by using DHCP-based autoconfiguration.

Figure 73: Initial Configuration



Related Topics

Automated CNS Configuration, on page 1633

Incremental (Partial) Configuration

After the network is running, new services can be added by using the Cisco IOS CNS agent. Incremental (partial) configurations can be sent to the switch. The actual configuration can be sent as an event payload by way of the event gateway (push operation) or as a signal event that triggers the switch to initiate a pull operation.

The switch can check the syntax of the configuration before applying it. If the syntax is correct, the switch applies the incremental configuration and publishes an event that signals success to the configuration server. If the switch does not apply the incremental configuration, it publishes an event showing an error status. When the switch has applied the incremental configuration, it can write it to nonvolatile random-access memory (NVRAM) or wait until signaled to do so.

Synchronized Configuration

When the switch receives a configuration, it can defer application of the configuration upon receipt of a write-signal event. The write-signal event tells the switch not to save the updated configuration into its NVRAM. The switch uses the updated configuration as its running configuration. This ensures that the switch configuration is synchronized with other network activities before saving the configuration in NVRAM for use at the next reboot.

Automated CNS Configuration

To enable automated CNS configuration of the switch, you must first complete the prerequisites listed in this topic. When you complete them, power on the switch. At the **setup** prompt, do nothing; the switch begins the initial configuration. When the full configuration file is loaded on your switch, you do not need to do anything else.

For more information on what happens during initial configuration, see "Related Topics."

Table 167: Prerequisites for Enabling Automatic Configuration

Device	Required Configuration
Access switch	Factory default (no configuration file)
Distribution switch	• IP helper address • Enable DHCP relay agent[29] • IP routing (if used as default gateway)
DHCP server	• IP address assignment • TFTP server IP address • Path to bootstrap configuration file on the TFTP server • Default gateway IP address

Device	Required Configuration
TFTP server	• A bootstrap configuration file that includes the CNS configuration commands that enable the switch to communicate with the Configuration Engine • The switch configured to use either the switch MAC address or the serial number (instead of the default hostname) to generate the ConfigID and EventID • The CNS event agent configured to push the configuration file to the switch
CNS Configuration Engine	One or more templates for each type of device, with the ConfigID of the device mapped to the template.

29 A DHCP Relay is needed only when the DHCP Server is on a different subnet from the client.

Related Topics

Initial Configuration, on page 1632

How to Configure the Configuration Engine

Enabling the CNS Event Agent

> **Note** You must enable the CNS event agent on the switch before you enable the CNS configuration agent.

Beginning in privileged EXEC mode, follow these steps to enable the CNS event agent on the switch.

SUMMARY STEPS

1. **configure terminal**
2. **cns event** {*hostname* | *ip-address*} [*port-number*] [[**keepalive** *seconds retry-count*] [**failover-time** *seconds*] [**reconnect-time** *time*] | **backup**]
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**cns event** {*hostname* | *ip-address*} [*port-number*] [[**keepalive** *seconds* *retry-count*] [**failover-time** *seconds*] [**reconnect-time** *time*] | **backup**] **Example:** `Switch(config)# cns event 10.180.1.27 keepalive 120 10`	Enables the event agent, and enters the gateway parameters. • For {*hostname* | *ip-address*}, enter either the hostname or the IP address of the event gateway. • (Optional) For *port number*, enter the port number for the event gateway. The default port number is 11011. • (Optional) For **keepalive** *seconds*, enter how often the switch sends keepalive messages. For *retry-count*, enter the number of unanswered keepalive messages that the switch sends before the connection is terminated. The default for each is 0. • (Optional) For **failover-time** *seconds*, enter how long the switch waits for the primary gateway route after the route to the backup gateway is established. • (Optional) For **reconnect-time** *time*, enter the maximum time interval that the switch waits before trying to reconnect to the event gateway. • (Optional) Enter **backup** to show that this is the backup gateway. (If omitted, this is the primary gateway.) **Note** Though visible in the command-line help string, the **encrypt** and the **clock-timeout** *time* keywords are not supported.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

This example shows how to enable the CNS event agent, set the IP address gateway to 10.180.1.27, set 120 seconds as the keepalive interval, and set 10 as the retry count.

```
Switch(config)# cns event 10.180.1.27 keepalive 120 10
```

What to Do Next

To verify information about the event agent, use the **show cns event connections** command in privileged EXEC mode.

To disable the CNS event agent, use the **no cns event** { *ip-address* | *hostname* } global configuration command.

Related Topics

Event Service, on page 1630

Enabling the Cisco IOS CNS Agent

Beginning in privileged EXEC mode, follow these steps to enable the Cisco IOS CNS agent on the switch.

Before You Begin

You must enable the CNS event agent on the switch before you enable this agent.

SUMMARY STEPS

1. **configure terminal**
2. **cns config initial** {*hostname* | *ip-address*} [*port-number*]
3. **cns config partial** {*hostname* | *ip-address*} [*port-number*]
4. **end**
5. Start the Cisco IOS CNS agent on the switch.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**cns config initial** {*hostname* | *ip-address*} [*port-number*] **Example:** `Switch(config)# cns config initial 10.180.1.27 10`	Enables the Cisco IOS CNS agent, and enters the configuration server parameters. • For {*hostname* | *ip-address*}, enter either the hostname or the IP address of the configuration server. • (Optional) For *port number*, enter the port number for the configuration server. This command enables the Cisco IOS CNS agent and initiates an initial configuration on the switch.
Step 3	**cns config partial** {*hostname* | *ip-address*} [*port-number*] **Example:** `Switch(config)# cns config partial 10.180.1.27 10`	Enables the Cisco IOS CNS agent, and enters the configuration server parameters. • For {*hostname* | *ip-address*}, enter either the hostname or the IP address of the configuration server. • (Optional) For *port number*, enter the port number for the configuration server.

	Command or Action	Purpose
		Enables the Cisco IOS CNS agent and initiates a partial configuration on the switch.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	Start the Cisco IOS CNS agent on the switch.	

What to Do Next

You can now use the Cisco Configuration Engine to remotely send incremental configurations to the switch.

Related Topics

Refreshing DeviceIDs, on page 1642

Enabling an Initial Configuration for Cisco IOS CNS Agent

Beginning in privileged EXEC mode, follow these steps to enable the CNS configuration agent and initiate an initial configuration on the switch.

SUMMARY STEPS

1. **configure terminal**
2. **cns template connect** *name*
3. **cli** *config-text*
4. Repeat Steps 2 to 3 to configure another CNS connect template.
5. **exit**
6. **cns connect** *name* [**retries** *number*] [**retry-interval** *seconds*] [**sleep** *seconds*] [**timeout** *seconds*]
7. **discover** {**controller** *controller-type* | **dlci** [**subinterface** *subinterface-number*] | **interface** [*interface-type*] | **line** *line-type*}
8. **template** *name* [... *name*]
9. Repeat Steps 7 to 8 to specify more interface parameters and CNS connect templates in the CNS connect profile.
10. **exit**
11. **hostname** *name*
12. **ip route** *network-number*
13. **cns id** *interface num* {**dns-reverse** | **ipaddress** | **mac-address**} [**event**] [**image**]
14. **cns id** {**hardware-serial** | **hostname** | **string** *string* | **udi**} [**event**] [**image**]
15. **cns config initial** {*hostname* | *ip-address*} [*port-number*] [**event**] [**no-persist**] [**page** *page*] [**source** *ip-address*] [**syntax-check**]
16. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**cns template connect** *name* **Example:** `Switch(config)# cns template connect template-dhcp`	Enters CNS template connect configuration mode, and specifies the name of the CNS connect template.
Step 3	**cli** *config-text* **Example:** `Switch(config-tmpl-conn)# cli ip address dhcp`	Enters a command line for the CNS connect template. Repeat this step for each command line in the template.
Step 4	Repeat Steps 2 to 3 to configure another CNS connect template.	

	Command or Action	Purpose
Step 5	**exit** **Example:** `Switch(config)# exit`	Returns to global configuration mode.
Step 6	**cns connect** *name* [**retries** *number*] [**retry-interval** *seconds*] [**sleep** *seconds*] [**timeout** *seconds*] **Example:** `Switch(config)# cns connect dhcp`	Enters CNS connect configuration mode, specifies the name of the CNS connect profile, and defines the profile parameters. The switch uses the CNS connect profile to connect to the Configuration Engine. • Enter the *name* of the CNS connect profile. • (Optional) For **retries** *number*, enter the number of connection retries. The range is 1 to 30. The default is 3. • (Optional) For **retry-interval** *seconds*, enter the interval between successive connection attempts to the Configuration Engine. The range is 1 to 40 seconds. The default is 10 seconds. • (Optional) For **sleep** *seconds*, enter the amount of time before which the first connection attempt occurs. The range is 0 to 250 seconds. The default is 0. • (Optional) For **timeout** *seconds*, enter the amount of time after which the connection attempts end. The range is 10 to 2000 seconds. The default is 120.
Step 7	**discover** {**controller** *controller-type* | **dlci** [**subinterface** *subinterface-number*] | **interface** [*interface-type*] | **line** *line-type*} **Example:** `Switch(config-cns-conn)# discover interface gigabitethernet`	Specifies the interface parameters in the CNS connect profile. • For **controller** *controller-type*, enter the controller type. • For **dlci**, enter the active data-link connection identifiers (DLCIs). (Optional) For **subinterface** *subinterface-number*, specify the point-to-point subinterface number that is used to search for active DLCIs. • For **interface** [*interface-type*], enter the type of interface. • For **line** *line-type*, enter the line type.
Step 8	**template** *name* [... *name*] **Example:** `Switch(config-cns-conn)# template template-dhcp`	Specifies the list of CNS connect templates in the CNS connect profile to be applied to the switch configuration. You can specify more than one template.
Step 9	Repeat Steps 7 to 8 to specify more interface parameters and CNS connect templates in the CNS connect profile.	

	Command or Action	Purpose
Step 10	**exit** **Example:** `Switch(config-cns-conn)# exit`	Returns to global configuration mode.
Step 11	**hostname** *name* **Example:** `Switch(config)# hostname device1`	Enters the hostname for the switch.
Step 12	**ip route** *network-number* **Example:** `RemoteSwitch(config)# ip route 172.28.129.22 255.255.255.255 11.11.11.1`	(Optional) Establishes a static route to the Configuration Engine whose IP address is *network-number*.
Step 13	**cns id** *interface num* {**dns-reverse** | **ipaddress** | **mac-address**} [**event**] [**image**] **Example:** `RemoteSwitch(config)# cns id GigabitEthernet1/0/1 ipaddress`	(Optional) Sets the unique EventID or ConfigID used by the Configuration Engine. If you enter this command, do not enter the **cns id** {**hardware-serial** | **hostname** | **string** *string* | **udi**} [**event**] [**image**] command. • For *interface num*, enter the type of interface. For example, ethernet, group-async, loopback, or virtual-template. This setting specifies from which interface the IP or MAC address should be retrieved to define the unique ID. • For {**dns-reverse** | **ipaddress** | **mac-address**}, enter **dns-reverse** to retrieve the hostname and assign it as the unique ID, enter **ipaddress** to use the IP address, or enter **mac-address** to use the MAC address as the unique ID. • (Optional) Enter **event** to set the ID to be the event-id value used to identify the switch. • (Optional) Enter **image** to set the ID to be the image-id value used to identify the switch. **Note** If both the **event** and **image** keywords are omitted, the image-id value is used to identify the switch.
Step 14	**cns id** {**hardware-serial** | **hostname** | **string** *string* | **udi**} [**event**] [**image**] **Example:** `RemoteSwitch(config)# cns id hostname`	(Optional) Sets the unique EventID or ConfigID used by the Configuration Engine. If you enter this command, do not enter the **cns id** *interface num* {**dns-reverse** | **ipaddress** | **mac-address**} [**event**] [**image**] command. • For { **hardware-serial** | **hostname** | **string** *string* | **udi** }, enter **hardware-serial** to set the switch serial number as the unique ID, enter **hostname** (the default) to select the switch hostname as the unique ID, enter an arbitrary text string for **string** *string* as the unique ID, or enter **udi** to set the unique device identifier (UDI) as the unique ID.

	Command or Action	Purpose
Step 15	**cns config initial** {*hostname* | *ip-address*} [*port-number*] [**event**] [**no-persist**] [**page** *page*] [**source** *ip-address*] [**syntax-check**] **Example:** `RemoteSwitch(config)# cns config initial 10.1.1.1 no-persist`	Enables the Cisco IOS agent, and initiates an initial configuration. • For {*hostname* | *ip-address*}, enter the hostname or the IP address of the configuration server. • (Optional) For *port-number*, enter the port number of the configuration server. The default port number is 80. • (Optional) Enable **event** for configuration success, failure, or warning messages when the configuration is finished. • (Optional) Enable **no-persist** to suppress the automatic writing to NVRAM of the configuration pulled as a result of entering the **cns config initial** global configuration command. If the **no-persist** keyword is not entered, using the **cns config initial** command causes the resultant configuration to be automatically written to NVRAM. • (Optional) For **page** *page*, enter the web page of the initial configuration. The default is /Config/config/asp. • (Optional) Enter **source** *ip-address* to use for source IP address. • (Optional) Enable **syntax-check** to check the syntax when this parameter is entered. **Note** Though visible in the command-line help string, the **encrypt**, **status** *url*, and **inventory** keywords are not supported.
Step 16	**end** **Example:** `RemoteSwitch(config)# end`	Returns to privileged EXEC mode.

This example shows how to configure an initial configuration on a remote switch when the switch configuration is unknown (the CNS Zero Touch feature).

```
Switch(config)# cns template connect template-dhcp
Switch(config-tmpl-conn)# cli ip address dhcp
Switch(config-tmpl-conn)# exit
Switch(config)# cns template connect ip-route
Switch(config-tmpl-conn)# cli ip route 0.0.0.0 0.0.0.0 ${next-hop}
Switch(config-tmpl-conn)# exit
Switch(config)# cns connect dhcp
Switch(config-cns-conn)# discover interface gigabitethernet
Switch(config-cns-conn)# template template-dhcp
Switch(config-cns-conn)# template ip-route
Switch(config-cns-conn)# exit
Switch(config)# hostname RemoteSwitch
RemoteSwitch(config)# cns config initial 10.1.1.1 no-persist
```

This example shows how to configure an initial configuration on a remote switch when the switch IP address is known. The Configuration Engine IP address is 172.28.129.22.

```
Switch(config)# cns template connect template-dhcp
Switch(config-tmpl-conn)# cli ip address dhcp
Switch(config-tmpl-conn)# exit
Switch(config)# cns template connect ip-route
Switch(config-tmpl-conn)# cli ip route 0.0.0.0 0.0.0.0 ${next-hop}
Switch(config-tmpl-conn)# exit
Switch(config)# cns connect dhcp
Switch(config-cns-conn)# discover interface gigabitethernet
Switch(config-cns-conn)# template template-dhcp
Switch(config-cns-conn)# template ip-route
Switch(config-cns-conn)# exit
Switch(config)# hostname RemoteSwitch
RemoteSwitch(config)# ip route 172.28.129.22 255.255.255.255 11.11.11.1
RemoteSwitch(config)# cns id ethernet 0 ipaddress
RemoteSwitch(config)# cns config initial 172.28.129.22 no-persist
```

What to Do Next

To verify information about the configuration agent, use the **show cns config connections** command in privileged EXEC mode.

To disable the CNS Cisco IOS agent, use the **no cns config initial** { *ip-address* | *hostname* } global configuration command.

Refreshing DeviceIDs

Beginning in privileged EXEC mode, follow these steps to refresh a DeviceID when changing the hostname on the switch.

SUMMARY STEPS

1. **show cns config connections**
2. Make sure that the CNS event agent is properly connected to the event gateway.
3. **show cns event connections**
4. Record from the output of Step 3 the information for the currently connected connection listed below. You will be using the IP address and port number in subsequent steps of these instructions.
5. **configure terminal**
6. **no cns event** *ip-address port-number*
7. **cns event** *ip-address port-number*
8. **end**
9. Make sure that you have reestablished the connection between the switch and the event connection by examining the output from **show cns event connections**.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show cns config connections** **Example:** `Switch# show cns config connections`	Displays whether the CNS event agent is connecting to the gateway, connected, or active, and the gateway used by the event agent, its IP address and port number.
Step 2	Make sure that the CNS event agent is properly connected to the event gateway.	Examine the output of **show cns config connections** for the following: • Connection is active. • Connection is using the currently configured switch hostname. The DeviceID will be refreshed to correspond to the new hostname configuration using these instructions.
Step 3	**show cns event connections** **Example:** `Switch# show cns event connections`	Displays the event connection information for your switch.
Step 4	Record from the output of Step 3 the information for the currently connected connection listed below. You will be using the IP address and port number in subsequent steps of these instructions.	
Step 5	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 6	**no cns event** *ip-address port-number* **Example:** `Switch(config)# no cns event 172.28.129.22 2012`	Specifies the IP address and port number that you recorded in Step 4 in this command. This command breaks the connection between the switch and the event gateway. It is necessary to first break, then reestablish, this connection to refresh the DeviceID.
Step 7	**cns event** *ip-address port-number* **Example:** `Switch(config)# cns event 172.28.129.22 2012`	Specifies the IP address and port number that you recorded in Step 4 in this command. This command reestablishes the connection between the switch and the event gateway.
Step 8	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 9	Make sure that you have reestablished the connection between the switch and the event connection by examining the output from **show cns event connections**.	

Related Topics

Enabling the Cisco IOS CNS Agent, on page 1636

Hostname and DeviceID, on page 1631

Enabling a Partial Configuration for Cisco IOS CNS Agent

Beginning in privileged EXEC mode, follow these steps to enable the Cisco IOS CNS agent and to initiate a partial configuration on the switch.

SUMMARY STEPS

1. **configure terminal**
2. **cns config partial** {*ip-address* | *hostname*} [*port-number*] [**source** *ip-address*]
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**cns config partial** {*ip-address* | *hostname*} [*port-number*] [**source** *ip-address*] **Example:** `Switch(config)# cns config partial 172.28.129.22 2013`	Enables the configuration agent, and initiates a partial configuration. • For {*ip-address* | *hostname*}, enter the IP address or the hostname of the configuration server. • (Optional) For *port-number*, enter the port number of the configuration server. The default port number is 80. • (Optional) Enter **source** *ip-address* to use for the source IP address. **Note** Though visible in the command-line help string, the **encrypt** keyword is not supported.

	Command or Action	Purpose
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

What to Do Next

To verify information about the configuration agent, use either the **show cns config stats** or the **show cns config outstanding** command in privileged EXEC mode.

To disable the Cisco IOS agent, use the **no cns config partial** { *ip-address* | *hostname* } global configuration command. To cancel a partial configuration, use the **cns config cancel** global configuration command.

Monitoring CNS Configurations

Table 168: CNS show Commands

Command	Purpose
show cns config connections `Switch# show cns config connections`	Displays the status of the CNS Cisco IOS CNS agent connections.
show cns config outstanding `Switch# show cns config outstanding`	Displays information about incremental (partial) CNS configurations that have started but are not yet completed.
show cns config stats `Switch# show cns config stats`	Displays statistics about the Cisco IOS CNS agent.
show cns event connections `Switch# show cns event connections`	Displays the status of the CNS event agent connections.
show cns event gateway `Switch# show cns event gateway`	Displays the event gateway information for your switch.
show cns event stats `Switch# show cns event stats`	Displays statistics about the CNS event agent.
show cns event subject `Switch# show cns event subject`	Displays a list of event agent subjects that are subscribed to by applications.

Additional References

Related Documents

Related Topic	Document Title
Configuration Engine Setup	*Cisco Configuration Engine Installation and Setup Guide, 1.5 for Linux* http://www.cisco.com/en/US/docs/net_mgmt/configuration_engine/1.5/installation_linux/guide/setup_1.html

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for the Configuration Engine

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 78

Configuring the Cisco Discovery Protocol

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Information About CDP

CDP Overview

CDP is a device discovery protocol that runs over Layer 2 (the data-link layer) on all Cisco-manufactured devices (routers, bridges, access servers, controllers, and switches) and allows network management applications to discover Cisco devices that are neighbors of already known devices. With CDP, network management applications can learn the device type and the Simple Network Management Protocol (SNMP) agent address of neighboring devices running lower-layer, transparent protocols. This feature enables applications to send SNMP queries to neighboring devices.

CDP runs on all media that support Subnetwork Access Protocol (SNAP). Because CDP runs over the data-link layer only, two systems that support different network-layer protocols can learn about each other.

Each CDP-configured device sends periodic messages to a multicast address, advertising at least one address at which it can receive SNMP messages. The advertisements also contain time-to-live, or holdtime information, which is the length of time a receiving device holds CDP information before discarding it. Each device also listens to the messages sent by other devices to learn about neighboring devices.

On the switch, CDP enables Network Assistant to display a graphical view of the network. The switch uses CDP to find cluster candidates and maintain information about cluster members and other devices up to three cluster-enabled devices away from the command switch by default.

CDP and Stacks

A switch stack appears as a single switch in the network. Therefore, CDP discovers the switch stack, not the individual stack members. The switch stack sends CDP messages to neighboring network devices when there are changes to the switch stack membership, such as stack members being added or removed.

Default CDP Configuration

This table shows the default CDP configuration.

Feature	Default Setting
CDP global state	Enabled
CDP interface state	Enabled
CDP timer (packet update frequency)	60 seconds
CDP holdtime (before discarding)	180 seconds
CDP Version-2 advertisements	Enabled

How to Configure CDP

Configuring CDP Characteristics

You can configure these CDP characteristics:

- Frequency of CDP updates
- Amount of time to hold the information before discarding it
- Whether or not to send Version-2 advertisements

Note Steps 2 through 4 are all optional and can be performed in any order.

Beginning in privileged EXEC mode, follow these steps to configure these characteristics.

SUMMARY STEPS

1. **configure terminal**
2. **cdp timer** *seconds*
3. **cdp holdtime** *seconds*
4. **cdp advertise-v2**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**cdp timer** *seconds* **Example:** `Switch(config)# cdp timer 20`	(Optional) Sets the transmission frequency of CDP updates in seconds. The range is 5 to 254; the default is 60 seconds.
Step 3	**cdp holdtime** *seconds* **Example:** `Switch(config)# cdp holdtime 60`	(Optional) Specifies the amount of time a receiving device should hold the information sent by your device before discarding it. The range is 10 to 255 seconds; the default is 180 seconds.
Step 4	**cdp advertise-v2** **Example:** `Switch(config)# cdp advertise-v2`	(Optional) Configures CDP to send Version-2 advertisements. This is the default state.
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Example

The following example shows how to configure CDP characteristics:

```
Switch# configure terminal
Switch(config)# cdp timer 50
Switch(config)# cdp holdtime 120
Switch(config)# cdp advertise-v2
Switch(config)# end
```

What to Do Next

Use the **no** form of the CDP commands to return to the default settings.

Related Topics

Monitoring and Maintaining CDP, on page 1656

Disabling CDP

CDP is enabled by default.

Note Switch clusters and other Cisco devices (such as Cisco IP Phones) regularly exchange CDP messages. Disabling CDP can interrupt cluster discovery and device connectivity.

Beginning in privileged EXEC mode, follow these steps to disable the CDP device discovery capability.

SUMMARY STEPS

1. **configure terminal**
2. **no cdp run**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**no cdp run** **Example:** `Switch(config)# no cdp run`	Disables CDP.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

What to Do Next

You must reenable CDP to use it.

Related Topics

Enabling CDP, on page 1653

Enabling CDP

CDP is enabled by default.

Note Switch clusters and other Cisco devices (such as Cisco IP Phones) regularly exchange CDP messages. Disabling CDP can interrupt cluster discovery and device connectivity.

Beginning in privileged EXEC mode, follow these steps to enable CDP when it has been disabled.

Before You Begin

CDP must be disabled, or it cannot be enabled.

SUMMARY STEPS

1. **configure terminal**
2. **cdp run**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**cdp run** **Example:** `Switch(config)# cdp run`	Enables CDP if it has been disabled.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Example

The following example shows how to enable CDP if it has been disabled:

```
Switch# configure terminal
Switch(config)# cdp run
Switch(config)# end
```

What to Do Next

Use the **show run all** command to show that CDP has been enabled. If you enter only **show run**, the enabling of CDP may not be displayed.

Related Topics

Disabling CDP, on page 1652

Disabling CDP on an Interface

CDP is enabled by default on all supported interfaces to send and to receive CDP information.

Note Switch clusters and other Cisco devices (such as Cisco IP Phones) regularly exchange CDP messages. Disabling CDP can interrupt cluster discovery and device connectivity.

Beginning in privileged EXEC mode, follow these steps to disable CDP on a port.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **no cdp enable**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Specifies the interface on which you are disabling CDP, and enters interface configuration mode.
Step 3	**no cdp enable** **Example:** `Switch(config-if)# no cdp enable`	Disables CDP on the interface specified in Step 2.

	Command or Action	Purpose
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Related Topics

Enabling CDP on an Interface, on page 1655

Enabling CDP on an Interface

CDP is enabled by default on all supported interfaces to send and to receive CDP information.

> **Note** Switch clusters and other Cisco devices (such as Cisco IP Phones) regularly exchange CDP messages. Disabling CDP can interrupt cluster discovery and device connectivity.

Beginning in privileged EXEC mode, follow these steps to enable CDP on a port on which it has been disabled.

Before You Begin

CDP must be disabled on the port that you are trying to CDP enable on, or it cannot be enabled.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **cdp enable**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Specifies the interface on which you are enabling CDP, and enters interface configuration mode.

	Command or Action	Purpose
Step 3	**cdp enable** **Example:** `Switch(config-if)# cdp enable`	Enables CDP on a disabled interface.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Example

The following example shows how to enable CDP on a disabled port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# cdp enable
Switch(config-if)# end
```

Related Topics

Disabling CDP on an Interface, on page 1654

Monitoring and Maintaining CDP

Table 169: Commands for Displaying CDP Information

Command	Description
clear cdp counters	Resets the traffic counters to zero.
clear cdp table	Deletes the CDP table of information about neighbors.
show cdp	Displays global information, such as frequency of transmissions and the holdtime for packets being sent.
show cdp entry *entry-name* [**version**] [**protocol**]	Displays information about a specific neighbor. You can enter an asterisk (*) to display all CDP neighbors, or you can enter the name of the neighbor about which you want information. You can also limit the display to information about the protocols enabled on the specified neighbor or information about the version of software running on the device.
show cdp interface [*interface-id*]	Displays information about interfaces where CDP is enabled. You can limit the display to the interface about which you want information.

Command	Description
show cdp neighbors [*interface-id*] [*detail*]	Displays information about neighbors, including device type, interface type and number, holdtime settings, capabilities, platform, and port ID. You can limit the display to neighbors of a specific interface or expand the display to provide more detailed information.
show cdp traffic	Displays CDP counters, including the number of packets sent and received and checksum errors.

Related Topics

Configuring CDP Characteristics, on page 1650

Additional References

Related Documents

Related Topic	Document Title
System Management Commands	*Configuration Fundamentals Command Reference, Cisco IOS XE Release 3S (Catalyst 3850 Switches)*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Cisco Discovery Protocol

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 79

Configuring Simple Network Management Protocol

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for SNMP

Supported SNMP Versions

This software release supports the following SNMP versions:

- SNMPv1—The Simple Network Management Protocol, a Full Internet Standard, defined in RFC 1157.

- SNMPv2C replaces the Party-based Administrative and Security Framework of SNMPv2Classic with the community-string-based Administrative Framework of SNMPv2C while retaining the bulk retrieval and improved error handling of SNMPv2Classic. It has these features:
  - SNMPv2—Version 2 of the Simple Network Management Protocol, a Draft Internet Standard, defined in RFCs 1902 through 1907.
  - SNMPv2C—The community-string-based Administrative Framework for SNMPv2, an Experimental Internet Protocol defined in RFC 1901.
- SNMPv3—Version 3 of the SNMP is an interoperable standards-based protocol defined in RFCs 2273 to 2275. SNMPv3 provides secure access to devices by authenticating and encrypting packets over the network and includes these security features:
  - Message integrity—Ensures that a packet was not tampered with in transit.
  - Authentication—Determines that the message is from a valid source.
  - Encryption—Mixes the contents of a package to prevent it from being read by an unauthorized source.

Note To select encryption, enter the **priv** keyword.

Both SNMPv1 and SNMPv2C use a community-based form of security. The community of managers able to access the agent's MIB is defined by an IP address access control list and password.

SNMPv2C includes a bulk retrieval function and more detailed error message reporting to management stations. The bulk retrieval function retrieves tables and large quantities of information, minimizing the number of round-trips required. The SNMPv2C improved error-handling includes expanded error codes that distinguish different kinds of error conditions; these conditions are reported through a single error code in SNMPv1. Error return codes in SNMPv2C report the error type.

SNMPv3 provides for both security models and security levels. A security model is an authentication strategy set up for a user and the group within which the user resides. A security level is the permitted level of security within a security model. A combination of the security level and the security model determine which security method is used when handling an SNMP packet. Available security models are SNMPv1, SNMPv2C, and SNMPv3.

The following table identifies characteristics and compares different combinations of security models and levels:

Table 170: SNMP Security Models and Levels

Model	Level	Authentication	Encryption	Result
SNMPv1	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
SNMPv2C	noAuthNoPriv	Community string	No	Uses a community string match for authentication.

Model	Level	Authentication	Encryption	Result
SNMPv3	noAuthNoPriv	Username	No	Uses a username match for authentication.
SNMPv3	authNoPriv	Message Digest 5 (MD5) or Secure Hash Algorithm (SHA)	No	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms.
SNMPv3	authPriv	MD5 or SHA	Data Encryption Standard (DES) or Advanced Encryption Standard (AES)	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Allows specifying the User-based Security Model (USM) with these encryption algorithms: • DES 56-bit encryption in addition to authentication based on the CBC-DES (DES-56) standard. • 3DES 168-bit encryption • AES 128-bit, 192-bit, or 256-bit encryption

You must configure the SNMP agent to use the SNMP version supported by the management station. Because an agent can communicate with multiple managers, you can configure the software to support communications using SNMPv1, SNMPv2C, or SNMPv3.

Restrictions for SNMP

Version Restrictions

- SNMPv1 does not support informs.

Information About SNMP

SNMP Overview

SNMP is an application-layer protocol that provides a message format for communication between managers and agents. The SNMP system consists of an SNMP manager, an SNMP agent, and a management information base (MIB). The SNMP manager can be part of a network management system (NMS) such as Cisco Prime Infrastructure. The agent and MIB reside on the switch. To configure SNMP on the switch, you define the relationship between the manager and the agent.

The SNMP agent contains MIB variables whose values the SNMP manager can request or change. A manager can get a value from an agent or store a value into the agent. The agent gathers data from the MIB, the repository for information about device parameters and network data. The agent can also respond to a manager's requests to get or set data.

An agent can send unsolicited traps to the manager. Traps are messages alerting the SNMP manager to a condition on the network. Traps can mean improper user authentication, restarts, link status (up or down), MAC address tracking, closing of a TCP connection, loss of connection to a neighbor, or other significant events.

The active switch handles the SNMP requests and traps for the whole switch stack. The active switch transparently manages any requests or traps that are related to all stack members. When a new active switch is elected, the new active switch continues to handle SNMP requests and traps as configured on the previous active switch, assuming that IP connectivity to the SNMP management stations is still in place after the new active switch has taken control.

SNMP Manager Functions

The SNMP manager uses information in the MIB to perform the operations described in the following table:

Table 171: SNMP Operations

Operation	Description
get-request	Retrieves a value from a specific variable.
get-next-request	Retrieves a value from a variable within a table.[30]
get-bulk-request[31]	Retrieves large blocks of data, such as multiple rows in a table, that would otherwise require the transmission of many small blocks of data.
get-response	Replies to a get-request, get-next-request, and set-request sent by an NMS.
set-request	Stores a value in a specific variable.

Operation	Description
trap	An unsolicited message sent by an SNMP agent to an SNMP manager when some event has occurred.

30 With this operation, an SNMP manager does not need to know the exact variable name. A sequential search is performed to find the needed variable from within a table.

31 The get-bulk command only works with SNMPv2 or later.

SNMP Agent Functions

The SNMP agent responds to SNMP manager requests as follows:

- Get a MIB variable—The SNMP agent begins this function in response to a request from the NMS. The agent retrieves the value of the requested MIB variable and responds to the NMS with that value.
- Set a MIB variable—The SNMP agent begins this function in response to a message from the NMS. The SNMP agent changes the value of the MIB variable to the value requested by the NMS.

The SNMP agent also sends unsolicited trap messages to notify an NMS that a significant event has occurred on the agent. Examples of trap conditions include, but are not limited to, when a port or module goes up or down, when spanning-tree topology changes occur, and when authentication failures occur.

SNMP Community Strings

SNMP community strings authenticate access to MIB objects and function as embedded passwords. In order for the NMS to access the switch, the community string definitions on the NMS must match at least one of the three community string definitions on the switch.

A community string can have one of the following attributes:

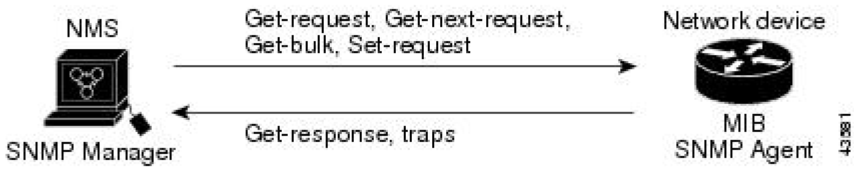
- Read-only (RO)—Gives all objects in the MIB except the community strings read access to authorized management stations, but does not allow write access.
- Read-write (RW)—Gives all objects in the MIB read and write access to authorized management stations, but does not allow access to the community strings.
- When a cluster is created, the command switch manages the exchange of messages among member switches and the SNMP application. The Network Assistant software appends the member switch number (@esN, where N is the switch number) to the first configured RW and RO community strings on the command switch and propagates them to the member switches.

SNMP MIB Variables Access

An example of an NMS is the Cisco Prime Infrastructure network management software. Cisco Prime Infrastructure 2.0 software uses the switch MIB variables to set device variables and to poll devices on the network for specific information. The results of a poll can be displayed as a graph and analyzed to troubleshoot internetworking problems, increase network performance, verify the configuration of devices, monitor traffic loads, and more.

As shown in the figure, the SNMP agent gathers data from the MIB. The agent can send traps, or notification of certain events, to the SNMP manager, which receives and processes the traps. Traps alert the SNMP manager to a condition on the network such as improper user authentication, restarts, link status (up or down), MAC address tracking, and so forth. The SNMP agent also responds to MIB-related queries sent by the SNMP manager in get-request, get-next-request, and set-request format.

Figure 74: SNMP Network



SNMP Notifications

SNMP allows the switch to send notifications to SNMP managers when particular events occur. SNMP notifications can be sent as traps or inform requests. In command syntax, unless there is an option in the command to select either traps or informs, the keyword traps refers to either traps or informs, or both. Use the **snmp-server host** command to specify whether to send SNMP notifications as traps or informs.

Note SNMPv1 does not support informs.

Traps are unreliable because the receiver does not send an acknowledgment when it receives a trap, and the sender cannot determine if the trap was received. When an SNMP manager receives an inform request, it acknowledges the message with an SNMP response protocol data unit (PDU). If the sender does not receive a response, the inform request can be sent again. Because they can be resent, informs are more likely than traps to reach their intended destination.

The characteristics that make informs more reliable than traps also consume more resources in the switch and in the network. Unlike a trap, which is discarded as soon as it is sent, an inform request is held in memory until a response is received or the request times out. Traps are sent only once, but an inform might be resent or retried several times. The retries increase traffic and contribute to a higher overhead on the network. Therefore, traps and informs require a trade-off between reliability and resources. If it is important that the SNMP manager receive every notification, use inform requests. If traffic on the network or memory in the switch is a concern and notification is not required, use traps.

SNMP ifIndex MIB Object Values

In an NMS, the IF-MIB generates and assigns an interface index (ifIndex) object value that is a unique number greater than zero to identify a physical or a logical interface. When the switch reboots or the switch software is upgraded, the switch uses this same value for the interface. For example, if the switch assigns a port 2 an ifIndex value of 10003, this value is the same after the switch reboots.

The switch uses one of the values in the following table to assign an ifIndex value to an interface:

Table 172: ifIndex Values

Interface Type	ifIndex Range
SVI[32]	1–4999

Interface Type	ifIndex Range
EtherChannel	5000–5012
Loopback	5013–5077
Tunnel	5078–5142
Physical (such as Gigabit Ethernet or SFP[33]-module interfaces)	10000–14500
Null	14501

32 SVI = switch virtual interface
33 SFP = small form-factor pluggable

Default SNMP Configuration

Feature	Default Setting
SNMP agent	Disabled[34].
SNMP trap receiver	None configured.
SNMP traps	None enabled except the trap for TCP connections (tty).
SNMP version	If no version keyword is present, the default is Version 1.
SNMPv3 authentication	If no keyword is entered, the default is the **noauth** (noAuthNoPriv) security level.
SNMP notification type	If no type is specified, all notifications are sent.

34 This is the default when the switch starts and the startup configuration does not have any **snmp-server** global configuration commands.

SNMP Configuration Guidelines

If the switch starts and the switch startup configuration has at least one **snmp-server** global configuration command, the SNMP agent is enabled.

An SNMP *group* is a table that maps SNMP users to SNMP views. An SNMP *user* is a member of an SNMP group. An SNMP *host* is the recipient of an SNMP trap operation. An SNMP *engine ID* is a name for the local or remote SNMP engine.

When configuring SNMP, follow these guidelines:

- When configuring an SNMP group, do not specify a notify view. The **snmp-server host** global configuration command auto-generates a notify view for the user and then adds it to the group associated with that user. Modifying the group's notify view affects all users associated with that group.
- To configure a remote user, specify the IP address or port number for the remote SNMP agent of the device where the user resides.
- Before you configure remote users for a particular agent, configure the SNMP engine ID, using the **snmp-server engineID** global configuration command with the **remote** option. The remote agent's

SNMP engine ID and user password are used to compute the authentication and privacy digests. If you do not configure the remote engine ID first, the configuration command fails.

- When configuring SNMP informs, you need to configure the SNMP engine ID for the remote agent in the SNMP database before you can send proxy requests or informs to it.
- If a local user is not associated with a remote host, the switch does not send informs for the **auth** (authNoPriv) and the **priv** (authPriv) authentication levels.
- Changing the value of the SNMP engine ID has significant results. A user's password (entered on the command line) is converted to an MD5 or SHA security digest based on the password and the local engine ID. The command-line password is then destroyed, as required by RFC 2274. Because of this deletion, if the value of the engine ID changes, the security digests of SNMPv3 users become invalid, and you need to reconfigure SNMP users by using the **snmp-server user** *username* global configuration command. Similar restrictions require the reconfiguration of community strings when the engine ID changes.

How to Configure SNMP

Disabling the SNMP Agent

The **no snmp-server** global configuration command disables all running versions (Version 1, Version 2C, and Version 3) of the SNMP agent on the device. You reenable all versions of the SNMP agent by the first **snmp-server** global configuration command that you enter. There is no Cisco IOS command specifically designated for enabling SNMP.

Beginning in privileged EXEC mode, follow these steps to disable the SNMP agent.

Before You Begin

The SNMP Agent must be enabled before it can be disabled. The SNMP agent is enabled by the first **snmp-server** global configuration command entered on the device.

SUMMARY STEPS

1. **configure terminal**
2. **no snmp-server**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	**no snmp-server** **Example:** `Switch(config)# no snmp-server`	Disables the SNMP agent operation.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring Community Strings

You use the SNMP community string to define the relationship between the SNMP manager and the agent. The community string acts like a password to permit access to the agent on the switch. Optionally, you can specify one or more of these characteristics associated with the string:

- An access list of IP addresses of the SNMP managers that are permitted to use the community string to gain access to the agent
- A MIB view, which defines the subset of all MIB objects accessible to the given community
- Read and write or read-only permission for the MIB objects accessible to the community

Beginning in privileged EXEC mode, follow these steps to configure a community string on the switch.

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server community** *string* [**view** *view-name*] [**ro** | **rw**] [*access-list-number*]
3. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	**snmp-server community** *string* [**view** *view-name*] [**ro** | **rw**] [*access-list-number*] **Example:** `Switch(config)# snmp-server community comaccess ro 4`	Configures the community string. **Note** The @ symbol is used for delimiting the context information. Avoid using the @ symbol as part of the SNMP community string when configuring this command. • For *string*, specify a string that acts like a password and permits access to the SNMP protocol. You can configure one or more community strings of any length. • (Optional) For *view-name*, specify the view record accessible to the community. • (Optional) Specify either read-only (**ro**) if you want authorized management stations to retrieve MIB objects, or specify read-write (**rw**) if you want authorized management stations to retrieve and modify MIB objects. By default, the community string permits read-only access to all objects. • (Optional) For *access-list-number*, enter an IP standard access list numbered from 1 to 99 and 1300 to 1999.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] **Example:** `Switch(config)# access-list 4 deny any`	(Optional) If you specified an IP standard access list number in Step 2, then create the list, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number specified in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, enter the IP address of the SNMP managers that are permitted to use the community string to gain access to the agent. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

This example shows how to assign the comaccess string to SNMP, to allow read-only access, and to specify that IP access list 4 can use the community string to gain access to the switch SNMP agent:

```
Switch(config)# snmp-server community comaccess ro 4
```

What to Do Next

To disable access for an SNMP community, set the community string for that community to the null string (do not enter a value for the community string).

To remove a specific community string, use the **no snmp-server** community string global configuration command.

You can specify an identification name (engine ID) for the local or remote SNMP server engine on the switch. You can configure an SNMP server group that maps SNMP users to SNMP views, and you can add new users to the SNMP group.

Configuring SNMP Groups and Users

You can specify an identification name (engine ID) for the local or remote SNMP server engine on the switch. You can configure an SNMP server group that maps SNMP users to SNMP views, and you can add new users to the SNMP group.

Beginning in privileged EXEC mode, follow these steps to configure SNMP groups and users on the switch.

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server engineID** {**local** *engineid-string* | **remote** *ip-address* [**udp-port** *port-number*] *engineid-string*}
3. **snmp-server group** *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]
4. **snmp-server user** *username group-name* {**remote** *host* [**udp-port** *port*]} {**v1** [**access** *access-list*] | **v2c** [**access** *access-list*] | **v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** | **sha**} *auth-password*] } [*priv* {**des** | **3des** | **aes** {**128** | **192** | **256**}} *priv-password*]
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**snmp-server engineID** {**local** *engineid-string* | **remote** *ip-address* [**udp-port** *port-number*] *engineid-string*} **Example:** `Switch(config)# snmp-server engineID local 1234`	Configures a name for either the local or remote copy of SNMP. • The *engineid-string* is a 24-character ID string with the name of the copy of SNMP. You need not specify the entire 24-character engine ID if it has trailing zeros. Specify only the portion of the engine ID up to the point where only zeros remain in the value. The Step Example configures an engine ID of 123400000000000000000000. • If you select **remote**, specify the *ip-address* of the device that contains the remote copy of SNMP and the optional User Datagram Protocol (UDP) port on the remote device. The default is 162.

	Command or Action	Purpose
Step 3	**snmp-server group** *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*] **Example:** `Switch(config)# snmp-server group public v2c access lmnop`	Configures a new SNMP group on the remote device. For *group-name*, specify the name of the group. Specify one of the following security models: • **v1** is the least secure of the possible security models. • **v2c** is the second least secure model. It allows transmission of informs and integers twice the normal width. • **v3**, the most secure, requires you to select one of the following authentication levels: **auth—**Enables the Message Digest 5 (MD5) and the Secure Hash Algorithm (SHA) packet authentication. **noauth—**Enables the noAuthNoPriv security level. This is the default if no keyword is specified. **priv—**Enables Data Encryption Standard (DES) packet encryption (also called privacy). (Optional) Enter **read** *readview* with a string (not to exceed 64 characters) that is the name of the view in which you can only view the contents of the agent. (Optional) Enter **write** *writeview* with a string (not to exceed 64 characters) that is the name of the view in which you enter data and configure the contents of the agent. (Optional) Enter **notify** *notifyview* with a string (not to exceed 64 characters) that is the name of the view in which you specify a notify, inform, or trap. (Optional) Enter **access** *access-list* with a string (not to exceed 64 characters) that is the name of the access list.
Step 4	**snmp-server user** *username* *group-name* {**remote** *host* [**udp-port** *port*]} {**v1** [**access** *access-list*] | **v2c** [**access** *access-list*] | **v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** | **sha**} *auth-password*] } [*priv* {**des** | **3des** | **aes** {**128** | **192** | **256**}} *priv-password*] **Example:** `Switch(config)#  snmp-server user Pat public v2c`	Adds a new user for an SNMP group. The *username* is the name of the user on the host that connects to the agent. The *group-name* is the name of the group to which the user is associated. Enter **remote** to specify a remote SNMP entity to which the user belongs and the hostname or IP address of that entity with the optional UDP port number. The default is 162. Enter the SNMP version number (**v1**, **v2c**, or **v3**). If you enter **v3**, you have these additional options: • **encrypted** specifies that the password appears in encrypted format. This keyword is available only when the **v3** keyword is specified. • **auth** is an authentication level setting session that can be either the HMAC-MD5-96 (**md5**) or the HMAC-SHA-96 (**sha**) authentication level and requires a password string *auth-password* (not to exceed 64 characters).

	Command or Action	Purpose
		If you enter **v3** you can also configure a private (**priv**) encryption algorithm and password string *priv-password* using the following keywords (not to exceed 64 characters): • **priv** specifies the User-based Security Model (USM). • **des** specifies the use of the 56-bit DES algorithm. • **3des** specifies the use of the 168-bit DES algorithm. • **aes** specifies the use of the DES algorithm. You must select either 128-bit, 192-bit, or 256-bit encryption. (Optional) Enter **access** *access-list* with a string (not to exceed 64 characters) that is the name of the access list.
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring SNMP Notifications

A trap manager is a management station that receives and processes traps. Traps are system alerts that the switch generates when certain events occur. By default, no trap manager is defined, and no traps are sent. Switches running this Cisco IOS release can have an unlimited number of trap managers.

Note Many commands use the word **traps** in the command syntax. Unless there is an option in the command to select either traps or informs, the keyword **traps** refers to traps, informs, or both. Use the **snmp-server host** global configuration command to specify whether to send SNMP notifications as traps or informs.

You can use the **snmp-server host** global configuration command for a specific host to receive the notification types listed in the following table. You can enable any or all of these traps and configure a trap manager to receive them.

Table 173: Device Notification Types

Notification Type Keyword	Description
bridge	Generates STP bridge MIB traps.
cluster	Generates a trap when the cluster configuration changes.
config	Generates a trap for SNMP configuration changes.
copy-config	Generates a trap for SNMP copy configuration changes.

Notification Type Keyword	Description
cpu threshold	Allow CPU-related traps.
entity	Generates a trap for SNMP entity changes.
envmon	Generates environmental monitor traps. You can enable any or all of these environmental traps: fan, shutdown, status, supply, temperature.
flash	Generates SNMP FLASH notifications. In a switch stack, you can optionally enable notification for flash insertion or removal, which would cause a trap to be issued whenever a switch in the stack is removed or inserted (physical removal, power cycle, or reload).
fru-ctrl	Generates entity field-replaceable unit (FRU) control traps. In the switch stack, this trap refers to the insertion or removal of a switch in the stack.
hsrp	Generates a trap for Hot Standby Router Protocol (HSRP) changes.
ipmulticast	Generates a trap for IP multicast routing changes.
mac-notification	Generates a trap for MAC address notifications.
ospf	Generates a trap for Open Shortest Path First (OSPF) changes. You can enable any or all of these traps: Cisco specific, errors, link-state advertisement, rate limit, retransmit, and state changes.
pim	Generates a trap for Protocol-Independent Multicast (PIM) changes. You can enable any or all of these traps: invalid PIM messages, neighbor changes, and rendezvous point (RP)-mapping changes.
port-security	Generates SNMP port security traps. You can also set a maximum trap rate per second. The range is from 0 to 1000; the default is 0, which means that there is no rate limit. **Note** When you configure a trap by using the notification type **port-security**, configure the port security trap first, and then configure the port security trap rate: 1 **snmp-server enable traps port-security** 2 **snmp-server enable traps port-security trap-rate** *rate*
snmp	Generates a trap for SNMP-type notifications for authentication, cold start, warm start, link up or link down.
storm-control	Generates a trap for SNMP storm-control. You can also set a maximum trap rate per minute. The range is from 0 to 1000; the default is 0 (no limit is imposed; a trap is sent at every occurrence).
stpx	Generates SNMP STP Extended MIB traps.
syslog	Generates SNMP syslog traps.
tty	Generates a trap for TCP connections. This trap is enabled by default.

Notification Type Keyword	Description
vlan-membership	Generates a trap for SNMP VLAN membership changes.
vlancreate	Generates SNMP VLAN created traps.
vlandelete	Generates SNMP VLAN deleted traps.
vtp	Generates a trap for VLAN Trunking Protocol (VTP) changes.

Beginning in privileged EXEC mode, follow these steps to configure the switch to send traps or informs to a host.

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server engineID remote** *ip-address engineid-string*
3. **snmp-server user** *username group-name* {**remote** *host* [**udp-port** *port*]} {**v1** [**access** *access-list*] | **v2c** [**access** *access-list*] | **v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** | **sha**} *auth-password*] }
4. **snmp-server group** *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]
5. **snmp-server host** *host-addr* [**informs** | **traps**] [**version** {**1** | **2c** | **3** {**auth** | **noauth** | **priv**}}] *community-string* [*notification-type*]
6. **snmp-server enable traps** *notification-types*
7. **snmp-server trap-source** *interface-id*
8. **snmp-server queue-length** *length*
9. **snmp-server trap-timeout** *seconds*
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**snmp-server engineID remote** *ip-address engineid-string* **Example:** `Switch(config)# snmp-server engineID remote 192.180.1.27 00000063000100a1c0b4011b`	Specifies the engine ID for the remote host.
Step 3	**snmp-server user** *username group-name* {**remote** *host* [**udp-port** *port*]} {**v1** [**access** *access-list*] |	Configures an SNMP user to be associated with the remote host created in Step 2.

	Command or Action	Purpose
	v2c [**access** *access-list*] | **v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** | **sha**} *auth-password*] } **Example:** `Switch(config)#  snmp-server user Pat public v2c`	**Note** You cannot configure a remote user for an address without first configuring the engine ID for the remote host. Otherwise, you receive an error message, and the command is not executed.
Step 4	**snmp-server group** *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*] **Example:** `Switch(config)# snmp-server group public v2c access lmnop`	Configures an SNMP group.
Step 5	**snmp-server host** *host-addr* [**informs** | **traps**] [**version** {**1** | **2c** | **3** {**auth** | **noauth** | **priv**}}] *community-string* [*notification-type*] **Example:** `Switch(config)# snmp-server host 203.0.113.1 comaccess snmp`	Specifies the recipient of an SNMP trap operation. For *host-addr*, specify the name or Internet address of the host (the targeted recipient). (Optional) Specify **traps** (the default) to send SNMP traps to the host. Specify **informs** to send SNMP informs to the host. (Optional) Specify the SNMP **version** (**1**, **2c**, or **3**). SNMPv1 does not support informs. (Optional) For Version 3, select authentication level **auth**, **noauth**, or **priv**. For *community-string*, when **version 1** or **version 2c** is specified, enter the password-like community string sent with the notification operation. When **version 3** is specified, enter the SNMPv3 username. The @ symbol is used for delimiting the context information. Avoid using the @ symbol as part of the SNMP community string when configuring this command. (Optional) For *notification-type*, use the keywords listed in the table above. If no type is specified, all notifications are sent.
Step 6	**snmp-server enable traps** *notification-types* **Example:** `Switch(config)# snmp-server enable traps snmp`	Enables the switch to send traps or informs and specifies the type of notifications to be sent. For a list of notification types, see the table above, or enter **snmp-server enable traps ?** To enable multiple types of traps, you must enter a separate **snmp-server enable traps** command for each trap type. **Note** When you configure a trap by using the notification type port-security, configure the port security trap first, and then configure the port security trap rate: 1 **snmp-server enable traps port-security** 2 **snmp-server enable traps port-security trap-rate** *rate*

	Command or Action	Purpose
Step 7	**snmp-server trap-source** *interface-id* **Example:** `Switch(config)# snmp-server trap-source GigabitEthernet1/0/1`	(Optional) Specifies the source interface, which provides the IP address for the trap message. This command also sets the source IP address for informs.
Step 8	**snmp-server queue-length** *length* **Example:** `Switch(config)# snmp-server queue-length 20`	(Optional) Establishes the message queue length for each trap host. The range is 1 to 1000; the default is 10.
Step 9	**snmp-server trap-timeout** *seconds* **Example:** `Switch(config)# snmp-server trap-timeout 60`	(Optional) Defines how often to resend trap messages. The range is 1 to 1000; the default is 30 seconds.
Step 10	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

What to Do Next

The **snmp-server host** command specifies which hosts receive the notifications. The **snmp-server enable trap** command globally enables the method for the specified notification (for traps and informs). To enable a host to receive an inform, you must configure an **snmp-server host informs** command for the host and globally enable informs by using the **snmp-server enable traps** command.

To remove the specified host from receiving traps, use the **no snmp-server host** *host* global configuration command. The **no snmp-server host** command with no keywords disables traps, but not informs, to the host. To disable informs, use the **no snmp-server host informs** global configuration command. To disable a specific trap type, use the **no snmp-server enable traps** *notification-types* global configuration command.

Setting the Agent Contact and Location Information

Beginning in privileged EXEC mode, follow these steps to set the system contact and location of the SNMP agent so that these descriptions can be accessed through the configuration file.

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server contact** *text*
3. **snmp-server location** *text*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**snmp-server contact** *text* **Example:** `Switch(config)# snmp-server contact Dial System Operator at beeper 21555`	Sets the system contact string.
Step 3	**snmp-server location** *text* **Example:** `Switch(config)# snmp-server location Building 3/Room 222`	Sets the system location string.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Limiting TFTP Servers Used Through SNMP

Beginning in privileged EXEC mode, follow these steps to limit the TFTP servers used for saving and loading configuration files through SNMP to the servers specified in an access list.

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server tftp-server-list** *access-list-number*
3. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**snmp-server tftp-server-list** *access-list-number* **Example:** `Switch(config)# snmp-server tftp-server-list 44`	Limits the TFTP servers used for configuration file copies through SNMP to the servers in the access list. For *access-list-number*, enter an IP standard access list numbered from 1 to 99 and 1300 to 1999.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] **Example:** `Switch(config)# access-list 44 permit 10.1.1.2`	Creates a standard access list, repeating the command as many times as necessary. For *access-list-number*, enter the access list number specified in Step 2. The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. For *source*, enter the IP address of the TFTP servers that can access the switch. (Optional) For *source-wildcard*, enter the wildcard bits, in dotted decimal notation, to be applied to the source. Place ones in the bit positions that you want to ignore. The access list is always terminated by an implicit deny statement for everything.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring Trap Flags for SNMP

SUMMARY STEPS

1. **configure terminal**
2. **trapflags ap** { **interfaceup** | **register**}
3. **trapflags client** {**dot11** | **excluded**}
4. **trapflags dot11-security** {**ids-sig-attack** | **wep-decrypt-error**}
5. **trapflags mesh**
6. **trapflags rogueap**
7. **trapflags rrm-params** {**channels** | **tx-power**}
8. **trapflags rrm-profile** {**coverage** | **interference** | **load** | **noise**}
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**trapflags ap** { **interfaceup** | **register**} **Example:** `Switch(config)# trapflags ap interfaceup`	Enables sending AP-related traps. Use the **no** form of the command to disable the trap flags. • **interfaceup–** Enables trap when a Cisco AP interface (A or B) comes up. • **register–** Enables trap when a Cisco AP registers with a Cisco switch.
Step 3	**trapflags client** {**dot11** | **excluded**} **Example:** `Switch(config)# trapflags client excluded`	Enables sending client-related dot11 traps. Use the **no** form of the command to disable the trap flags. • **dot11–** Enables Dot11 traps for clients. • **excluded–** Enables excluded traps for clients.
Step 4	**trapflags dot11-security** {**ids-sig-attack** | **wep-decrypt-error**} **Example:** `Switch(config)# trapflags dot11-security wep-decrypt-error`	Enables sending 802.11 security-related traps. Use the **no** form of the command to disable the trap flags. • **ids-sig-attack–** Enables IDS signature attack traps. • **wep-decrypt-error–** Enables traps for WEP decrypt error for clients.

	Command or Action	Purpose
Step 5	**trapflags mesh** **Example:** `Switch(config)# trapflags mesh`	Enables trap for the mesh. Use the **no** form of the command to disable the trap flags.
Step 6	**trapflags rogueap** **Example:** `Switch(config)# trapflags rogueap`	Enables trap for rogue AP detection. Use the **no** form of the command to disable the trap flags.
Step 7	**trapflags rrm-params** {**channels** | **tx-power**} **Example:** `Switch(config)# trapflags rrm-params tx-power`	Enables sending RRM-parameter update-related traps. Use the **no** form of the command to disable the trap flags. • **channels–** Enables trap when RF Manager automatically changes a channel number for the Cisco AP interface. • **tx-power–** Enables the trap when RF Manager automatically changes Tx-Power level for the Cisco AP interface.
Step 8	**trapflags rrm-profile** {**coverage** | **interference** | **load** | **noise**} **Example:** `Switch(config)# trapflags rrm-profile interference`	Enables sending RRM-profile-related traps. Use the **no** form of the command to disable the trap flags. • **coverage–** Enables the trap when the coverage profile maintained by RF Manager fails. • **interference–** Enables the trap when the interference profile maintained by RF Manager fails. • **load–** Enables trap when the load profile maintained by RF Manager fails. • **noise–** Enables trap when the noise profile maintained by RF Manager fails.
Step 9	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Enabling SNMP Wireless Trap Notification

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server enable traps wireless** [**AP** | **RRM** | **bsn80211SecurityTrap** | **bsnAPParamUpdate** | **bsnAPProfile** | **bsnAccessPoint** | **bsnMobileStation** | **bsnRogue** | **client** | **mfp** | **rogue**]
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**snmp-server enable traps wireless** [**AP** | **RRM** | **bsn80211SecurityTrap** | **bsnAPParamUpdate** | **bsnAPProfile** | **bsnAccessPoint** | **bsnMobileStation** | **bsnRogue** | **client** | **mfp** | **rogue**] **Example:** `Switch(config)# snmp-server enable traps wireless AP`	Enables SNMP wireless trap notification. • **AP–** Enables access point traps. • **RRM–** Enables RRM traps. • **bsn80211SecurityTrap–** Enables the security-related trap. • **bsnAPParamUpdate–** Enables the trap for AP parameters that get updated. • **bsnAPProfile–** Enables BSN AP profile traps. • **bsnAccessPoint–** Enables BSN access point traps. • **bsnMobileStation–** Controls wireless client traps. • **bsnRogue–** Enables BSN rogue-related traps. • **client–** Enables client traps. • **mfp–** Enables MFP traps. • **rogue–** Enables rogue-related traps.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Monitoring SNMP Status

To display SNMP input and output statistics, including the number of illegal community string entries, errors, and requested variables, use the **show snmp** privileged EXEC command. You also can use the other privileged EXEC commands listed in the table to display SNMP information.

Table 174: Commands for Displaying SNMP Information

Command	Purpose
show snmp	Displays SNMP statistics.
show snmp engineID	Displays information on the local SNMP engine and all remote engines that have been configured on the device.
show snmp group	Displays information on each SNMP group on the network.
show snmp pending	Displays information on pending SNMP requests.
show snmp sessions	Displays information on the current SNMP sessions.
show snmp user	Displays information on each SNMP user name in the SNMP users table. **Note** You must use this command to display SNMPv3 configuration information for **auth** | **noauth** | **priv** mode. This information is not displayed in the **show running-config** output.

SNMP Examples

This example shows how to enable all versions of SNMP. The configuration permits any SNMP manager to access all objects with read-only permissions using the community string *public*. This configuration does not cause the switch to send any traps.

```
Switch(config)# snmp-server community public
```

This example shows how to permit any SNMP manager to access all objects with read-only permission using the community string *public*. The switch also sends VTP traps to the hosts 192.180.1.111 and 192.180.1.33 using SNMPv1 and to the host 192.180.1.27 using SNMPv2C. The community string *public* is sent with the traps.

```
Switch(config)# snmp-server community public
Switch(config)# snmp-server enable traps vtp
Switch(config)# snmp-server host 192.180.1.27 version 2c public
Switch(config)# snmp-server host 192.180.1.111 version 1 public
Switch(config)# snmp-server host 192.180.1.33 public
```

This example shows how to allow read-only access for all objects to members of access list 4 that use the *comaccess* community string. No other SNMP managers have access to any objects. SNMP Authentication Failure traps are sent by SNMPv2C to the host *cisco.com* using the community string *public*.

```
Switch(config)# snmp-server community comaccess ro 4
Switch(config)# snmp-server enable traps snmp authentication
Switch(config)# snmp-server host cisco.com version 2c public
```

This example shows how to send Entity MIB traps to the host *cisco.com*. The community string is restricted. The first line enables the switch to send Entity MIB traps in addition to any traps previously enabled. The second line specifies the destination of these traps and overwrites any previous **snmp-server** host commands for the host *cisco.com*.

```
Switch(config)# snmp-server enable traps entity
Switch(config)# snmp-server host cisco.com restricted entity
```

This example shows how to enable the switch to send all traps to the host *myhost.cisco.com* using the community string *public*:

```
Switch(config)# snmp-server enable traps
Switch(config)# snmp-server host myhost.cisco.com public
```

This example shows how to associate a user with a remote host and to send **auth** (authNoPriv) authentication-level informs when the user enters global configuration mode:

```
Switch(config)# snmp-server engineID remote 192.180.1.27 00000063000100a1c0b4011b
Switch(config)# snmp-server group authgroup v3 auth
Switch(config)# snmp-server user authuser authgroup remote 192.180.1.27 v3 auth md5 mypassword
Switch(config)# snmp-server user authuser authgroup v3 auth md5 mypassword
Switch(config)# snmp-server host 192.180.1.27 informs version 3 auth authuser config
Switch(config)# snmp-server enable traps
Switch(config)# snmp-server inform retries 0
```

CHAPTER 80

Configuring Cache Services Using the Web Cache Communication Protocol

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for WCCP

Before configuring WCCP on your switch, make sure you adhere to the following configuration prerequisites:

- The application engines and switches in the same service group must be in the same subnetwork directly connected to the switch that has WCCP enabled.
- Configure the switch interfaces that are connected to the clients, the application engines, and the server as Layer 3 interfaces (routed ports and switch virtual interfaces [SVIs]). For WCCP packet redirection to work, the servers, application engines, and clients must be on different subnets.

- Use only nonreserved multicast addresses when configuring a single multicast address for each application engine.
- WCCP entries and PBR entries use the same TCAM region. WCCP is supported only on the templates that support PBR: access, routing, and dual IPv4/v6 routing.
- When TCAM entries are not available to add WCCP entries, packets are not redirected and are forwarded by using the standard routing tables.
- The number of available policy-based routing (PBR) labels are reduced as more interfaces are enabled for WCCP ingress redirection. For every interface that supports service groups, one label is consumed. The WCCP labels are taken from the PBR labels. You need to monitor and manage the labels that are available between PBR and WCCP. When labels are not available, the switch cannot add service groups. However, if another interface has the same sequence of service groups, a new label is not needed, and the group can be added to the interface.
- The routing maximum transmission unit (MTU) size configured on the stack member switches should be larger than the client MTU size. The MAC-layer MTU size configured on ports connected to application engines should consider the GRE tunnel header bytes.

Restrictions for WCCP

Unsupported WCCP Features

The following WCCP features are not supported in this software release:

- Packet redirection on an outbound interface that is configured by using the **ip wccp redirect out** interface configuration command.
- The GRE forwarding method for packet redirection.
- GRE redirect and return.
- The hash assignment method for load balancing.
- SNMP support for WCCP.
- Hash assignments in hardware. You can load balance using mask assignments only.
- Redirection for fragmented packets. This is a security feature.

General Restrictions

- Maximum number of service groups: eight ingress and eight egress.
- You cannot configure WCCP and VPN routing/forwarding (VRF) on the same switch interface.
- You cannot configure WCCP and PBR on the same switch interface.
- You cannot configure WCCP and a private VLAN (PVLAN) on the same switch interface.
- The **ip wccp redirect exclude in** command allows you to exclude ingress packets from egress WCCP methods. It is not needed on the interface to CE.
- When no cache engine is available, matching packets are dropped. This is closed group support. There is no VRF-aware WCCP support and no IPv6 WCCP.

- When the device is configured with the **ip wccp check services all** command, if the redirect ACL fails to match on packet, it will be checked against the next priority service group.

Information About WCCP

WCCP Overview

Note To use this feature, the device must be running the IP Services feature set.

WCCP is a Cisco-developed content-routing technology that you can use to integrate wide-area application engines (referred to as application engines) into your network infrastructure. The application engines transparently store frequently accessed content and then fulfill successive requests for the same content, eliminating repetitive transmissions of identical content from servers. Application engines accelerate content delivery and ensure maximum scalability and availability of content. In a service-provider network, you can deploy the WCCP and application engine solution at the points of presence (POPs). In an enterprise network, you can deploy the WCCP and application engine solution at a regional site or small branch office.

The WCCP and Cisco cache engines (or other application engines running WCCP) localize traffic patterns in the network, enabling content requests to be fulfilled locally.

WCCP enables supported Cisco routers and switches to transparently redirect content requests. With transparent redirection, users do not have to configure their browsers to use a web proxy. Instead, they can use the target URL to request content, and their requests are automatically redirected to an application engine. The word transparent means that the end user does not know that a requested file (such as a web page) came from the application engine instead of from the originally specified server.

When an application engine receives a request, it attempts to service it from its own local cache. If the requested information is not present, the application engine sends a separate request to the end server to retrieve the requested information. After receiving the requested information, the application engine forwards it to the requesting client and also caches it to fulfill future requests.

With WCCP, the application-engine cluster (a series of application engines) can service multiple routers or switches.

WCCP Message Exchange

The following sequence of events describes the WCCP message exchange:

1. The application engines send their IP addresses to the WCCP-enabled switch by using WCCP, signaling their presence through a Here I am message. The switch and application engines communicate to each other through a control channel based on UDP port 2048.
2. The WCCP-enabled switch uses the application engine IP information to create a cluster view (a list of application engines in the cluster). This view is sent through an I see you message to each application engine in the cluster, essentially making all the application engines aware of each other. A stable view is established after the membership of the cluster remains the same for a certain amount of time.
3. When a stable view is established, the application engine in the cluster with the lowest IP address is elected as the designated application engine.

WCCP Negotiation

In the exchange of WCCP protocol messages, the designated application engine and the WCCP-enabled switch negotiate these items:

- Forwarding method (the method by which the switch forwards packets to the application engine). The switch rewrites the Layer 2 header by replacing the packet destination MAC address with the target application engine MAC address. It then forwards the packet to the application engine. This forwarding method requires the target application engine to be directly connected to the switch at Layer 2.
- Assignment method (the method by which packets are distributed among the application engines in the cluster). The switch uses some bits of the destination IP address, the source IP address, the destination Layer 4 port, and the source Layer 4 port to determine which application engine receives the redirected packets.
- Packet-return method (the method by which packets are returned from the application engine to the switch for normal forwarding). These are the typical reasons why an application engine rejects packets and starts the packet-return feature:
    - The application engine is overloaded and has no room to service the packets.
    - The application engine receives an error message (such as a protocol or authentication error) from the server and uses the dynamic client bypass feature. The bypass enables clients to bypass the application engines and to connect directly to the server.

The application engine returns a packet to the WCCP-enabled switch to forward to the server as if the application engine is not present. The application engine does not intercept the reconnection attempt. In this way, the application engine effectively cancels the redirection of a packet to the application engine and creates a bypass flow. If the return method is generic-route encapsulation (GRE), the switch receives the returned packet through a GRE tunnel that is configured in the application engine. The switch CPU uses Cisco Express Forwarding to send these packets to the target server. If the return method is Layer 2 rewrite, the packets are forwarded in hardware to the target server. When the server responds with the information, the switch uses normal Layer 3 forwarding to return the information to the requesting client.

MD5 Security

WCCP provides an optional security component in each protocol message to enable the switch to use MD5 authentication on messages between the switch and the application engine. Messages that do not authenticate by MD5 (when authentication of the switch is enabled) are discarded by the switch. The password string is combined with the MD5 value to create security for the connection between the switch and the application engine. You must configure the same password on each application engine.

Packet Redirection and Service Groups

You can configure WCCP to classify traffic for redirection, such as FTP, proxy-web-cache handling, and audio and video applications. This classification, known as a service group, is based on the protocol type (TCP or UDP) and the Layer 4 source destination port numbers. The service groups are identified either by well-known names such as web-cache, which means TCP port 80, or a service number, 0 to 99. Service groups are configured to map to a protocol and Layer 4 port numbers and are established and maintained independently. WCCP allows dynamic service groups, where the classification criteria are provided dynamically by a participating application engine.

You can configure up to 8 service groups on a switch or switch stack and up to 32 cache engines per service group. WCCP maintains the priority of the service group in the group definition. WCCP uses the priority to configure the service groups in the switch hardware. For example, if service group 1 has a priority of 100 and looks for destination port 80, and service group 2 has a priority of 50 and looks for source port 80, the incoming packet with source and destination port 80 is forwarded by using service group 1 because it has the higher priority.

WCCP supports a cluster of application engines for every service group. Redirected traffic can be sent to any one of the application engines. The switch supports the mask assignment method of load balancing the traffic among the application engines in the cluster for a service group.

After WCCP is configured on the switch, the switch forwards all service group packets received from clients to the application engines. However, the following packets are not redirected:

- Packets originating from the application engine and targeted to the server.
- Packets originating from the application engine and targeted to the client.
- Packets returned or rejected by the application engine. These packets are sent to the server.

You can configure a single multicast address per service group for sending and receiving protocol messages. When there is a single multicast address, the application engine sends a notification to one address, which provides coverage for all routers in the service group, for example, 225.0.0.0. If you add and remove routers dynamically, using a single multicast address provides easier configuration because you do not need to specifically enter the addresses of all devices in the WCCP network.

You can use a router group list to validate the protocol packets received from the application engine. Packets matching the address in the group list are processed, packets not matching the group list address are dropped.

To disable caching for specific clients, servers, or client/server pairs, you can use a WCCP redirect access control list (ACL). Packets that do not match the redirect ACL bypass the cache and are forwarded normally.

Before WCCP packets are redirected, the switch examines ACLs associated with all inbound features configured on the interface and permits or denies packet forwarding based on how the packet matches the entries in the ACL.

Note Beginning with Cisco IOS Release 12.2(58)SE, both permit and deny ACL entries are supported in WCCP redirect lists.

When packets are redirected, the output ACLs associated with the redirected interface are applied to the packets. Any ACLs associated with the original port are not applied unless you specifically configure the required output ACLs on the redirected interfaces.

WCCP and Switch Stacks

WCCP support is the same for a switch stack as for a standalone switch. WCCP configuration information is propagated to all switches in the stack. All switches in the stack, including the stack master, process the information and program their hardware.

The stack master performs the following WCCP functions:

- Receives protocol packets from any WCCP-enabled interface and sends them out any WCCP-enabled interface in the stack.
- Processes the WCCP configuration and propagates the information to all stack members.

- Distributes the WCCP information to any switch that joins the stack.
- Programs its hardware with the WCCP information it processes.

Stack members receive the WCCP information from the master switch and program their hardware.

Default WCCP Configuration

Feature	Default Setting
WCCP enable state	WCCP services are disabled.
Protocol version	WCCPv2.
Redirecting traffic received on an interface	Disabled.

How to Configure WCCP

Enabling the Cache Service

For WCCP packet redirection to operate, you must configure the switch interface connected to the client to redirect inbound packets.

This procedure shows how to configure these features on routed ports. To configure these features on SVIs, see the configuration examples that follow the procedure.

Beginning in privileged EXEC mode, follow these steps to enable the cache service, to set a multicast group address or group list, to configure routed interfaces, to redirect inbound packets received from a client to the application engine, enable an interface to listen for a multicast address, and to set a password. This procedure is required.

Before You Begin

Configure the SDM template, and reboot the device.

SUMMARY STEPS

1. **configure terminal**
2. **ip wccp** {**web-cache** | *service-number*} [**group-address** *groupaddress*] [**group-list** *access-list*] [**redirect-list** *access-list*] [**password** *encryption-number password*]
3. **interface** *interface-id*
4. **no switchport**
5. **ip address** *ip-address subnet-mask*
6. **no shutdown**
7. **exit**
8. **interface** *interface-id*
9. **no switchport**
10. **ip address** *ip-address subnet-mask*
11. **no shutdown**
12. **ip wccp** {**web-cache** | *service-number*} **redirect in**
13. **ip wccp** {**web-cache** | *service-number*} **group-listen**
14. **exit**
15. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip wccp** {**web-cache**	*service-number*} [**group-address** *groupaddress*] [**group-list** *access-list*] [**redirect-list** *access-list*] [**password** *encryption-number password*] **Example:** `Switch(config)# ip wccp web-cache`	Enables the cache service, and specifies the service number that corresponds to a dynamic service that is defined by the application engine. By default, this feature is disabled. (Optional) For **group-address** *groupaddress*, specifies the multicast group address used by the switches and the application engines to participate in the service group. (Optional) For **group-list** *access-list*, if a multicast group address is not used, specify a list of valid IP addresses that correspond to the application engines that are participating in the service group. (Optional) For **redirect-list** *access-list*, specify the redirect service for specific hosts or specific packets from hosts. (Optional) For **password** *encryption-number password*, specify an encryption number. The range is 0 to 7. Use 0 for not encrypted, and use 7 for proprietary. Specify a password name up to seven characters in length. The switch combines the password with the MD5 authentication value to create security for the connection between

	Command or Action	Purpose
		the switch and the application engine. By default, no password is configured, and no authentication is performed. You must configure the same password on each application engine. When authentication is enabled, the switch discards messages that are not authenticated.
Step 3	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Specifies the interface connected to the application engine or the server, and enters interface configuration mode.
Step 4	**no switchport** **Example:** `Switch(config-if)# no switchport`	Enters Layer 3 mode.
Step 5	**ip address** *ip-address subnet-mask* **Example:** `Switch(config-if)# ip address 172.20.10.30 255.255.255.0`	Configures the IP address and subnet mask.
Step 6	**no shutdown** **Example:** `Switch(config-if)# no shutdown`	Enables the interface.
Step 7	**exit** **Example:** `Switch(config-if)# exit`	Returns to global configuration mode. Repeat Steps 3 through 7 for each application engine and server.
Step 8	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/2`	Specifies the interface connected to the client, and enters interface configuration mode.
Step 9	**no switchport** **Example:** `Switch(config-if)# no switchport`	Enters Layer 3 mode.

	Command or Action	Purpose
Step 10	**ip address** *ip-address subnet-mask* **Example:** `Switch(config-if)# ip address 175.20.20.10 255.255.255.0`	Configures the IP address and subnet mask.
Step 11	**no shutdown** **Example:** `Switch(config-if)# no shutdown`	Enables the interface.
Step 12	**ip wccp** {**web-cache** | *service-number*} **redirect in** **Example:** `Switch(config-if)# ip wccp web-cache redirect in`	Redirects packets received from the client to the application engine. Enable this on the interface connected to the client.
Step 13	**ip wccp** {**web-cache** | *service-number*} **group-listen** **Example:** `Switch(config-if)# ip wccp web-cache group-listen`	(Optional) When using a multicast group address, the **group-listen** keyword enables the interface to listen for the multicast address. Enable this on the interface connected to the application engine.
Step 14	**exit** **Example:** `Switch(config-if)# exit`	Returns to global configuration mode. Repeat Steps 8 through 14 for each client.
Step 15	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuration Examples

This example shows how to configure routed interfaces and to enable the cache service with a multicast group address and a redirect access list. Gigabit Ethernet port 1 is connected to the application engine, is configured as a routed port with an IP address of 172.20.10.30, and is reenabled. Gigabit Ethernet port 2 is connected through the Internet to the server, is configured as a routed port with an IP address of 175.20.20.10, and is reenabled. Gigabit Ethernet ports 3 to 6 are connected to the clients and are configured as routed ports with IP addresses 175.20.30.20, 175.20.40.30, 175.20.50.40, and 175.20.60.50. The switch listens for multicast traffic and redirects packets received from the client interfaces to the application engine.

```
Switch# configure terminal
Switch(config)# ip wccp web-cache 80 group-address 224.1.1.100 redirect list 12
```

```
Switch(config)# access-list 12 permit host 10.1.1.1
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# no switchport
Switch(config-if)# ip address 172.20.10.30 255.255.255.0
Switch(config-if)# no shutdown
Switch(config-if)# ip wccp web-cache group-listen
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# no switchport
Switch(config-if)# ip address 175.20.20.10 255.255.255.0
Switch(config-if)# no shutdown
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/3
Switch(config-if)# no switchport
Switch(config-if)# ip address 175.20.30.20 255.255.255.0
Switch(config-if)# no shutdown
Switch(config-if)# ip wccp web-cache redirect in
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/4
Switch(config-if)# no switchport
Switch(config-if)# ip address 175.20.40.30 255.255.255.0
Switch(config-if)# no shutdown
Switch(config-if)# ip wccp web-cache redirect in
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/5
Switch(config-if)# no switchport
Switch(config-if)# ip address 175.20.50.40 255.255.255.0
Switch(config-if)# no shutdown
Switch(config-if)# ip wccp web-cache redirect in
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/6
Switch(config-if)# no switchport
Switch(config-if)# ip address 175.20.60.50 255.255.255.0
Switch(config-if)# no shutdown
Switch(config-if)# ip wccp web-cache redirect in
Switch(config-if)# exit
```

This example shows how to configure SVIs and how to enable the cache service with a multicast group list. VLAN 299 is created and configured with an IP address of 175.20.20.10. Gigabit Ethernet port 1 is connected through the Internet to the server and is configured as an access port in VLAN 299. VLAN 300 is created and configured with an IP address of 172.20.10.30. Gigabit Ethernet port 2 is connected to the application engine and is configured as an access port in VLAN 300. VLAN 301 is created and configured with an IP address of 175.20.30.50. Fast Ethernet ports 3 to 6, which are connected to the clients, are configured as access ports in VLAN 301. The switch redirects packets received from the client interfaces to the application engine.

Note Beginning with Cisco IOS Release 12.2(58)SE, both permit and deny ACL entries are supported in WCCP redirect lists.

```
Switch# configure terminal
Switch(config)# ip wccp web-cache 80 group-list 15
Switch(config)# access-list 15 permit host 171.69.198.102
Switch(config)# access-list 15 permit host 171.69.198.104
Switch(config)# access-list 15 permit host 171.69.198.106
Switch(config)# vlan 299
Switch(config-vlan)# exit
Switch(config)# interface vlan 299
Switch(config-if)# ip address 175.20.20.10 255.255.255.0
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 299
Switch(config)# vlan 300
Switch(config-vlan)# exit
Switch(config)# interface vlan 300
Switch(config-if)# ip address 171.69.198.100 255.255.255.0
```

```
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 300
Switch(config-if)# exit
Switch(config)# vlan 301
Switch(config-vlan)# exit
Switch(config)# interface vlan 301
Switch(config-if)# ip address 175.20.30.20 255.255.255.0
Switch(config-if)# ip wccp web-cache redirect in
Switch(config-if)# exit
Switch(config)# interface range gigabitethernet1/0/3 - 6
Switch(config-if-range)# switchport mode access
Switch(config-if-range)# switchport access vlan 301
Switch(config-if-range)# exit
```

What to Do Next

To disable the cache service, use the **no ip wccp web-cache** global configuration command. To disable inbound packet redirection, use the **no ip wccp web-cache redirect in** interface configuration command. After completing this procedure, configure the application engines in the network.

Feature History and Information for Web Cache Communication Protocol

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 81

Configuring Service Level Agreements

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Restrictions on SLAs

This section lists the restrictions on SLAs.

The following are restrictions on IP SLAs network performance measurement:

- The switch does not support VoIP service levels using the gatekeeper registration delay operations measurements.
- Only a Cisco IOS device can be a source for a destination IP SLAs responder.

- You cannot configure the IP SLAs responder on non-Cisco devices and Cisco IOS IP SLAs can send operational packets only to services native to those devices.

Related Topics

Implementing IP SLA Network Performance Measurement, on page 1703
Network Performance Measurement with Cisco IOS IP SLAs, on page 1697
IP SLA Responder and IP SLA Control Protocol, on page 1698

Information About SLAs

Cisco IOS IP Service Level Agreements (SLAs)

Cisco IOS IP SLAs send data across the network to measure performance between multiple network locations or across multiple network paths. They simulate network data and IP services and collect network performance information in real time. Cisco IOS IP SLAs generate and analyze traffic either between Cisco IOS devices or from a Cisco IOS device to a remote IP device such as a network application server. Measurements provided by the various Cisco IOS IP SLA operations can be used for troubleshooting, for problem analysis, and for designing network topologies.

Depending on the specific Cisco IOS IP SLA operations, various network performance statistics are monitored within the Cisco device and stored in both command-line interface (CLI) and Simple Network Management Protocol (SNMP) MIBs. IP SLA packets have configurable IP and application layer options such as source and destination IP address, User Datagram Protocol (UDP)/TCP port numbers, a type of service (ToS) byte (including Differentiated Services Code Point [DSCP] and IP Prefix bits), Virtual Private Network (VPN) routing/forwarding instance (VRF), and URL web address.

Because Cisco IP SLAs are Layer 2 transport independent, you can configure end-to-end operations over disparate networks to best reflect the metrics that an end user is likely to experience. IP SLAs collect a unique subset of the following performance metrics:

- Delay (both round-trip and one-way)
- Jitter (directional)
- Packet loss (directional)
- Packet sequencing (packet ordering)
- Path (per hop)
- Connectivity (directional)
- Server or website download time

Because Cisco IOS IP SLAs is SNMP-accessible, it can also be used by performance-monitoring applications like Cisco Prime Internetwork Performance Monitor (IPM) and other third-party Cisco partner performance management products.

Using IP SLAs can provide the following benefits:

- Service-level agreement monitoring, measurement, and verification.
- Network performance monitoring

- ◦ Measurement of jitter, latency, or packet loss in the network.
- ◦ Continuous, reliable, and predictable measurements.

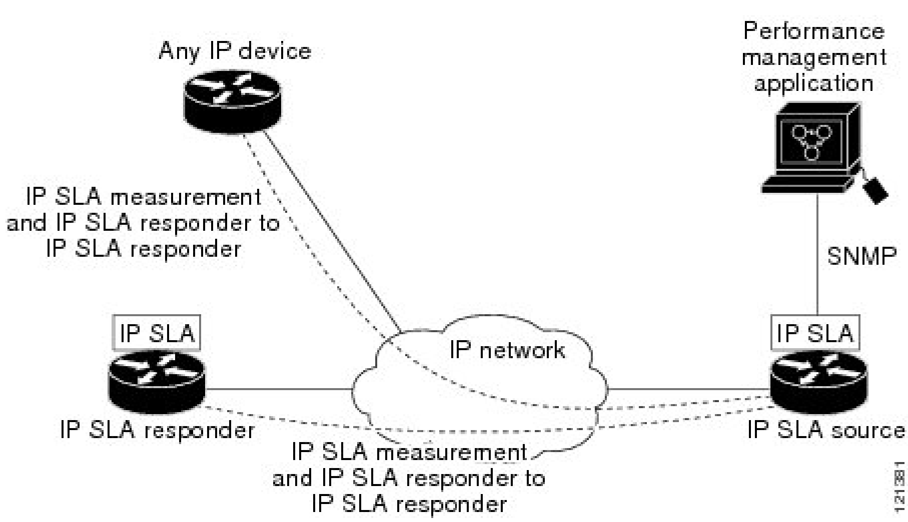
- IP service network health assessment to verify that the existing QoS is sufficient for new IP services.
- Edge-to-edge network availability monitoring for proactive verification and connectivity testing of network resources (for example, shows the network availability of an NFS server used to store business critical data from a remote site).
- Network operation troubleshooting by providing consistent, reliable measurement that immediately identifies problems and saves troubleshooting time.
- Multiprotocol Label Switching (MPLS) performance monitoring and network verification (if the switch supports MPLS).

Network Performance Measurement with Cisco IOS IP SLAs

You can use IP SLAs to monitor the performance between any area in the network—core, distribution, and edge—without deploying a physical probe. It uses generated traffic to measure network performance between two networking devices.

The following figure shows how IP SLAs begin when the source device sends a generated packet to the destination device. After the destination device receives the packet, depending on the type of IP SLAs operation, it responds with time-stamp information for the source to make the calculation on performance metrics. An IP SLAs operation performs a network measurement from the source device to a destination in the network using a specific protocol such as UDP.

Figure 75: Cisco IOS IP SLAs Operation



Related Topics

Implementing IP SLA Network Performance Measurement, on page 1703

Restrictions on SLAs, on page 1695

IP SLA Responder and IP SLA Control Protocol

The IP SLA responder is a component embedded in the destination Cisco device that allows the system to anticipate and respond to IP SLA request packets. The responder provides accurate measurements without the need for dedicated probes. The responder uses the Cisco IOS IP SLA Control Protocol to provide a mechanism through which it can be notified on which port it should listen and respond.

> **Note** The IP SLA responder can be a Cisco IOS Layer 2, responder-configurable switch. The responder does not need to support full IP SLA functionality.

The following figure shows where the Cisco IOS IP SLA responder fits in the IP network. The responder listens on a specific port for control protocol messages sent by an IP SLA operation. Upon receipt of the control message, it enables the specified UDP or TCP port for the specified duration. During this time, the responder accepts the requests and responds to them. It disables the port after it responds to the IP SLA packet, or when the specified time expires. MD5 authentication for control messages is available for added security.

You do not need to enable the responder on the destination device for all IP SLA operations. For example, a responder is not required for services that are already provided by the destination router (such as Telnet or HTTP).

Related Topics

Restrictions on SLAs, on page 1695

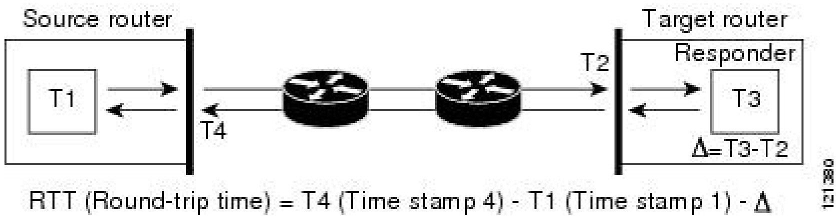
Response Time Computation for IP SLAs

Switches, controllers, and routers can take tens of milliseconds to process incoming packets due to other high priority processes. This delay affects the response times because the test-packet reply might be in a queue while waiting to be processed. In this situation, the response times would not accurately represent true network delays. IP SLAs minimize these processing delays on the source device as well as on the target device (if the responder is being used) to determine true round-trip times. IP SLA test packets use time stamping to minimize the processing delays.

When the IP SLA responder is enabled, it allows the target device to take time stamps when the packet arrives on the interface at interrupt level and again just as it is leaving, eliminating the processing time. This time stamping is made with a granularity of sub-milliseconds (ms).

The following figure demonstrates how the responder works. Four time stamps are taken to make the calculation for round-trip time. At the target router, with the responder functionality enabled, time stamp 2 (TS2) is subtracted from time stamp 3 (TS3) to produce the time spent processing the test packet as represented by delta. This delta value is then subtracted from the overall round-trip time. Notice that the same principle is applied by IP SLAs on the source router where the incoming time stamp 4 (TS4) is also taken at the interrupt level to allow for greater accuracy.

Figure 76: Cisco IOS IP SLA Responder Time Stamping



An additional benefit of the two time stamps at the target device is the ability to track one-way delay, jitter, and directional packet loss. Because much network behavior is asynchronous, it is critical to have these statistics. However, to capture one-way delay measurements, you must configure both the source router and target router with Network Time Protocol (NTP) so that the source and target are synchronized to the same clock source. One-way jitter measurements do not require clock synchronization.

IP SLAs Operation Scheduling

When you configure an IP SLAs operation, you must schedule the operation to begin capturing statistics and collecting error information. You can schedule an operation to start immediately or to start at a certain month, day, and hour. You can use the *pending* option to set the operation to start at a later time. The pending option is an internal state of the operation that is visible through SNMP. The pending state is also used when an operation is a reaction (threshold) operation waiting to be triggered. You can schedule a single IP SLAs operation or a group of operations at one time.

You can schedule several IP SLAs operations by using a single command through the Cisco IOS CLI or the CISCO RTTMON-MIB. Scheduling the operations to run at evenly distributed times allows you to control the amount of IP SLAs monitoring traffic. This distribution of IP SLA operations helps minimize the CPU utilization and thus improves network scalability.

For more details about the IP SLA multi-operations scheduling functionality, see the “IP SLAs—Multiple Operation Scheduling” chapter of the *Cisco IOS IP SLAs Configuration Guide*.

IP SLA Operation Threshold Monitoring

To support successful service level agreement monitoring, you must have mechanisms that notify you immediately of any possible violation. IP SLAs can send SNMP traps that are triggered by events such as the following:

- Connection loss
- Timeout
- Round-trip time threshold
- Average jitter threshold
- One-way packet loss
- One-way jitter
- One-way mean opinion score (MOS)

- One-way latency

An IP SLA threshold violation can also trigger another IP SLA operation for further analysis. For example, the frequency could be increased or an Internet Control Message Protocol (ICMP) path echo or ICMP path jitter operation could be initiated for troubleshooting.

ICMP Echo

The ICMP echo operation measures the end-to-end response time between a Cisco device and any other device that uses IP. The response time is computed by measuring the time it takes to send an ICMP echo request message to a destination and receive an ICMP echo reply. Many customers use IP SLA ICMP-based operations, in-house ping testing, or ping-based dedicated probes to measure this response time. The IP SLA ICMP echo operation conforms to the same specifications as ICMP ping testing, and both methods result in the same response times.

Related Topics

Analyzing IP Service Levels by Using the ICMP Echo Operation, on page 1709

UDP Jitter

Jitter is a simple term that describes interpacket delay variance. When multiple packets are sent consecutively at an interval of 10 ms from source to destination, the destination should receive them 10 ms apart (if the network is behaving correctly). However, if there are delays in the network (such as queuing, arriving through alternate routes, and so on), the time interval between packet arrivals might be more or less than 10 ms. A positive jitter value indicates that the packets arrived more than 10 ms apart. A negative jitter value indicates that the packets arrived less than 10 ms apart. If the packets arrive 12 ms apart, the positive jitter is 2 ms; if the packets arrive 8 ms apart, the negative jitter is 2 ms. For delay-sensitive networks, positive jitter values are undesirable, and a jitter value of 0 is ideal.

In addition to monitoring jitter, the IP SLA UDP jitter operation can be used as a multipurpose data gathering operation. The packets generated by IP SLAs carry sequence information and time stamps from the source and operational target that include packet sending and receiving data. Based on this data, UDP jitter operations measure the following:

- Per-direction jitter (source to destination and destination to source)
- Per-direction packet-loss
- Per-direction delay (one-way delay)
- Round-trip delay (average round-trip time)

Because the paths for the sending and receiving of data can be different (asymmetric), you can use the per-direction data to more readily identify where congestion or other problems are occurring in the network.

The UDP jitter operation generates synthetic (simulated) UDP traffic and sends a number of UDP packets, each of a specified size, sent a specified number of milliseconds apart, from a source router to a target router, at a given frequency. By default, ten packet-frames, each with a payload size of 10 bytes are generated every 10 ms, and the operation is repeated every 60 seconds. You can configure each of these parameters to best simulate the IP service you want to provide.

To provide accurate one-way delay (latency) measurements, time synchronization (as provided by NTP) is required between the source and the target device. Time synchronization is not required for the one-way jitter and packet loss measurements. If the time is not synchronized between the source and target devices, one-way jitter and packet loss data is returned, but values of 0 are returned for the one-way delay measurements provided by the UDP jitter operation.

Related Topics

Analyzing IP Service Levels by Using the UDP Jitter Operation, on page 1706

Configuration Guidelines

For information on the IP SLA commands, see the *Cisco IOS IP SLAs Command Reference, Release 12.4T* command reference.

For detailed descriptions and configuration procedures, see the *Cisco IOS IP SLAs Configuration Guide, Release 12.4TL*.

Not all of the IP SLA commands or operations described in the referenced guide are supported on the switch. The switch supports IP service level analysis by using UDP jitter, UDP echo, HTTP, TCP connect, ICMP echo, ICMP path echo, ICMP path jitter, FTP, DNS, and DHCP, as well as multiple operation scheduling and proactive threshold monitoring. It does not support VoIP service levels using the gatekeeper registration delay operations measurements.

Before configuring any IP SLAs application, you can use the **show ip sla application** privileged EXEC command to verify that the operation type is supported on your software image. This is an example of the output from the command:

```
Switch# show ip sla application

        IP Service Level Agreements
Version: Round Trip Time MIB 2.2.0, Infrastructure Engine-III

Supported Operation Types:
        icmpEcho, path-echo, path-jitter, udpEcho, tcpConnect, http
        dns, udpJitter, dhcp, ftp, udpApp, wspApp

Supported Features:
        IPSLAs Event Publisher

IP SLAs low memory water mark: 33299323
Estimated system max number of entries: 24389

Estimated number of configurable operations: 24389
Number of Entries configured  : 0
Number of active Entries      : 0
Number of pending Entries     : 0
Number of inactive Entries    : 0
Time of last change in whole IP SLAs: *13:04:37.668 UTC Wed Dec 19 2012
```

How to Configure IP SLAs Operations

This section does not include configuration information for all available operations as the configuration information details are included in the *Cisco IOS IP SLAs Configuration Guide*. It does include several operations as examples, including configuring the responder, configuring a UDP jitter operation, which requires a responder, and configuring an ICMP echo operation, which does not require a responder. For details about configuring other operations, see the *Cisco IOS IP SLAs Configuration Guide*.

Configuring the IP SLA Responder

The IP SLA responder is available only on Cisco IOS software-based devices, including some Layer 2 switches that do not support full IP SLA functionality.

Beginning in privileged EXEC mode, follow these steps to configure the IP SLA responder on the target device (the operational target):

SUMMARY STEPS

1. **configure terminal**
2. **ip sla responder** {**tcp-connect** | **udp-echo**} **ipaddress** *ip-address* **port** *port-number*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip sla responder** {**tcp-connect** | **udp-echo**} **ipaddress** *ip-address* **port** *port-number* **Example:** `Switch(config)# ip sla responder udp-echo 172.29.139.134 5000`	Configures the switch as an IP SLA responder. The keywords have these meanings: • **tcp-connect**—Enables the responder for TCP connect operations. • **udp-echo**—Enables the responder for User Datagram Protocol (UDP) echo or jitter operations. • **ipaddress** *ip-address*—Enter the destination IP address. • **port** *port-number*—Enter the destination port number. **Note** The IP address and port number must match those configured on the source device for the IP SLA operation.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

UDP Jitter Example

This example shows how to configure the device as a responder for the UDP jitter IP SLA operation in the next procedure:

```
Switch(config)# ip sla responder udp-echo 172.29.139.134 5000
```

Implementing IP SLA Network Performance Measurement

Beginning in privileged EXEC mode, follow these steps to implement IP SLA network performance measurement on your switch:

Before You Begin

Use the **show ip sla application** privileged EXEC command to verify that the desired operation type is supported on your software image.

SUMMARY STEPS

1. **configure terminal**
2. **ip sla** *operation-number*
3. **udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] [**num-packets** *number-of-packets*] [**interval** *interpacket-interval*]
4. **frequency** *seconds*
5. **threshold** *milliseconds*
6. **exit**
7. **ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [:*ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**]
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip sla** *operation-number* **Example:** `Switch(config)# ip sla 10`	Creates an IP SLA operation, and enters IP SLA configuration mode.

	Command or Action	Purpose
Step 3	**udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] [**num-packets** *number-of-packets*] [**interval** *interpacket-interval*] **Example:** `Switch(config-ip-sla)# udp-jitter 172.29.139.134 5000`	Configures the IP SLA operation as the operation type of your choice (a UDP jitter operation is used in the example), and enters its configuration mode (UDP jitter configuration mode is used in the example). • *destination-ip-address* | *destination-hostname*—Specifies the destination IP address or hostname. • *destination-port*—Specifies the destination port number in the range from 1 to 65535. • (Optional) **source-ip** {*ip-address* | *hostname*}—Specifies the source IP address or hostname. When a source IP address or hostname is not specified, IP SLA chooses the IP address nearest to the destination • (Optional) **source-port** *port-number*—Specifies the source port number in the range from 1 to 65535. When a port number is not specified, IP SLA chooses an available port. • (Optional) **control**—Enables or disables sending of IP SLA control messages to the IP SLA responder. By default, IP SLA control messages are sent to the destination device to establish a connection with the IP SLA responder • (Optional) **num-packets** *number-of-packets*—Enters the number of packets to be generated. The range is 1 to 6000; the default is 10. • (Optional) **interval** *inter-packet-interval*—Enters the interval between sending packets in milliseconds. The range is 1 to 6000; the default value is 20 ms.
Step 4	**frequency** *seconds* **Example:** `Switch(config-ip-sla-jitter)# frequency 45`	(Optional) Configures options for the SLA operation. This example sets the rate at which a specified IP SLA operation repeats. The range is from 1 to 604800 seconds; the default is 60 seconds.
Step 5	**threshold** *milliseconds* **Example:** `Switch(config-ip-sla-jitter)# threshold 200`	(Optional) Configures threshold conditions. This example sets the threshold of the specified IP SLA operation to 200. The range is from 0 to 60000 milliseconds.
Step 6	**exit** **Example:** `Switch(config-ip-sla-jitter)# exit`	Exits the SLA operation configuration mode (UDP jitter configuration mode in this example), and returns to global configuration mode.
Step 7	**ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm*	Configures the scheduling parameters for an individual IP SLA operation.

	Command or Action	Purpose
	[:*ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**] **Example:** `Switch(config)# ip sla schedule 10 start-time now life forever`	• *operation-number*—Enter the RTR entry number. • (Optional) **life**—Sets the operation to run indefinitely (**forever**) or for a specific number of *seconds*. The range is from 0 to 2147483647. The default is 3600 seconds (1 hour). • (Optional) **start-time**—Enters the time for the operation to begin collecting information: To start at a specific time, enter the hour, minute, second (in 24-hour notation), and day of the month. If no month is entered, the default is the current month. Enter **pending** to select no information collection until a start time is selected. Enter **now** to start the operation immediately. Enter **after** *hh:mm:ss* to show that the operation should start after the entered time has elapsed. • (Optional) **ageout** *seconds*—Enter the number of seconds to keep the operation in memory when it is not actively collecting information. The range is 0 to 2073600 seconds, the default is 0 seconds (never ages out). • (Optional) **recurring**—Set the operation to automatically run every day.
Step 8	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

UDP Jitter Configuration

This example shows how to configure a UDP jitter IP SLA operation:

```
Switch(config)# ip sla 10
Switch(config-ip-sla)# udp-jitter 172.29.139.134 5000
Switch(config-ip-sla-jitter)# frequency 30
Switch(config-ip-sla-jitter)# exit
Switch(config)# ip sla schedule 5 start-time now life forever
Switch(config)# end
Switch# show ip sla configuration 10
IP SLAs, Infrastructure Engine-II.

Entry number: 10
Owner:
Tag:
Type of operation to perform: udp-jitter
Target address/Source address: 1.1.1.1/0.0.0.0
Target port/Source port: 2/0
Request size (ARR data portion): 32
Operation timeout (milliseconds): 5000
Packet Interval (milliseconds)/Number of packets: 20/10
```

```
Type Of Service parameters: 0x0
Verify data: No
Vrf Name:
Control Packets: enabled
Schedule:
    Operation frequency (seconds): 30
    Next Scheduled Start Time: Pending trigger
    Group Scheduled : FALSE
    Randomly Scheduled : FALSE
    Life (seconds): 3600
    Entry Ageout (seconds): never
    Recurring (Starting Everyday): FALSE
    Status of entry (SNMP RowStatus): notInService
Threshold (milliseconds): 5000
Distribution Statistics:
    Number of statistic hours kept: 2
    Number of statistic distribution buckets kept: 1
    Statistic distribution interval (milliseconds): 20
Enhanced History:
```

Related Topics

Network Performance Measurement with Cisco IOS IP SLAs, on page 1697

Restrictions on SLAs, on page 1695

Analyzing IP Service Levels by Using the UDP Jitter Operation

Beginning in privileged EXEC mode, follow these steps to configure a UDP jitter operation on the source device:

Before You Begin

You must enable the IP SLA responder on the target device (the operational target) to configure a UDP jitter operation on the source device.

SUMMARY STEPS

1. **configure terminal**
2. **ip sla** *operation-number*
3. **udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] [**num-packets** *number-of-packets*] [**interval** *interpacket-interval*]
4. **frequency** *seconds*
5. **exit**
6. **ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [:*ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**]
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip sla** *operation-number* **Example:** `Switch(config)# ip sla 10`	Creates an IP SLA operation, and enters IP SLA configuration mode.
Step 3	**udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] [**num-packets** *number-of-packets*] [**interval** *interpacket-interval*] **Example:** `Switch(config-ip-sla)# udp-jitter 172.29.139.134 5000`	Configures the IP SLA operation as a UDP jitter operation, and enters UDP jitter configuration mode. • *destination-ip-address* | *destination-hostname*—Specifies the destination IP address or hostname. • *destination-port*—Specifies the destination port number in the range from 1 to 65535. • (Optional) **source-ip** {*ip-address* | *hostname*}—Specifies the source IP address or hostname. When a source IP address or hostname is not specified, IP SLA chooses the IP address nearest to the destination. • (Optional) **source-port** *port-number*—Specifies the source port number in the range from 1 to 65535. When a port number is not specified, IP SLA chooses an available port. • (Optional) **control**—Enables or disables sending of IP SLA control messages to the IP SLA responder. By default, IP SLA control messages are sent to the destination device to establish a connection with the IP SLA responder. • (Optional) **num-packets** *number-of-packets*—Enters the number of packets to be generated. The range is 1 to 6000; the default is 10. • (Optional) **interval** *inter-packet-interval*—Enters the interval between sending packets in milliseconds. The range is 1 to 6000; the default value is 20 ms.
Step 4	**frequency** *seconds* **Example:** `Switch(config-ip-sla-jitter)# frequency 45`	(Optional) Sets the rate at which a specified IP SLA operation repeats. The range is from 1 to 604800 seconds; the default is 60 seconds.

	Command or Action	Purpose
Step 5	**exit** **Example:** `Switch(config-ip-sla-jitter)# exit`	Exits UDP jitter configuration mode, and returns to global configuration mode.
Step 6	**ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [:*ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**] **Example:** `Switch(config)# ip sla schedule 10 start-time now life forever`	Configures the scheduling parameters for an individual IP SLA operation. • *operation-number*—Enter the RTR entry number. • (Optional) **life**—Sets the operation to run indefinitely (**forever**) or for a specific number of *seconds*. The range is from 0 to 2147483647. The default is 3600 seconds (1 hour). • (Optional) **start-time**—Enters the time for the operation to begin collecting information: To start at a specific time, enter the hour, minute, second (in 24-hour notation), and day of the month. If no month is entered, the default is the current month. Enter **pending** to select no information collection until a start time is selected. Enter **now** to start the operation immediately. Enter **after** *hh:mm:ss* to show that the operation should start after the entered time has elapsed. • (Optional) **ageout** *seconds*—Enter the number of seconds to keep the operation in memory when it is not actively collecting information. The range is 0 to 2073600 seconds, the default is 0 seconds (never ages out). • (Optional) **recurring**—Set the operation to automatically run every day.
Step 7	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring a UDP Jitter IP SLA Operation

This example shows how to configure a UDP jitter IP SLA operation:

```
Switch(config)# ip sla 10
Switch(config-ip-sla)# udp-jitter 172.29.139.134 5000
Switch(config-ip-sla-jitter)# frequency 30
Switch(config-ip-sla-jitter)# exit
Switch(config)# ip sla schedule 5 start-time now life forever
Switch(config)# end
Switch# show ip sla configuration 10
```

```
IP SLAs, Infrastructure Engine-II.

Entry number: 10
Owner:
Tag:
Type of operation to perform: udp-jitter
Target address/Source address: 1.1.1.1/0.0.0.0
Target port/Source port: 2/0
Request size (ARR data portion): 32
Operation timeout (milliseconds): 5000
Packet Interval (milliseconds)/Number of packets: 20/10
Type Of Service parameters: 0x0
Verify data: No
Vrf Name:
Control Packets: enabled
Schedule:
    Operation frequency (seconds): 30
    Next Scheduled Start Time: Pending trigger
    Group Scheduled : FALSE
    Randomly Scheduled : FALSE
    Life (seconds): 3600
    Entry Ageout (seconds): never
    Recurring (Starting Everyday): FALSE
    Status of entry (SNMP RowStatus): notInService
Threshold (milliseconds): 5000
Distribution Statistics:
    Number of statistic hours kept: 2
    Number of statistic distribution buckets kept: 1
    Statistic distribution interval (milliseconds): 20
Enhanced History:
```

Related Topics

UDP Jitter, on page 1700

Analyzing IP Service Levels by Using the ICMP Echo Operation

Beginning in privileged EXEC mode, follow these steps to configure an ICMP echo operation on the source device:

Before You Begin

This operation does not require the IP SLA responder to be enabled.

SUMMARY STEPS

1. **configure terminal**
2. **ip sla** *operation-number*
3. **icmp-echo** {*destination-ip-address* | *destination-hostname*} [**source-ip** {*ip-address* | *hostname*} | **source-interface** *interface-id*]
4. **frequency** *seconds*
5. **exit**
6. **ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [*:ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**]
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**ip sla** *operation-number* **Example:** `Switch(config)# ip sla 10`	Creates an IP SLA operation and enters IP SLA configuration mode.
Step 3	**icmp-echo** {*destination-ip-address* | *destination-hostname*} [**source-ip** {*ip-address* | *hostname*} | **source-interface** *interface-id*] **Example:** `Switch(config-ip-sla)# icmp-echo 172.29.139.134`	Configures the IP SLA operation as an ICMP Echo operation and enters ICMP echo configuration mode. • *destination-ip-address* | *destination-hostname*—Specifies the destination IP address or hostname. • (Optional) **source-ip** {*ip-address* | *hostname*}—Specifies the source IP address or hostname. When a source IP address or hostname is not specified, IP SLA chooses the IP address nearest to the destination. • (Optional) **source-interface** *interface-id*—Specifies the source interface for the operation.
Step 4	**frequency** *seconds* **Example:** `Switch(config-ip-sla-echo)# frequency 30`	(Optional) Sets the rate at which a specified IP SLA operation repeats. The range is from 1 to 604800 seconds; the default is 60 seconds.
Step 5	**exit** **Example:** `Switch(config-ip-sla-echo)# exit`	Exits UDP echo configuration mode, and returns to global configuration mode.
Step 6	**ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [*:ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**] **Example:** `Switch(config)# ip sla schedule 5 start-time now life forever`	Configures the scheduling parameters for an individual IP SLA operation. • *operation-number*—Enter the RTR entry number. • (Optional) **life**—Sets the operation to run indefinitely (**forever**) or for a specific number of *seconds*. The range is from 0 to 2147483647. The default is 3600 seconds (1 hour) • (Optional) **start-time**—Enter the time for the operation to begin collecting information:

	Command or Action	Purpose
		To start at a specific time, enter the hour, minute, second (in 24-hour notation), and day of the month. If no month is entered, the default is the current month. Enter **pending** to select no information collection until a start time is selected. Enter **now** to start the operation immediately. Enter **after** *hh:mm:ss* to indicate that the operation should start after the entered time has elapsed. • (Optional) **ageout** *seconds*—Enter the number of seconds to keep the operation in memory when it is not actively collecting information. The range is 0 to 2073600 seconds; the default is 0 seconds (never ages out). • (Optional) **recurring—**Sets the operation to automatically run every day.
Step 7	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring an ICMP Echo IP SLA Operation

This example shows how to configure an ICMP echo IP SLA operation:

```
Switch(config)# ip sla 12
Switch(config-ip-sla)# icmp-echo 172.29.139.134
Switch(config-ip-sla-echo)# frequency 30
Switch(config-ip-sla-echo)# exit
Switch(config)# ip sla schedule 5 start-time now life forever
Switch(config)# end
Switch# show ip sla configuration 22
IP SLAs, Infrastructure Engine-II.

Entry number: 12
Owner:
Tag:
Type of operation to perform: echo
Target address: 2.2.2.2
Source address: 0.0.0.0
Request size (ARR data portion): 28
Operation timeout (milliseconds): 5000
Type Of Service parameters: 0x0
Verify data: No
Vrf Name:
Schedule:
    Operation frequency (seconds): 60
    Next Scheduled Start Time: Pending trigger
    Group Scheduled : FALSE
    Randomly Scheduled : FALSE
    Life (seconds): 3600
```

```
    Entry Ageout (seconds): never
    Recurring (Starting Everyday): FALSE
    Status of entry (SNMP RowStatus): notInService
Threshold (milliseconds): 5000
Distribution Statistics:
    Number of statistic hours kept: 2
    Number of statistic distribution buckets kept: 1
    Statistic distribution interval (milliseconds): 20
History Statistics:
    Number of history Lives kept: 0
    Number of history Buckets kept: 15
    History Filter Type: None
Enhanced History:
```

Related Topics

IP SLA Operation Threshold Monitoring, on page 1699

Monitoring IP SLA Operations

The following table describes the commands used to display IP SLA operation configurations and results:

Table 175: Monitoring IP SLA Operations

show ip sla application	Displays global information about Cisco IOS IP SLAs.
show ip sla authentication	Displays IP SLA authentication information.
show ip sla configuration [*entry-number*]	Displays configuration values including all defaults for all IP SLA operations or a specific operation.
show ip sla enhanced-history {**collection-statistics** | **distribution statistics**} [*entry-number*]	Displays enhanced history statistics for collected history buckets or distribution statistics for all IP SLA operations or a specific operation.
show ip sla ethernet-monitor configuration [*entry-number*]	Displays IP SLA automatic Ethernet configuration.
show ip sla group schedule [*schedule-entry-number*]	Displays IP SLA group scheduling configuration and details.
show ip sla history [*entry-number* | **full** | **tabular**]	Displays history collected for all IP SLA operations.
show ip sla mpls-lsp-monitor {**collection-statistics** | **configuration** | **ldp operational-state** | **scan-queue** | **summary** [*entry-number*] | **neighbors**}	Displays MPLS label switched path (LSP) Health Monitor operations.

show ip sla reaction-configuration [*entry-number*]	Displays the configured proactive threshold monitoring settings for all IP SLA operations or a specific operation.
show ip sla reaction-trigger [*entry-number*]	Displays the reaction trigger information for all IP SLA operations or a specific operation.
show ip sla responder	Displays information about the IP SLA responder.
show ip sla statistics [*entry-number* | **aggregated** | **details**]	Displays current or aggregated operational status and statistics.

Monitoring IP SLA Operation Examples

The following example shows all IP SLAs by application:

```
Switch# show ip sla application

        IP Service Level Agreements
Version: Round Trip Time MIB 2.2.0, Infrastructure Engine-III

Supported Operation Types:
        icmpEcho, path-echo, path-jitter, udpEcho, tcpConnect, http
        dns, udpJitter, dhcp, ftp, udpApp, wspApp

Supported Features:
        IPSLAs Event Publisher

IP SLAs low memory water mark: 33299323
Estimated system max number of entries: 24389

Estimated number of configurable operations: 24389
Number of Entries configured  : 0
Number of active Entries      : 0
Number of pending Entries     : 0
Number of inactive Entries    : 0
Time of last change in whole IP SLAs: *13:04:37.668 UTC Wed Dec 19 2012
```

The following example shows all IP SLA distribution statistics:

```
Switch# show ip sla enhanced-history distribution-statistics

Point by point Enhanced History
Entry    = Entry Number
Int      = Aggregation Interval
BucI     = Bucket Index
StartT   = Aggregation Start Time
Pth      = Path index
Hop      = Hop in path index
Comps    = Operations completed
OvrTh    = Operations completed over thresholds
SumCmp   = Sum of RTT (milliseconds)
SumCmp2L = Sum of RTT squared low 32 bits (milliseconds)
SumCmp2H = Sum of RTT squared high 32 bits (milliseconds)
TMax     = RTT maximum (milliseconds)
TMin     = RTT minimum (milliseconds)

Entry Int BucI StartT     Pth Hop Comps OvrTh SumCmp     SumCmp2L   SumCmp2H   T
Max    TMin
```

Feature History and Information for Service Level Agreements

Release	Modification
Cisco IOS XE 3.2SE	This feature was introduced.

CHAPTER 82

Configuring SPAN and RSPAN

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Related Topics

Feature History and Information for Troubleshooting Software Configuration, on page 235

Prerequisites for SPAN and RSPAN

SPAN

- You can limit SPAN traffic to specific VLANs by using the **filter vlan** keyword. If a trunk port is being monitored, only traffic on the VLANs specified with this keyword is monitored. By default, all VLANs are monitored on a trunk port.

RSPAN

- We recommend that you configure an RSPAN VLAN before you configure an RSPAN source or a destination session.

Restrictions for SPAN and RSPAN

SPAN

- On each switch, you can configure a maximum of 8 source sessions and 58 RSPAN destination sessions. A source session is either a local SPAN session or an RSPAN source session.
- For SPAN sources, you can monitor traffic for a single port or VLAN or a series or range of ports or VLANs for each session. You cannot mix source ports and source VLANs within a single SPAN session.
- The destination port cannot be a source port; a source port cannot be a destination port.
- You cannot have two SPAN sessions using the same destination port.
- When you configure a switch port as a SPAN destination port, it is no longer a normal switch port; only monitored traffic passes through the SPAN destination port.
- Entering SPAN configuration commands does not remove previously configured SPAN parameters. You must enter the **no monitor session** {*session_number* **|** **all** **|** **local** **|** **remote**} global configuration command to delete configured SPAN parameters.
- For local SPAN, outgoing packets through the SPAN destination port carry the original encapsulation headers**—**untagged or IEEE 802.1Q**—**if the **encapsulation replicate** keywords are specified. If the keywords are not specified, the packets are sent in native form.
- You can configure a disabled port to be a source or destination port, but the SPAN function does not start until the destination port and at least one source port or source VLAN are enabled.
- You cannot mix source VLANs and filter VLANs within a single SPAN session.
- Stacking is supported only on switches running the LAN Base image.

Traffic monitoring in a SPAN session has the following restrictions:

- Sources can be ports or VLANs, but you cannot mix source ports and source VLANs in the same session.
- The switch supports up to two local SPAN or RSPAN source sessions.
    - You can run both a local SPAN and an RSPAN source session in the same switch or switch stack. The switch or switch stack supports a total of 66 source and RSPAN destination sessions.
    - You can configure two separate SPAN or RSPAN source sessions with separate or overlapping sets of SPAN source ports and VLANs. Both switched and routed ports can be configured as SPAN sources and destinations.
- You can have multiple destination ports in a SPAN session, but no more than 66 destination ports per switch stack.
- SPAN sessions do not interfere with the normal operation of the switch. However, an oversubscribed SPAN destination, for example, a 10-Mb/s port monitoring a 100-Mb/s port, can result in dropped or lost packets.

- When SPAN or RSPAN is enabled, each packet being monitored is sent twice, once as normal traffic and once as a monitored packet. Monitoring a large number of ports or VLANs could potentially generate large amounts of network traffic.
- You can configure SPAN sessions on disabled ports; however, a SPAN session does not become active unless you enable the destination port and at least one source port or VLAN for that session.
- The switch does not support a combination of local SPAN and RSPAN in a single session.
    - An RSPAN source session cannot have a local destination port.
    - An RSPAN destination session cannot have a local source port.
    - An RSPAN destination session and an RSPAN source session that are using the same RSPAN VLAN cannot run on the same switch or switch stack.

RSPAN

- RSPAN does not support BPDU packet monitoring or other Layer 2 switch protocols.
- The RSPAN VLAN is configured only on trunk ports and not on access ports. To avoid unwanted traffic in RSPAN VLANs, make sure that the VLAN remote-span feature is supported in all the participating switches.
- RSPAN VLANs are included as sources for port-based RSPAN sessions when source trunk ports have active RSPAN VLANs. RSPAN VLANs can also be sources in SPAN sessions. However, since the switch does not monitor spanned traffic, it does not support egress spanning of packets on any RSPAN VLAN identified as the destination of an RSPAN source session on the switch.
- If you enable VTP and VTP pruning, RSPAN traffic is pruned in the trunks to prevent the unwanted flooding of RSPAN traffic across the network for VLAN IDs that are lower than 1005.
- To use RSPAN, the switch must be running the LAN Base image.

Flow-Based SPAN (FSPAN) and Flow-Based RSPAN (FRSPAN)

- More than one SPAN or RSPAN session can have ACLs attached. However, the size of the ACL should not exceed the size of the TCAM memory allocated to FSPAN.
- When no FSPAN ACLs are attached, FSPAN is disabled, and all traffic is copied to the SPAN destination ports.
- When you attach an empty FSPAN ACL to a SPAN session, it does not filter packets, and all traffic is monitored.
- FSPAN ACLs cannot be applied to per-port-per-VLAN sessions. You can configure per-port-per-VLAN sessions by first configuring a port-based session and then configuring specific VLANs to the session. For example:

```
Switch(config)# monitor session session_number source interface interface-id
Switch(config)# monitor session session_number filter vlan vlan-id
Switch(config)# monitor session session_number filter ip access-group {access-list-number|
name}
```

Note Both the **filter vlan** and **filter ip access-group** commands cannot be configured at the same time. Configuring one results in rejection of the other.

- EtherChannels are not supported in an FSPAN session.
- FSPAN ACLs with TCP flags or the **log** keyword are not supported.
- If you configure an IPv6 FSPAN ACL when the switch is running the advanced IP Services feature set but later run a different feature set, after rebooting the switch, the switch might lose the IPv6 FSPAN ACL configuration.
- IPv6 FSPAN ACLs are supported only on IPv6-enabled SDM templates. If you configure an IPv6 FSPAN ACL when running an IPv6 enabled SDM template, but later configure a non-IPv6 SDM template and reboot the switch, you lose the IPv6 FSPAN ACL configuration.

Information About SPAN and RSPAN

SPAN and RSPAN

You can analyze network traffic passing through ports or VLANs by using SPAN or RSPAN to send a copy of the traffic to another port on the switch or on another switch that has been connected to a network analyzer or other monitoring or security device. SPAN copies (or mirrors) traffic received or sent (or both) on source ports or source VLANs to a destination port for analysis. SPAN does not affect the switching of network traffic on the source ports or VLANs. You must dedicate the destination port for SPAN use. Except for traffic that is required for the SPAN or RSPAN session, destination ports do not receive or forward traffic.

Only traffic that enters or leaves source ports or traffic that enters or leaves source VLANs can be monitored by using SPAN; traffic routed to a source VLAN cannot be monitored. For example, if incoming traffic is being monitored, traffic that gets routed from another VLAN to the source VLAN cannot be monitored; however, traffic that is received on the source VLAN and routed to another VLAN can be monitored.

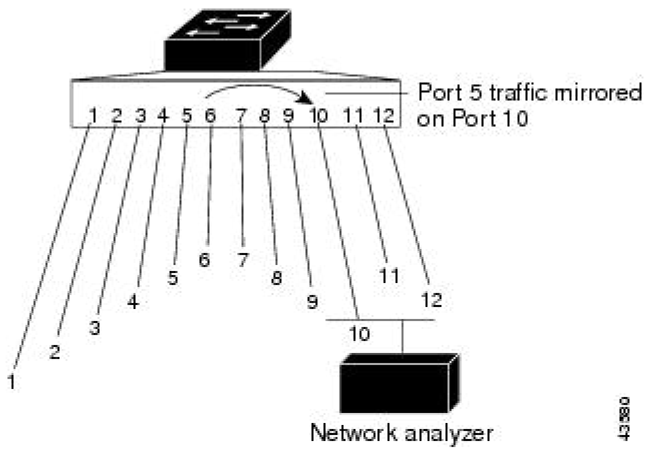
You can use the SPAN or RSPAN destination port to inject traffic from a network security device. For example, if you connect a Cisco Intrusion Detection System (IDS) sensor appliance to a destination port, the IDS device can send TCP reset packets to close down the TCP session of a suspected attacker.

Local SPAN

Local SPAN supports a SPAN session entirely within one switch; all source ports or source VLANs and destination ports are in the same switch or switch stack. Local SPAN copies traffic from one or more source ports in any VLAN or from one or more VLANs to a destination port for analysis.

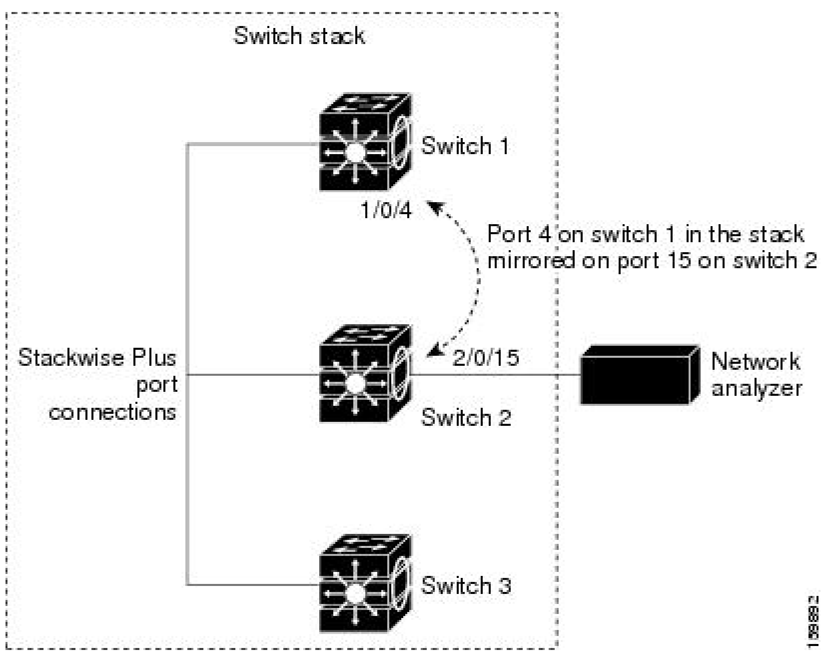
All traffic on port 5 (the source port) is mirrored to port 10 (the destination port). A network analyzer on port 10 receives all network traffic from port 5 without being physically attached to port 5.

Figure 77: Example of Local SPAN Configuration on a Single Device



This is an example of a local SPAN in a switch stack, where the source and destination ports reside on different stack members.

Figure 78: Example of Local SPAN Configuration on a Device Stack

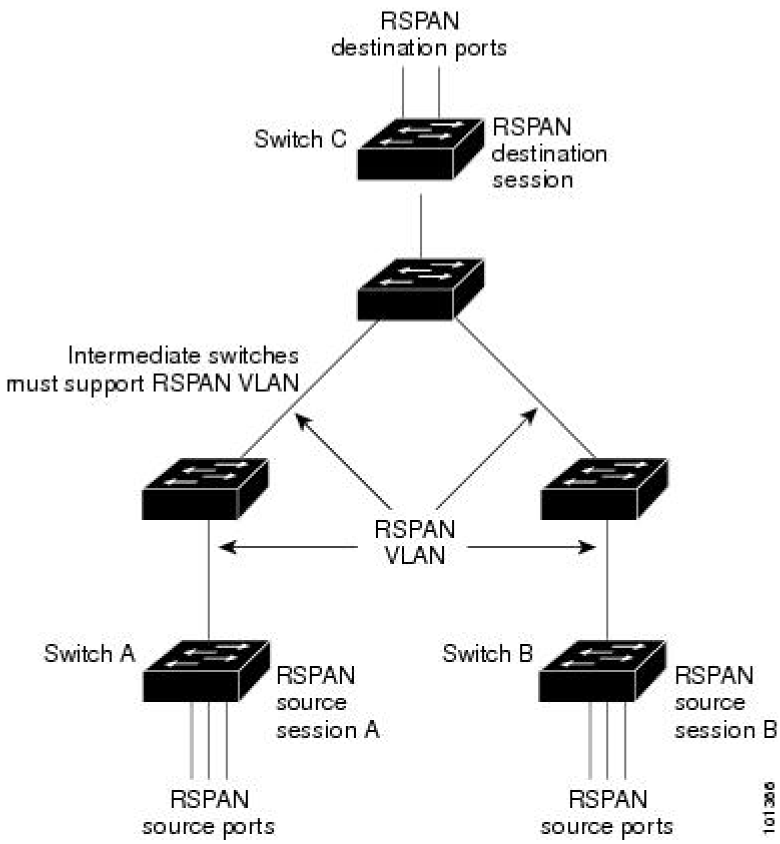


Remote SPAN

RSPAN supports source ports, source VLANs, and destination ports on different switches (or different switch stacks), enabling remote monitoring of multiple switches across your network.

The figure below shows source ports on Switch A and Switch B. The traffic for each RSPAN session is carried over a user-specified RSPAN VLAN that is dedicated for that RSPAN session in all participating switches. The RSPAN traffic from the source ports or VLANs is copied into the RSPAN VLAN and forwarded over trunk ports carrying the RSPAN VLAN to a destination session monitoring the RSPAN VLAN. Each RSPAN source switch must have either ports or VLANs as RSPAN sources. The destination is always a physical port, as shown on Switch C in the figure.

Figure 79: Example of RSPAN Configuration



SPAN and RSPAN Concepts and Terminology

- SPAN Sessions
- Monitored Traffic
- Source Ports
- Source VLANs
- VLAN Filtering
- Destination Port
- RSPAN VLAN

SPAN Sessions

SPAN sessions (local or remote) allow you to monitor traffic on one or more ports, or one or more VLANs, and send the monitored traffic to one or more destination ports.

A local SPAN session is an association of a destination port with source ports or source VLANs, all on a single network device. Local SPAN does not have separate source and destination sessions. Local SPAN sessions gather a set of ingress and egress packets specified by the user and form them into a stream of SPAN data, which is directed to the destination port.

RSPAN consists of at least one RSPAN source session, an RSPAN VLAN, and at least one RSPAN destination session. You separately configure RSPAN source sessions and RSPAN destination sessions on different network devices. To configure an RSPAN source session on a device, you associate a set of source ports or source VLANs with an RSPAN VLAN. The output of this session is the stream of SPAN packets that are sent to the RSPAN VLAN. To configure an RSPAN destination session on another device, you associate the destination port with the RSPAN VLAN. The destination session collects all RSPAN VLAN traffic and sends it out the RSPAN destination port.

An RSPAN source session is very similar to a local SPAN session, except for where the packet stream is directed. In an RSPAN source session, SPAN packets are relabeled with the RSPAN VLAN ID and directed over normal trunk ports to the destination switch.

An RSPAN destination session takes all packets received on the RSPAN VLAN, strips off the VLAN tagging, and presents them on the destination port. The session presents a copy of all RSPAN VLAN packets (except Layer 2 control packets) to the user for analysis.

A single RSPAN session with multiple source and destination ports can be in the same session but more than one source session with the source being the same remote vlan is not allowed.

Traffic monitoring in a SPAN session has these restrictions:

- Sources can be ports or VLANs, but you cannot mix source ports and source VLANs in the same session.
- The switch supports up to two local SPAN or RSPAN source sessions.
    - You can run both a local SPAN and an RSPAN source session in the same switch or switch stack. The switch or switch stack supports a total of 64 source and RSPAN destination sessions.
    - You can configure two separate SPAN or RSPAN source sessions with separate or overlapping sets of SPAN source ports and VLANs. Both switched and routed ports can be configured as SPAN sources and destinations.
- You can have multiple destination ports in a SPAN session, but no more than 64 destination ports per switch stack.
- SPAN sessions do not interfere with the normal operation of the switch. However, an oversubscribed SPAN destination, for example, a 10-Mb/s port monitoring a 100-Mb/s port, can result in dropped or lost packets.
- When SPAN or RSPAN is enabled, each packet being monitored is sent twice, once as normal traffic and once as a monitored packet. Therefore monitoring a large number of ports or VLANs could potentially generate large amounts of network traffic.
- You can configure SPAN sessions on disabled ports; however, a SPAN session does not become active unless you enable the destination port and at least one source port or VLAN for that session.
- The switch does not support a combination of local SPAN and RSPAN in a single session.
    - An RSPAN source session cannot have a local destination port.

◦ An RSPAN destination session cannot have a local source port.

◦ An RSPAN destination session and an RSPAN source session that are using the same RSPAN VLAN cannot run on the same switch or switch stack.

Monitored Traffic

SPAN sessions can monitor these traffic types:

- Receive (Rx) SPAN—Receive (or ingress) SPAN monitors as much as possible all of the packets received by the source interface or VLAN before any modification or processing is performed by the switch. A copy of each packet received by the source is sent to the destination port for that SPAN session.

  Packets that are modified because of routing or quality of service (QoS)—for example, modified Differentiated Services Code Point (DSCP)—are copied before modification.

  Features that can cause a packet to be dropped during receive processing have no effect on ingress SPAN; the destination port receives a copy of the packet even if the actual incoming packet is dropped. These features include IP standard and extended input access control lists (ACLs), ingress QoS policing, VLAN ACLs, and egress QoS policing.

- Transmit (Tx) SPAN—Transmit (or egress) SPAN monitors as much as possible all of the packets sent by the source interface after all modification and processing is performed by the switch. A copy of each packet sent by the source is sent to the destination port for that SPAN session. The copy is provided after the packet is modified.

  Packets that are modified because of routing (for example, with modified time-to-live (TTL), MAC address, or QoS values) are duplicated (with the modifications) at the destination port.

  Features that can cause a packet to be dropped during transmit processing also affect the duplicated copy for SPAN. These features include IP standard and extended output ACLs and egress QoS policing.

- Both—In a SPAN session, you can also monitor a port or VLAN for both received and sent packets. This is the default.

The default configuration for local SPAN session ports is to send all packets untagged. SPAN also does not normally monitor bridge protocol data unit (BPDU) packets and Layer 2 protocols, such as Cisco Discovery Protocol (CDP), VLAN Trunk Protocol (VTP), Dynamic Trunking Protocol (DTP), Spanning Tree Protocol (STP), and Port Aggregation Protocol (PAgP). However, when you enter the **encapsulation replicate** keywords when configuring a destination port, these changes occur:

- Packets are sent on the destination port with the same encapsulation (untagged or IEEE 802.1Q) that they had on the source port.
- Packets of all types, including BPDU and Layer 2 protocol packets, are monitored.

Therefore, a local SPAN session with encapsulation replicate enabled can have a mixture of untagged and IEEE 802.1Q tagged packets appear on the destination port.

Switch congestion can cause packets to be dropped at ingress source ports, egress source ports, or SPAN destination ports. In general, these characteristics are independent of one another. For example:

- A packet might be forwarded normally but dropped from monitoring due to an oversubscribed SPAN destination port.
- An ingress packet might be dropped from normal forwarding, but still appear on the SPAN destination port.

- An egress packet dropped because of switch congestion is also dropped from egress SPAN.

In some SPAN configurations, multiple copies of the same source packet are sent to the SPAN destination port. For example, a bidirectional (both Rx and Tx) SPAN session is configured for the Rx monitor on port A and Tx monitor on port B. If a packet enters the switch through port A and is switched to port B, both incoming and outgoing packets are sent to the destination port. Both packets are the same unless a Layer 3 rewrite occurs, in which case the packets are different because of the packet modification.

Source Ports

A source port (also called a monitored port) is a switched or routed port that you monitor for network traffic analysis. In a local SPAN session or RSPAN source session, you can monitor source ports or VLANs for traffic in one or both directions. The switch supports any number of source ports (up to the maximum number of available ports on the switch) and any number of source VLANs (up to the maximum number of VLANs supported). However, the switch supports a maximum of two sessions (local or RSPAN) with source ports or VLANs. You cannot mix ports and VLANs in a single session.

A source port has these characteristics:

- It can be monitored in multiple SPAN sessions.
- Each source port can be configured with a direction (ingress, egress, or both) to monitor.
- It can be any port type (for example, EtherChannel, Gigabit Ethernet, and so forth).
- For EtherChannel sources, you can monitor traffic for the entire EtherChannel or individually on a physical port as it participates in the port channel.
- It can be an access port, trunk port, routed port, or voice VLAN port.
- It cannot be a destination port.
- Source ports can be in the same or different VLANs.
- You can monitor multiple source ports in a single session.

Source VLANs

VLAN-based SPAN (VSPAN) is the monitoring of the network traffic in one or more VLANs. The SPAN or RSPAN source interface in VSPAN is a VLAN ID, and traffic is monitored on all the ports for that VLAN.

VSPAN has these characteristics:

- All active ports in the source VLAN are included as source ports and can be monitored in either or both directions.
- On a given port, only traffic on the monitored VLAN is sent to the destination port.
- If a destination port belongs to a source VLAN, it is excluded from the source list and is not monitored.
- If ports are added to or removed from the source VLANs, the traffic on the source VLAN received by those ports is added to or removed from the sources being monitored.
- You cannot use filter VLANs in the same session with VLAN sources.
- You can monitor only Ethernet VLANs.

VLAN Filtering

When you monitor a trunk port as a source port, by default, all VLANs active on the trunk are monitored. You can limit SPAN traffic monitoring on trunk source ports to specific VLANs by using VLAN filtering.

- VLAN filtering applies only to trunk ports or to voice VLAN ports.
- VLAN filtering applies only to port-based sessions and is not allowed in sessions with VLAN sources.
- When a VLAN filter list is specified, only those VLANs in the list are monitored on trunk ports or on voice VLAN access ports.
- SPAN traffic coming from other port types is not affected by VLAN filtering; that is, all VLANs are allowed on other ports.
- VLAN filtering affects only traffic forwarded to the destination SPAN port and does not affect the switching of normal traffic.

Destination Port

Each local SPAN session or RSPAN destination session must have a destination port (also called a monitoring port) that receives a copy of traffic from the source ports or VLANs and sends the SPAN packets to the user, usually a network analyzer.

A destination port has these characteristics:

- For a local SPAN session, the destination port must reside on the same switch or switch stack as the source port. For an RSPAN session, it is located on the switch containing the RSPAN destination session. There is no destination port on a switch or switch stack running only an RSPAN source session.
- When a port is configured as a SPAN destination port, the configuration overwrites the original port configuration. When the SPAN destination configuration is removed, the port reverts to its previous configuration. If a configuration change is made to the port while it is acting as a SPAN destination port, the change does not take effect until the SPAN destination configuration had been removed.

Note When QoS is configured on the SPAN destination port, QoS takes effect immediately.

- If the port was in an EtherChannel group, it is removed from the group while it is a destination port. If it was a routed port, it is no longer a routed port.
- It can be any Ethernet physical port.
- It cannot be a secure port.
- It cannot be a source port.
- It can be an EtherChannel group.
- It cannot be a VLAN.
- It can participate in only one SPAN session at a time (a destination port in one SPAN session cannot be a destination port for a second SPAN session).
- When it is active, incoming traffic is disabled. The port does not transmit any traffic except that required for the SPAN session. Incoming traffic is never learned or forwarded on a destination port.
- If ingress traffic forwarding is enabled for a network security device, the destination port forwards traffic at Layer 2.
- It does not participate in any of the Layer 2 protocols (STP, VTP, CDP, DTP, PagP).
- A destination port that belongs to a source VLAN of any SPAN session is excluded from the source list and is not monitored.

- The maximum number of destination ports in a switch or switch stack is 64.

Local SPAN and RSPAN destination ports function differently with VLAN tagging and encapsulation:

- For local SPAN, if the **encapsulation replicate** keywords are specified for the destination port, these packets appear with the original encapsulation (untagged, ISL, or IEEE 802.1Q). If these keywords are not specified, packets appear in the untagged format. Therefore, the output of a local SPAN session with **encapsulation replicate** enabled can contain a mixture of untagged, ISL, or IEEE 802.1Q-tagged packets.
- For RSPAN, the original VLAN ID is lost because it is overwritten by the RSPAN VLAN identification. Therefore, all packets appear on the destination port as untagged.

RSPAN VLAN

The RSPAN VLAN carries SPAN traffic between RSPAN source and destination sessions. RSPAN VLAN has these special characteristics:

- All traffic in the RSPAN VLAN is always flooded.
- No MAC address learning occurs on the RSPAN VLAN.
- RSPAN VLAN traffic only flows on trunk ports.
- RSPAN VLANs must be configured in VLAN configuration mode by using the **remote-span** VLAN configuration mode command.
- STP can run on RSPAN VLAN trunks but not on SPAN destination ports.
- An RSPAN VLAN cannot be a private-VLAN primary or secondary VLAN.

For VLANs 1 to 1005 that are visible to VLAN Trunking Protocol (VTP), the VLAN ID and its associated RSPAN characteristic are propagated by VTP. If you assign an RSPAN VLAN ID in the extended VLAN range (1006 to 4094), you must manually configure all intermediate switches.

It is normal to have multiple RSPAN VLANs in a network at the same time with each RSPAN VLAN defining a network-wide RSPAN session. That is, multiple RSPAN source sessions anywhere in the network can contribute packets to the RSPAN session. It is also possible to have multiple RSPAN destination sessions throughout the network, monitoring the same RSPAN VLAN and presenting traffic to the user. The RSPAN VLAN ID separates the sessions.

SPAN and RSPAN Interaction with Other Features

SPAN interacts with these features:

- Routing—SPAN does not monitor routed traffic. VSPAN only monitors traffic that enters or exits the switch, not traffic that is routed between VLANs. For example, if a VLAN is being Rx-monitored and the switch routes traffic from another VLAN to the monitored VLAN, that traffic is not monitored and not received on the SPAN destination port.
- STP—A destination port does not participate in STP while its SPAN or RSPAN session is active. The destination port can participate in STP after the SPAN or RSPAN session is disabled. On a source port, SPAN does not affect the STP status. STP can be active on trunk ports carrying an RSPAN VLAN.
- CDP—A SPAN destination port does not participate in CDP while the SPAN session is active. After the SPAN session is disabled, the port again participates in CDP.
- VTP—You can use VTP to prune an RSPAN VLAN between switches.

- VLAN and trunking—You can modify VLAN membership or trunk settings for source or destination ports at any time. However, changes in VLAN membership or trunk settings for a destination port do not take effect until you remove the SPAN destination configuration. Changes in VLAN membership or trunk settings for a source port immediately take effect, and the respective SPAN sessions automatically adjust accordingly.
- EtherChannel—You can configure an EtherChannel group as a source port or a SPAN destination port. When a group is configured as a SPAN source, the entire group is monitored.

  If a physical port is added to a monitored EtherChannel group, the new port is added to the SPAN source port list. If a port is removed from a monitored EtherChannel group, it is automatically removed from the source port list.

  A physical port that belongs to an EtherChannel group can be configured as a SPAN source port and still be a part of the EtherChannel. In this case, data from the physical port is monitored as it participates in the EtherChannel. However, if a physical port that belongs to an EtherChannel group is configured as a SPAN destination, it is removed from the group. After the port is removed from the SPAN session, it rejoins the EtherChannel group. Ports removed from an EtherChannel group remain members of the group, but they are in the inactive or suspended state.

  If a physical port that belongs to an EtherChannel group is a destination port and the EtherChannel group is a source, the port is removed from the EtherChannel group and from the list of monitored ports.
- Multicast traffic can be monitored. For egress and ingress port monitoring, only a single unedited packet is sent to the SPAN destination port. It does not reflect the number of times the multicast packet is sent.
- A private-VLAN port cannot be a SPAN destination port.
- A secure port cannot be a SPAN destination port.

  For SPAN sessions, do not enable port security on ports with monitored egress when ingress forwarding is enabled on the destination port. For RSPAN source sessions, do not enable port security on any ports with monitored egress.
- An IEEE 802.1x port can be a SPAN source port. You can enable IEEE 802.1x on a port that is a SPAN destination port; however, IEEE 802.1x is disabled until the port is removed as a SPAN destination.

  For SPAN sessions, do not enable IEEE 802.1x on ports with monitored egress when ingress forwarding is enabled on the destination port. For RSPAN source sessions, do not enable IEEE 802.1x on any ports that are egress monitored.

SPAN and RSPAN and Device Stacks

Because the stack of switches represents one logical switch, local SPAN source ports and destination ports can be in different switches in the stack. Therefore, the addition or deletion of switches in the stack can affect a local SPAN session, as well as an RSPAN source or destination session. An active session can become inactive when a switch is removed from the stack or an inactive session can become active when a switch is added to the stack.

Flow-Based SPAN

You can control the type of network traffic to be monitored in SPAN or RSPAN sessions by using flow-based SPAN (FSPAN) or flow-based RSPAN (FRSPAN), which apply access control lists (ACLs) to the monitored traffic on the source ports. The FSPAN ACLs can be configured to filter IPv4, IPv6, and non-IP monitored traffic.

You apply an ACL to a SPAN session through the interface. It is applied to all the traffic that is monitored on all interfaces in the SPAN session.The packets that are permitted by this ACL are copied to the SPAN destination port. No other packets are copied to the SPAN destination port.

The original traffic continues to be forwarded, and any port, VLAN, and router ACLs attached are applied. The FSPAN ACL does not have any effect on the forwarding decisions. Similarly, the port, VLAN, and router ACLs do not have any effect on the traffic monitoring. If a security input ACL denies a packet and it is not forwarded, the packet is still copied to the SPAN destination ports if the FSPAN ACL permits it. But if the security output ACL denies a packet and it is not sent, it is not copied to the SPAN destination ports. However, if the security output ACL permits the packet to go out, it is only copied to the SPAN destination ports if the FSPAN ACL permits it. This is also true for an RSPAN session.

You can attach three types of FSPAN ACLs to the SPAN session:

- IPv4 FSPAN ACL— Filters only IPv4 packets.
- IPv6 FSPAN ACL— Filters only IPv6 packets.
- MAC FSPAN ACL— Filters only non-IP packets.

The security ACLs have higher priority than the FSPAN ACLs on a switch. If FSPAN ACLs are applied, and you later add more security ACLs that cannot fit in the hardware memory, the FSPAN ACLs that you applied are removed from memory to allow space for the security ACLs. A system message notifies you of this action, which is called unloading. When there is again space for the FSPAN ACLs to reside in memory, they are added to the hardware memory on the switch. A system message notifies you of this action, which is called reloading. The IPv4, IPv6 and MAC FSPAN ACLs can be unloaded or reloaded independently.

If a VLAN-based FSPAN session configured on a stack cannot fit in the hardware memory on one or more switches, it is treated as unloaded on those switches, and traffic meant for the FSPAN ACL and sourcing on that switch is not copied to the SPAN destination ports. The FSPAN ACL continues to be correctly applied, and traffic is copied to the SPAN destination ports on the switches where the FSPAN ACL fits in the hardware memory.

When an empty FSPAN ACL is attached, some hardware functions copy all traffic to the SPAN destination ports for that ACL. If sufficient hardware resources are not available, even an empty FSPAN ACL can be unloaded.

IPv4 and MAC FSPAN ACLs are supported on all feature sets. IPv6 FSPAN ACLs are supported only in the advanced IP Services feature set.

Default SPAN and RSPAN Configuration

Table 176: Default SPAN and RSPAN Configuration

Feature	Default Setting
SPAN state (SPAN and RSPAN)	Disabled.
Source port traffic to monitor	Both received and sent traffic (**both**).
Encapsulation type (destination port)	Native form (untagged packets).
Ingress forwarding (destination port)	Disabled.

Feature	Default Setting
VLAN filtering	On a trunk interface used as a source port, all VLANs are monitored.
RSPAN VLANs	None configured.

Configuration Guidelines

SPAN Configuration Guidelines

- To remove a source or destination port or VLAN from the SPAN session, use the **no monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} global configuration command or the **no monitor session** *session_number* **destination interface** *interface-id* global configuration command. For destination interfaces, the **encapsulation** options are ignored with the **no** form of the command.
- To monitor all VLANs on the trunk port, use the **no monitor session** *session_number* **filter** global configuration command.

RSPAN Configuration Guidelines

- All the SPAN configuration guidelines apply to RSPAN.
- As RSPAN VLANs have special properties, you should reserve a few VLANs across your network for use as RSPAN VLANs; do not assign access ports to these VLANs.
- You can apply an output ACL to RSPAN traffic to selectively filter or monitor specific packets. Specify these ACLs on the RSPAN VLAN in the RSPAN source switches.
- For RSPAN configuration, you can distribute the source ports and the destination ports across multiple switches in your network.
- Access ports (including voice VLAN ports) on the RSPAN VLAN are put in the inactive state.
- You can configure any VLAN as an RSPAN VLAN as long as these conditions are met:
    - The same RSPAN VLAN is used for an RSPAN session in all the switches.
    - All participating switches support RSPAN.

FSPAN and FRSPAN Configuration Guidelines

- When at least one FSPAN ACL is attached, FSPAN is enabled.
- When you attach at least one FSPAN ACL that is not empty to a SPAN session, and you have not attached one or more of the other FSPAN ACLs (for instance, you have attached an IPv4 ACL that is not empty, and have not attached IPv6 and MAC ACLs), FSPAN blocks the traffic that would have been filtered by the unattached ACLs. Therefore, this traffic is not monitored.

How to Configure SPAN and RSPAN

Creating a Local SPAN Session

Beginning in privileged EXEC mode, follow these steps to create a SPAN session and specify the source (monitored) ports or VLANs and the destination (monitoring) ports.

SUMMARY STEPS

1. **configure terminal**
2. **no monitor session** {*session_number* | **all** | **local** | **remote**}
3. **monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]
4. **monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]}
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** `Switch(config)# no monitor session all`	Removes any existing SPAN configuration for the session. • For *session_number*, the range is 1 to 66. • **all—**Removes all SPAN sessions. • **local—**Removes all local sessions. • **remote—**Removes all remote SPAN sessions.
Step 3	**monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**] **Example:** `Switch(config)# monitor session 1 source interface gigabitethernet1/0/1`	Specifies the SPAN session and the source port (monitored port). • For *session_number*, the range is 1 to 66. • For *interface-id*, specifies the source port to monitor. Valid interfaces include physical interfaces and port-channel logical interfaces (**port-channel** *port-channel-number*). Valid port-channel numbers are 1 to 48. • For *vlan-id*, specify the source VLAN to monitor. The range is 1 to 4094 (excluding the RSPAN VLAN). **Note** A single session can include multiple sources (ports or VLANs) defined in a series of commands, but you cannot combine source ports and source VLANs in one session.

	Command or Action	Purpose
		• (Optional) [**,** | **-**] Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. • (Optional) **both** | **rx** | **tx**—Specifies the direction of traffic to monitor. If you do not specify a traffic direction, the source interface sends both sent and received traffic. ◦ **both**—Monitors both received and sent traffic. ◦ **rx**—Monitors received traffic. ◦ **tx**—Monitors sent traffic. **Note** You can use the **monitor session** *session_number* **source** command multiple times to configure multiple source ports.
Step 4	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]} **Example:** `Switch(config)# monitor session 1 destination interface gigabitethernet1/0/2 encapsulation replicate`	Specifies the SPAN session and the destination port (monitoring port). **Note** For local SPAN, you must use the same session number for the source and destination interfaces. • For *session_number*, specify the session number entered in step 3. • For *interface-id*, specify the destination port. The destination interface must be a physical port; it cannot be an EtherChannel, and it cannot be a VLAN. • (Optional) [**,** | **-**] Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. (Optional) **encapsulation replicate** specifies that the destination interface replicates the source interface encapsulation method. If not selected, the default is to send packets in native form (untagged). **Note** You can use **monitor session** *session_number* **destination** command multiple times to configure multiple destination ports.
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Creating a Local SPAN Session and Configuring Incoming Traffic

Beginning in privileged EXEC mode, follow these steps to create a SPAN session, to specify the source ports or VLANs and the destination ports, and to enable incoming traffic on the destination port for a network security device (such as a Cisco IDS Sensor Appliance).

SUMMARY STEPS

1. **configure terminal**
2. **no monitor session** {*session_number* | **all** | **local** | **remote**}
3. **monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]
4. **monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate]** [**ingress** {**dot1q vlan** *vlan-id* | **isl** | **untagged vlan** *vlan-id* | **vlan** *vlan-id*}]}
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** `Switch(config)# no monitor session all`	Removes any existing SPAN configuration for the session. • For *session_number*, the range is 1 to 66. • **all—**Removes all SPAN sessions. • **local—**Removes all local sessions. • **remote—**Removes all remote SPAN sessions.
Step 3	**monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**] **Example:** `Switch(config)# monitor session 2 source gigabitethernet1/0/1 rx`	Specifies the SPAN session and the source port (monitored port).
Step 4	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate]** [**ingress** {**dot1q vlan** *vlan-id* | **isl** | **untagged vlan** *vlan-id* | **vlan** *vlan-id*}]} **Example:** `Switch(config)# monitor session 2 destination interface gigabitethernet1/0/2 encapsulation replicate ingress dot1q vlan 6`	Specifies the SPAN session, the destination port, the packet encapsulation, and the ingress VLAN and encapsulation. • For *session_number*, specify the session number entered in Step 3. • For *interface-id*, specify the destination port. The destination interface must be a physical port; it cannot be an EtherChannel, and it cannot be a VLAN. • (Optional) [**,** | **-**]—Specifies a series or range of interfaces. Enter a space before and after the comma or hyphen. • (Optional) **encapsulation replicate** specifies that the destination interface replicates the source interface encapsulation method. If not selected, the default is to send packets in native form (untagged).

	Command or Action	Purpose
		• **ingress** enables forwarding of incoming traffic on the destination port and to specify the encapsulation type: ◦ **dot1q vlan** *vlan-id*—Accepts incoming packets with IEEE 802.1Q encapsulation with the specified VLAN as the default VLAN. ◦ **isl**—Forwards ingress packets with ISL encapsulation. ◦ **untagged vlan** *vlan-id* or **vlan** *vlan-id*—Accepts incoming packets with untagged encapsulation type with the specified VLAN as the default VLAN. • **dot1q vlan** *vlan-id*—Accepts incoming packets with IEEE 802.1Q encapsulation with the specified VLAN as the default VLAN. • **isl**—Forwards ingress packets with ISL encapsulation. • **untagged vlan** *vlan-id* or **vlan** *vlan-id*—Accepts incoming packets with untagged encapsulation type with the specified VLAN as the default VLAN.
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Specifying VLANs to Filter

Beginning in privileged EXEC mode, follow these steps to limit SPAN source traffic to specific VLANs.

SUMMARY STEPS

1. **configure terminal**
2. **no monitor session** {*session_number* | **all** | **local** | **remote**}
3. **monitor session** *session_number* **source interface** *interface-id*
4. **monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**]
5. **monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]}
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** `Switch(config)# no monitor session all`	Removes any existing SPAN configuration for the session. • For *session_number*, the range is 1 to 66. • **all**—Removes all SPAN sessions. • **local**—Removes all local sessions. • **remote**—Removes all remote SPAN sessions.
Step 3	**monitor session** *session_number* **source interface** *interface-id* **Example:** `Switch(config)# monitor session 2 source interface gigabitethernet1/0/2 rx`	Specifies the characteristics of the source port (monitored port) and SPAN session. • For *session_number*, the range is 1 to 66. • For *interface-id*, specify the source port to monitor. The interface specified must already be configured as a trunk port.
Step 4	**monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**] **Example:** `Switch(config)# monitor session 2 filter vlan 1 - 5 , 9`	Limits the SPAN source traffic to specific VLANs. • For *session_number*, enter the session number specified in Step 3. • For *vlan-id*, the range is 1 to 4094. • (Optional) Use a comma (**,**) to specify a series of VLANs, or use a hyphen (**-**) to specify a range of VLANs. Enter a space before and after the comma; enter a space before and after the hyphen.
Step 5	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]} **Example:** `Switch(config)# monitor session 2 destination interface gigabitethernet1/0/1`	Specifies the SPAN session and the destination port (monitoring port). • For *session_number*, specify the session number entered in Step 3. • For *interface-id*, specify the destination port. The destination interface must be a physical port; it cannot be an EtherChannel, and it cannot be a VLAN. • (Optional) [**,** | **-**] Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. • (Optional) **encapsulation replicate** specifies that the destination interface replicates the source interface encapsulation method. If not selected, the default is to send packets in native form (untagged).

	Command or Action	Purpose
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring a VLAN as an RSPAN VLAN

Beginning in privileged EXEC mode, follow these steps to create a new VLAN, then configure it to be the RSPAN VLAN for the RSPAN session.

SUMMARY STEPS

1. **configure terminal**
2. **vlan** *vlan-id*
3. **remote-span**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**vlan** *vlan-id* **Example:** `Switch(config)# vlan 100`	Enters a VLAN ID to create a VLAN, or enters the VLAN ID of an existing VLAN, and enters VLAN configuration mode. The range is 2 to 1001 and 1006 to 4094. The RSPAN VLAN cannot be VLAN 1 (the default VLAN) or VLAN IDs 1002 through 1005 (reserved for Token Ring and FDDI VLANs).
Step 3	**remote-span** **Example:** `Switch(config-vlan)# remote-span`	Configures the VLAN as an RSPAN VLAN.
Step 4	**end** **Example:** `Switch(config-vlan)# end`	Returns to privileged EXEC mode.

What to Do Next

You must create the RSPAN VLAN in all switches that will participate in RSPAN. If the RSPAN VLAN-ID is in the normal range (lower than 1005) and VTP is enabled in the network, you can create the RSPAN VLAN in one switch, and VTP propagates it to the other switches in the VTP domain. For extended-range VLANs (greater than 1005), you must configure RSPAN VLAN on both source and destination switches and any intermediate switches.

Use VTP pruning to get an efficient flow of RSPAN traffic, or manually delete the RSPAN VLAN from all trunks that do not need to carry the RSPAN traffic.

To remove the remote SPAN characteristic from a VLAN and convert it back to a normal VLAN, use the **no remote-span** VLAN configuration command.

To remove a source port or VLAN from the SPAN session, use the **no monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} global configuration command. To remove the RSPAN VLAN from the session, use the **no monitor session** *session_number* **destination remote vlan** *vlan-id*.

Creating an RSPAN Source Session

Beginning in privileged EXEC mode, follow these steps to create and start an RSPAN source session and to specify the monitored source and the destination RSPAN VLAN.

SUMMARY STEPS

1. **configure terminal**
2. **no monitor session** {*session_number* | **all** | **local** | **remote**}
3. **monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]
4. **monitor session** *session_number* **destination remote vlan** *vlan-id*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** `Switch(config)# no monitor session 1`	Removes any existing SPAN configuration for the session. • For *session_number*, the range is 1 to 66. • **all**—Removes all SPAN sessions. • **local**—Removes all local sessions. • **remote**—Removes all remote SPAN sessions.

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td>Step 3</td><td>monitor session session_number source {interface interface-id | vlan vlan-id} [, | -] [both | rx | tx]

Example:

<code>Switch(config)# monitor session 1 source interface gigabitethernet1/0/1 tx</code></td><td>Specifies the RSPAN session and the source port (monitored port).
For session_number, the range is 1 to 66.
Enter a source port or source VLAN for the RSPAN session:
For interface-id, specifies the source port to monitor. Valid interfaces include physical interfaces and port-channel logical interfaces (port-channel port-channel-number). Valid port-channel numbers are 1 to 48.
For vlan-id, specifies the source VLAN to monitor. The range is 1 to 4094 (excluding the RSPAN VLAN).
A single session can include multiple sources (ports or VLANs), defined in a series of commands, but you cannot combine source ports and source VLANs in one session. (Optional) [, | -]—Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. (Optional) both | rx | tx—Specifies the direction of traffic to monitor. If you do not specify a traffic direction, the source interface sends both sent and received traffic. both—Monitors both received and sent traffic. rx—Monitors received traffic. tx—Monitors sent traffic.</td></tr>
<tr><td>Step 4</td><td>monitor session session_number destination remote vlan vlan-id

Example:

<code>Switch(config)# monitor session 1 destination remote vlan 100</code></td><td>Specifies the RSPAN session, the destination RSPAN VLAN, and the destination-port group.
For session_number, enter the number defined in Step 3.
For vlan-id, specify the source RSPAN VLAN to monitor.</td></tr>
<tr><td>Step 5</td><td>end

Example:

<code>Switch(config)# end</code></td><td>Returns to privileged EXEC mode.</td></tr>
</table>

Specifying VLANs to Filter

Beginning in privileged EXEC mode, follow these steps to configure the RSPAN source session to limit RSPAN source traffic to specific VLANs.

SUMMARY STEPS

1. **configure terminal**
2. **no monitor session** {*session_number* | **all** | **local** | **remote**}
3. **monitor session** *session_number* **source interface** *interface-id*
4. **monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**]
5. **monitor session** *session_number* **destination remote vlan** *vlan-id*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** `Switch(config)# no monitor session 2`	Removes any existing SPAN configuration for the session. • For *session_number*, the range is 1 to 66. • **all—**Removes all SPAN sessions. • **local—**Removes all local sessions. • **remote—**Removes all remote SPAN sessions.
Step 3	**monitor session** *session_number* **source interface** *interface-id* **Example:** `Switch(config)# monitor session 2 source interface gigabitethernet1/0/2 rx`	Specifies the characteristics of the source port (monitored port) and SPAN session. • For *session_number*, the range is 1 to 66. • For *interface-id*, specify the source port to monitor. The interface specified must already be configured as a trunk port.
Step 4	**monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**] **Example:** `Switch(config)# monitor session 2 filter vlan 1 - 5 , 9`	Limits the SPAN source traffic to specific VLANs. • For *session_number*, enter the session number specified in step 3. • For *vlan-id*, the range is 1 to 4094. • (Optional) **,** | **-** Use a comma (**,**) to specify a series of VLANs or use a hyphen (**-**) to specify a range of VLANs. Enter a space before and after the comma; enter a space before and after the hyphen.
Step 5	**monitor session** *session_number* **destination remote vlan** *vlan-id*	Specifies the RSPAN session and the destination remote VLAN (RSPAN VLAN).

	Command or Action	Purpose
	Example: `Switch(config)# monitor session 2 destination remote vlan 902`	• For *session_number*, enter the session number specified in Step 3. • For *vlan-id*, specify the RSPAN VLAN to carry the monitored traffic to the destination port.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Creating an RSPAN Destination Session

You configure an RSPAN destination session on a different switch or switch stack; that is, not the switch or switch stack on which the source session was configured.

Beginning in privileged EXEC mode, follow these steps to define the RSPAN VLAN on that switch, to create an RSPAN destination session, and to specify the source RSPAN VLAN and the destination port.

SUMMARY STEPS

1. **configure terminal**
2. **vlan** *vlan-id*
3. **remote-span**
4. **exit**
5. **no monitor session** {*session_number* | **all** | **local** | **remote**}
6. **monitor session** *session_number* **source remote vlan** *vlan-id*
7. **monitor session** *session_number* **destination interface** *interface-id*
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	**vlan** *vlan-id* **Example:** `Switch(config)# vlan 901`	Specifies the VLAN ID of the RSPAN VLAN created from the source switch, and enters VLAN configuration mode. If both switches are participating in VTP and the RSPAN VLAN ID is from 2 to 1005, Steps 2 through 4 are not required because the RSPAN VLAN ID is propagated through the VTP network.
Step 3	**remote-span** **Example:** `Switch(config-vlan)# remote-span`	Identifies the VLAN as the RSPAN VLAN.
Step 4	**exit** **Example:** `Switch(config-vlan)# exit`	Returns to global configuration mode.
Step 5	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** `Switch(config)# no monitor session 1`	Removes any existing SPAN configuration for the session. • For *session_number*, the range is 1 to 66. • **all—**Removes all SPAN sessions. • **local—**Removes all local sessions. • **remote—**Removes all remote SPAN sessions.
Step 6	**monitor session** *session_number* **source remote vlan** *vlan-id* **Example:** `Switch(config)# monitor session 1 source remote vlan 901`	Specifies the RSPAN session and the source RSPAN VLAN. • For *session_number*, the range is 1 to 66. • For *vlan-id*, specify the source RSPAN VLAN to monitor.
Step 7	**monitor session** *session_number* **destination interface** *interface-id* **Example:** `Switch(config)# monitor session 1 destination interface gigabitethernet2/0/1`	Specifies the RSPAN session and the destination interface. • For *session_number*, enter the number defined in Step 6. In an RSPAN destination session, you must use the same session number for the source RSPAN VLAN and the destination port. • For *interface-id*, specify the destination interface. The destination interface must be a physical interface. • Though visible in the command-line help string, **encapsulation replicate** is not supported for RSPAN. The original VLAN ID is overwritten by the RSPAN VLAN ID, and all packets appear on the destination port as untagged.

	Command or Action	Purpose
Step 8	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Creating an RSPAN Destination Session and Configuring Incoming Traffic

Beginning in privileged EXEC mode, follow these steps to create an RSPAN destination session, to specify the source RSPAN VLAN and the destination port, and to enable incoming traffic on the destination port for a network security device (such as a Cisco IDS Sensor Appliance).

SUMMARY STEPS

1. **configure terminal**
2. **no monitor session** {*session_number* | **all** | **local** | **remote**}
3. **monitor session** *session_number* **source remote vlan** *vlan-id*
4. **monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**ingress** {**dot1q vlan** *vlan-id* | **isl** | **untagged vlan** *vlan-id* | **vlan** *vlan-id*}]}
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** `Switch(config)# no monitor session 2`	Removes any existing SPAN configuration for the session. • For *session_number*, the range is 1 to 66. • **all**—Removes all SPAN sessions. • **local**—Removes all local sessions. • **remote**—Removes all remote SPAN sessions.
Step 3	**monitor session** *session_number* **source remote vlan** *vlan-id*	Specifies the RSPAN session and the source RSPAN VLAN. • For *session_number*, the range is 1 to 66.

	Command or Action	Purpose
	Example: `Switch(config)# monitor session 2 source remote vlan 901`	• For *vlan-id*, specify the source RSPAN VLAN to monitor.
Step 4	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**ingress** {**dot1q vlan** *vlan-id* | **isl** | **untagged vlan** *vlan-id* | **vlan** *vlan-id*}]} **Example:** `Switch(config)# monitor session 2 destination interface gigabitethernet1/0/2 ingress vlan 6`	Specifies the SPAN session, the destination port, the packet encapsulation, and the incoming VLAN and encapsulation. • For *session_number*, enter the number defined in Step 4. In an RSPAN destination session, you must use the same session number for the source RSPAN VLAN and the destination port. • For *interface-id*, specify the destination interface. The destination interface must be a physical interface. • Though visible in the command-line help string, **encapsulation replicate** is not supported for RSPAN. The original VLAN ID is overwritten by the RSPAN VLAN ID, and all packets appear on the destination port as untagged. • (Optional) [**,** | **-**] Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. • Enter **ingress** with additional keywords to enable forwarding of incoming traffic on the destination port and to specify the encapsulation type: ◦ **dot1q vlan** *vlan-id*—Forwards incoming packets with IEEE 802.1Q encapsulation with the specified VLAN as the default VLAN. ◦ **isl**—Forwards ingress packets with ISL encapsulation. ◦ **untagged vlan** *vlan-id* or **vlan** *vlan-id*—Forwards incoming packets with untagged encapsulation type with the specified VLAN as the default VLAN.
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring an FSPAN Session

Beginning in privileged EXEC mode, follow these steps to create a SPAN session, specify the source (monitored) ports or VLANs and the destination (monitoring) ports, and configure FSPAN for the session.

SUMMARY STEPS

1. **configure terminal**
2. **no monitor session** {*session_number* | **all** | **local** | **remote**}
3. **monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]
4. **monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]}
5. **monitor session** *session_number* **filter** {**ip** | **ipv6** | **mac**} **access-group** {*access-list-number* | *name*}
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** `Switch(config)# no monitor session 2`	Removes any existing SPAN configuration for the session. • For *session_number*, the range is 1 to 66. • **all**—Removes all SPAN sessions. • **local**—Removes all local sessions. • **remote**—Removes all remote SPAN sessions.
Step 3	**monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**] **Example:** `Switch(config)# monitor session 2 source interface gigabitethernet1/0/1`	Specifies the SPAN session and the source port (monitored port). • For *session_number*, the range is 1 to 66. • For *interface-id*, specifies the source port to monitor. Valid interfaces include physical interfaces and port-channel logical interfaces (**port-channel** *port-channel-number*). Valid port-channel numbers are 1 to 48. • For *vlan-id*, specify the source VLAN to monitor. The range is 1 to 4094 (excluding the RSPAN VLAN). **Note** A single session can include multiple sources (ports or VLANs) defined in a series of commands, but you cannot combine source ports and source VLANs in one session. • (Optional) [**,** | **-**]—Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. • (Optional) [**both** | **rx** | **tx**]—Specifies the direction of traffic to monitor. If you do not specify a traffic direction, the SPAN monitors both sent and received traffic. ◦ **both**—Monitors both sent and received traffic. This is the default. ◦ **rx**—Monitors received traffic.

	Command or Action	Purpose
		◦ **tx**—Monitors sent traffic. **Note** You can use the **monitor session** *session_number* **source** command multiple times to configure multiple source ports.
Step 4	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]} **Example:** `Switch(config)# monitor session 2 destination interface gigabitethernet1/0/2 encapsulation replicate`	Specifies the SPAN session and the destination port (monitoring port). • For *session_number*, specify the session number entered in Step 3. • For **destination**, specify the following parameters: ◦ For *interface-id*, specify the destination port. The destination interface must be a physical port; it cannot be an EtherChannel, and it cannot be a VLAN. ◦ (Optional) [**,** | **-**] Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. ◦ (Optional) **encapsulation replicate** specifies that the destination interface replicates the source interface encapsulation method. If not selected, the default is to send packets in native form (untagged). **Note** For local SPAN, you must use the same session number for the source and destination interfaces. You can use **monitor session** *session_number* **destination** command multiple times to configure multiple destination ports.
Step 5	**monitor session** *session_number* **filter** {**ip** | **ipv6** | **mac**} **access-group** {*access-list-number* | *name*} **Example:** `Switch(config)# monitor session 2 filter ipv6 access-group 4`	Specifies the SPAN session, the types of packets to filter, and the ACLs to use in an FSPAN session. • For *session_number*, specify the session number entered in Step 3. • For *access-list-number*, specify the ACL number that you want to use to filter traffic. • For *name*, specify the ACL name that you want to use to filter traffic.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring an FRSPAN Session

Beginning in privileged EXEC mode, follow these steps to start an RSPAN source session, specify the monitored source and the destination RSPAN VLAN, and configure FRSPAN for the session.

SUMMARY STEPS

1. **configure terminal**
2. **no monitor session** {*session_number* | **all** | **local** | **remote**}
3. **monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]
4. **monitor session** *session_number* **destination remote vlan** *vlan-id*
5. **vlan** *vlan-id*
6. **remote-span**
7. **exit**
8. **monitor session** *session_number* **filter** {**ip** | **ipv6** | **mac**} **access-group** {*access-list-number* | *name*}
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** `Switch(config)# no monitor session 2`	Removes any existing SPAN configuration for the session. • For *session_number*, the range is 1 to 66. • **all—**Removes all SPAN sessions. • **local—**Removes all local sessions. • **remote—**Removes all remote SPAN sessions.
Step 3	**monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**] **Example:** `Switch(config)# monitor session 2 source interface gigabitethernet1/0/1`	Specifies the SPAN session and the source port (monitored port). • For *session_number*, the range is 1 to 66. • For *interface-id*, specifies the source port to monitor. Valid interfaces include physical interfaces and port-channel logical interfaces (**port-channel** *port-channel-number*). Valid port-channel numbers are 1 to 48. • For *vlan-id*, specify the source VLAN to monitor. The range is 1 to 4094 (excluding the RSPAN VLAN). **Note** A single session can include multiple sources (ports or VLANs) defined in a series of commands, but you cannot combine source ports and source VLANs in one session. • (Optional) [**,** | **-**]—Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen.

	Command or Action	Purpose
		• (Optional) [**both** | **rx** | **tx**]**—**Specifies the direction of traffic to monitor. If you do not specify a traffic direction, the SPAN monitors both sent and received traffic. • **both—**Monitors both sent and received traffic. This is the default. • **rx—**Monitors received traffic. • **tx—**Monitors sent traffic. **Note** You can use the **monitor session** *session_number* **source** command multiple times to configure multiple source ports.
Step 4	**monitor session** *session_number* **destination remote vlan** *vlan-id* **Example:** `Switch(config)#` **`monitor session 2 destination remote vlan 5`**	Specifies the RSPAN session and the destination RSPAN VLAN. • For *session_number*, enter the number defined in Step 3. • For *vlan-id*, specify the destination RSPAN VLAN to monitor.
Step 5	**vlan** *vlan-id* **Example:** `Switch(config)#` **`vlan 10`**	Enters the VLAN configuration mode. For *vlan-id*, specify the source RSPAN VLAN to monitor.
Step 6	**remote-span** **Example:** `Switch(config-vlan)#` **`remote-span`**	Specifies that the VLAN you specified in Step 5 is part of the RSPAN VLAN.
Step 7	**exit** **Example:** `Switch(config-vlan)#` **`exit`**	Returns to global configuration mode.
Step 8	**monitor session** *session_number* **filter** {**ip** | **ipv6** | **mac**} **access-group** {*access-list-number* | *name*} **Example:** `Switch(config)#` **`monitor session 2 filter ip access-group 7`**	Specifies the RSPAN session, the types of packets to filter, and the ACLs to use in an FRSPAN session. • For *session_number*, specify the session number entered in Step 3. • For *access-list-number*, specify the ACL number that you want to use to filter traffic. • For *name*, specify the ACL name that you want to use to filter traffic.
Step 9	**end** **Example:** `Switch(config)#` **`end`**	Returns to privileged EXEC mode.

Monitoring SPAN and RSPAN Operations

The following table describes the command used to display SPAN and RSPAN operations configuration and results to monitor operations:

Table 177: Monitoring SPAN and RSPAN Operations

Command	Purpose
show monitor	Displays the current SPAN, RSPAN, FSPAN, or FRSPAN configuration.

SPAN and RSPAN Configuration Examples

Example: Configuring Local SPAN

This example shows how to set up SPAN session 1 for monitoring source port traffic to a destination port. First, any existing SPAN configuration for session 1 is deleted, and then bidirectional traffic is mirrored from source Gigabit Ethernet port 1 to destination Gigabit Ethernet port 2, retaining the encapsulation method.

```
Switch(config)# no monitor session 1
Switch(config)# monitor session 1 source interface gigabitethernet1/0/1
Switch(config)# monitor session 1 destination interface gigabitethernet1/0/2
encapsulation replicate
Switch(config)# end
```

This example shows how to remove port 1 as a SPAN source for SPAN session 1:

```
Switch(config)# no monitor session 1 source interface gigabitethernet1/0/1
Switch(config)# end
```

This example shows how to disable received traffic monitoring on port 1, which was configured for bidirectional monitoring:

```
Switch(config)# no monitor session 1 source interface gigabitethernet1/0/1 rx
```

The monitoring of traffic received on port 1 is disabled, but traffic sent from this port continues to be monitored.

This example shows how to remove any existing configuration on SPAN session 2, configure SPAN session 2 to monitor received traffic on all ports belonging to VLANs 1 through 3, and send it to destination Gigabit Ethernet port 2. The configuration is then modified to also monitor all traffic on all ports belonging to VLAN 10.

```
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source vlan 1 - 3 rx
Switch(config)# monitor session 2 destination interface gigabitethernet1/0/2
Switch(config)# monitor session 2 source vlan 10
Switch(config)# end
```

This example shows how to remove any existing configuration on SPAN session 2, configure SPAN session 2 to monitor received traffic on Gigabit Ethernet source port 1, and send it to destination Gigabit Ethernet port 2 with the same egress encapsulation type as the source port, and to enable ingress forwarding with IEEE 802.1Q encapsulation and VLAN 6 as the default ingress VLAN:

```
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source gigabitethernet1/0/1 rx
Switch(config)# monitor session 2 destination interface gigabitethernet1/0/2 encapsulation
 replicate ingress dot1q vlan 6
Switch(config)# end
```

This example shows how to remove any existing configuration on SPAN session 2, configure SPAN session 2 to monitor traffic received on Gigabit Ethernet trunk port 2, and send traffic for only VLANs 1 through 5 and VLAN 9 to destination Gigabit Ethernet port 1:

```
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source interface gigabitethernet1/0/2 rx
Switch(config)# monitor session 2 filter vlan 1 - 5 , 9
Switch(config)# monitor session 2 destination interface gigabitethernet1/0/1
Switch(config)# end
```

Examples: Creating an RSPAN VLAN

This example shows how to create the RSPAN VLAN 901:

```
Switch(config)# vlan 901
Switch(config-vlan)# remote span
Switch(config-vlan)# end
```

This example shows how to remove any existing RSPAN configuration for session 1, configure RSPAN session 1 to monitor multiple source interfaces, and configure the destination as RSPAN VLAN 901:

```
Switch(config)# no monitor session 1
Switch(config)# monitor session 1 source interface gigabitethernet1/0/1 tx
Switch(config)# monitor session 1 source interface gigabitethernet1/0/2 rx
Switch(config)# monitor session 1 source interface port-channel 2
Switch(config)# monitor session 1 destination remote vlan 901
Switch(config)# end
```

This example shows how to remove any existing configuration on RSPAN session 2, configure RSPAN session 2 to monitor traffic received on trunk port 2, and send traffic for only VLANs 1 through 5 and 9 to destination RSPAN VLAN 902:

```
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source interface gigabitethernet1/0/2 rx
Switch(config)# monitor session 2 filter vlan 1 - 5 , 9
Switch(config)# monitor session 2 destination remote vlan 902
Switch(config)# end
```

This example shows how to configure VLAN 901 as the source remote VLAN and port 1 as the destination interface:

```
Switch(config)# monitor session 1 source remote vlan 901
Switch(config)# monitor session 1 destination interface gigabitethernet2/0/1
Switch(config)# end
```

This example shows how to configure VLAN 901 as the source remote VLAN in RSPAN session 2, to configure Gigabit Ethernet source port 2 as the destination interface, and to enable forwarding of incoming traffic on the interface with VLAN 6 as the default receiving VLAN:

```
Switch(config)# monitor session 2 source remote vlan 901
Switch(config)# monitor session 2 destination interface gigabitethernet1/0/2 ingress vlan 6
Switch(config)# end
```

Feature History and Information for SPAN and RSPAN

Release	Modification
Cisco IOS XE 3.2SE	Switch Port Analyzer (SPAN): Allows monitoring of switch traffic on a port or VLAN using a sniffer/analyzer or RMON probe. This feature was introduced.
Cisco IOS XE 3.2SE	Flow-based Switch Port Analyzer (SPAN): Provides a method to capture only required data between end hosts by using specified filters. The filters are defined in terms of access lists that limit IPv4, IPv6 or IPv4 + IPv6, or non-IP traffic (MAC) between specified source and destination addresses. This feature was introduced.
Cisco IOS XE 3.2SE	SPAN destination port support on EtherChannels: Provides the ability to configure a SPAN destination port on an EtherChannel. This feature was introduced.
Cisco IOS XE 3.2SE	Switch Port Analyzer (SPAN) - distributed egress SPAN: Provides distributed egress SPAN functionality onto line cards in conjunction with ingress SPAN already been distributed to line cards. By distributing egress SPAN functionalities onto line cards, the performance of the system is improved. This feature was introduced.

INDEX

D

F

G

H

I

M

N

O

Q

R

S

W

Z