



System Management Configuration Guide, Cisco IOS XE Fuji 16.9.x (Catalyst 9300 Switches)

First Published: 2018-07-18

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



CONTENTS

CHAPTER 1

Administering the Device 1

Information About Administering the Device	1
System Time and Date Management	1
System Clock	1
Network Time Protocol	2
NTP Stratum	3
NTP Associations	3
Authoritative NTP Server	5
NTP Security	5
NTP Services on a Specific Interface	6
Source IP Address for NTP Packets	6
NTP Implementation	6
System Name and Prompt	7
Stack System Name and Prompt	8
Default System Name and Prompt Configuration	8
DNS	8
Default DNS Settings	8
Login Banners	8
Default Banner Configuration	9
MAC Address Table	9
MAC Address Table Creation	9
MAC Addresses and VLANs	9
MAC Addresses and Device Stacks	9
Default MAC Address Table Settings	10
ARP Table Management	10
How to Administer the Device	10

Configuring the Time and Date Manually	10
Setting the System Clock	10
Configuring the Time Zone	11
Configuring Summer Time (Daylight Saving Time)	12
Configuring NTP	16
Default NTP Configuration	16
Configuring NTP Authentication	16
Configuring Poll-Based NTP Associations	17
Configuring Broadcast-Based NTP Associations	19
Configuring NTP Access Restrictions	21
Configuring a System Name	23
Setting Up DNS	24
Configuring a Message-of-the-Day Login Banner	26
Configuring a Login Banner	27
Managing the MAC Address Table	29
Changing the Address Aging Time	29
Configuring MAC Address Change Notification Traps	30
Configuring MAC Address Move Notification Traps	32
Configuring MAC Threshold Notification Traps	34
Adding and Removing Static Address Entries	36
Configuring Unicast MAC Address Filtering	37
Monitoring and Maintaining Administration of the Device	38
Configuration Examples for Device Administration	39
Example: Setting the System Clock	39
Examples: Configuring Summer Time	39
Example: Configuring a MOTD Banner	40
Example: Configuring a Login Banner	40
Example: Configuring MAC Address Change Notification Traps	41
Example: Configuring MAC Threshold Notification Traps	41
Example: Adding the Static Address to the MAC Address Table	41
Example: Configuring Unicast MAC Address Filtering	41
Additional References for Device Administration	42
Feature History for Device Administration	42

CHAPTER 2	Boot Integrity Visibility	43
	Information About Boot Integrity Visibility	43
	Verifying the Software Image and Hardware	43
	Verifying Platform Identity and Software Integrity	44
	Additional References for Boot Integrity Visibility	46
	Feature History for Boot Integrity Visibility	47

CHAPTER 3	Performing Device Setup Configuration	49
	Restrictions for Software Install	49
	Information About Performing Device Setup Configuration	49
	Device Boot Process	49
	Software Install Overview	50
	Software Boot Modes	50
	Installing the Software Package	51
	Terminating a Software Install	52
	Devices Information Assignment	52
	Default Switch Information	53
	DHCP-Based Autoconfiguration Overview	53
	DHCP Client Request Process	54
	DHCP-based Autoconfiguration and Image Update	54
	Restrictions for DHCP-based Autoconfiguration	55
	DHCP Autoconfiguration	55
	DHCP Auto-Image Update	55
	DHCP Server Configuration Guidelines	55
	Purpose of the TFTP Server	56
	Purpose of the DNS Server	57
	How to Obtain Configuration Files	57
	How to Control Environment Variables	58
	Environment Variables for TFTP	58
	Scheduled Reload of the Software Image	59
	How to Perform Device Setup Configuration	59
	Configuring DHCP Autoconfiguration (Only Configuration File)	59
	Configuring DHCP Auto-Image Update (Configuration File and Image)	61

Configuring the Client to Download Files from DHCP Server	65
Manually Assigning IP Information to Multiple SVIs	66
Modifying the Device Startup Configuration	67
Specifying the Filename to Read and Write the System Configuration	67
Manually Booting the Switch	68
Booting the Device in Installed Mode	70
Booting the Device in Bundle Mode	72
Configuring a Scheduled Software Image Reload	72
Monitoring Device Setup Configuration	74
Examples: Displaying Software Bootup in Install Mode	74
Example: Emergency Installation	77
Configuration Examples for Performing Device Setup	78
Example: Managing an Update Package	78
Verifying Software Install	88
Example: Configuring a Device as a DHCP Server	91
Example: Configuring DHCP Auto-Image Update	92
Example: Configuring a Device to Download Configurations from a DHCP Server	92
Examples: Scheduling Software Image Reload	93
Additional References For Performing Device Setup	93
Feature History for Performing Device Setup Configuration	93

CHAPTER 4**Configuring Smart Licensing 95**

Prerequisites for Configuring Smart Licensing	95
Introduction to Smart Licensing	95
Overview of CSSM	96
Connecting to CSSM	96
Linking Existing Licenses to CSSM	98
Configuring a Connection to CSSM and Setting Up the License Level	98
Setting Up a Connection to CSSM	99
Configuring the Call Home Service for Direct Cloud Access	101
Configuring the Call Home Service for Direct Cloud Access through an HTTPs Proxy Server	103
Configuring the Call Home Service for Cisco Smart Software Manager On-Prem	106
Configuring the License Level	108
Registering a Device on CSSM	110

Generating a New Token from CSSM	110
Registering a Device with the New Token	112
Verifying the License Status After Registration	113
Canceling a Device's Registration in CSSM	114
Monitoring Smart Licensing Configuration	115
Configuration Examples for Smart Licensing	116
Example: Viewing the Call Home Profile	116
Example: Viewing the License Information Before Registering	116
Example: Registering a Device	119
Example: Viewing the License Status After Registering	119
Additional References	122
Feature Information for Smart Licensing	123

CHAPTER 5**Configuring Application Visibility and Control in a Wired Network 125**

Finding Feature Information	125
Information About Application Visibility and Control in a Wired Network	125
Supported AVC Class Map and Policy Map Formats	126
Restrictions for Wired Application Visibility and Control	127
How to Configure Application Visibility and Control	129
Configuring Application Visibility and Control in a Wired Network	129
Enabling Application Recognition on an interface	129
Creating AVC QoS Policy	130
Applying a QoS Policy to the switch port	132
Configuring Wired AVC Flexible Netflow	133
NBAR2 Custom Applications	149
NBAR2 Dynamic Hitless Protocol Pack Upgrade	151
Monitoring Application Visibility and Control	153
Examples: Application Visibility and Control Configuration	154
Basic Troubleshooting - Questions and Answers	163
Additional References for Application Visibility and Control	164
Feature History for Application Visibility and Control in a Wired Network	165

CHAPTER 6**Configuring SDM Templates 167**

SDM Templates	167
---------------	-----

SDM Templates and Switch Stacks	167
How to Configure SDM Templates	168
Setting the SDM Template	168
Monitoring and Maintaining SDM Templates	169
Configuration Examples for SDM Templates	169
Examples: Displaying SDM Templates	169
Examples: Configuring SDM Templates	170
Additional References for SDM Templates	170
Feature History for SDM Templates	170
<hr/>	
CHAPTER 7	Configuring System Message Logs 173
Finding Feature Information	173
Information About Configuring System Message Logs	173
System Message Logging	173
System Log Message Format	174
Default System Message Logging Settings	175
Syslog Message Limits	175
How to Configure System Message Logs	176
Setting the Message Display Destination Device	176
Synchronizing Log Messages	177
Disabling Message Logging	179
Enabling and Disabling Time Stamps on Log Messages	180
Enabling and Disabling Sequence Numbers in Log Messages	180
Defining the Message Severity Level	181
Limiting Syslog Messages Sent to the History Table and to SNMP	182
Logging Messages to a UNIX Syslog Daemon	183
Monitoring and Maintaining System Message Logs	184
Monitoring Configuration Archive Logs	184
Configuration Examples for System Message Logs	184
Example: Stacking System Message	184
Example: Switch System Message	185
Feature History and Information For System Message Logs	185
<hr/>	
CHAPTER 8	Configuring Online Diagnostics 187

Information About Configuring Online Diagnostics	187
Generic Online Diagnostics (GOLD)	188
TestPortTxMonitoring	188
TestUnusedPortLoopback	188
How to Configure Online Diagnostics	189
Starting Online Diagnostic Tests	189
Configuring Online Diagnostics	190
Scheduling Online Diagnostics	190
Configuring Health-Monitoring Diagnostics	191
Monitoring and Maintaining Online Diagnostics	194
Configuration Examples for Online Diagnostic Tests	194
Examples: Start Diagnostic Tests	194
Example: Configure a Health Monitoring Test	194
Examples: Schedule Diagnostic Test	194
Examples: Displaying Online Diagnostics	194
Additional References for Online Diagnostics	196
Feature Information for Configuring Online Diagnostics	196

CHAPTER 9

Managing Configuration Files	199
Prerequisites for Managing Configuration Files	199
Restrictions for Managing Configuration Files	199
Information About Managing Configuration Files	199
Types of Configuration Files	199
Configuration Mode and Selecting a Configuration Source	200
Configuration File Changes Using the CLI	200
Location of Configuration Files	200
Copy Configuration Files from a Network Server to the Device	201
Copying a Configuration File from the Device to a TFTP Server	201
Copying a Configuration File from the Device to an RCP Server	201
Copying a Configuration File from the Device to an FTP Server	203
Copying files through a VRF	204
Copy Configuration Files from a Switch to Another Switch	204
Configuration Files Larger than NVRAM	205
Configuring the Device to Download Configuration Files	205

How to Manage Configuration File Information	206
Displaying Configuration File Information	206
Modifying the Configuration File	207
Copying a Configuration File from the Device to a TFTP Server	209
What to Do Next	210
Copying a Configuration File from the Device to an RCP Server	210
Examples	211
What to Do Next	211
Copying a Configuration File from the Device to the FTP Server	211
Examples	213
What to Do Next	213
Copying a Configuration File from a TFTP Server to the Device	213
What to Do Next	214
Copying a Configuration File from the rcp Server to the Device	215
Examples	216
What to Do Next	216
Copying a Configuration File from an FTP Server to the Device	216
Examples	218
What to Do Next	218
Maintaining Configuration Files Larger than NVRAM	218
Compressing the Configuration File	218
Storing the Configuration in Flash Memory on Class A Flash File Systems	220
Loading the Configuration Commands from the Network	222
Copying Configuration Files from Flash Memory to the Startup or Running Configuration	223
Copying Configuration Files Between Flash Memory File Systems	224
Copying a Configuration File from an FTP Server to Flash Memory Devices	225
What to Do Next	226
Copying a Configuration File from an RCP Server to Flash Memory Devices	226
Copying a Configuration File from a TFTP Server to Flash Memory Devices	227
Re-executing the Configuration Commands in the Startup Configuration File	228
Clearing the Startup Configuration	228
Deleting a Specified Configuration File	229
Specifying the CONFIG_FILE Environment Variable on Class A Flash File Systems	230
What to Do Next	232

Configuring the Device to Download Configuration Files	232
Configuring the Device to Download the Network Configuration File	232
Configuring the Device to Download the Host Configuration File	234
Feature History for Managing Configuration Files	235

CHAPTER 10**Configuration Replace and Configuration Rollback 237**

Prerequisites for Configuration Replace and Configuration Rollback	237
Restrictions for Configuration Replace and Configuration Rollback	238
Information About Configuration Replace and Configuration Rollback	238
Configuration Archive	238
Configuration Replace	239
Configuration Rollback	240
Configuration Rollback Confirmed Change	240
Benefits of Configuration Replace and Configuration Rollback	240
How to Use Configuration Replace and Configuration Rollback	241
Creating a Configuration Archive	241
Performing a Configuration Replace or Configuration Rollback Operation	242
Monitoring and Troubleshooting the Feature	244
Configuration Examples for Configuration Replace and Configuration Rollback	247
Creating a Configuration Archive	247
Replacing the Current Running Configuration with a Saved Cisco IOS Configuration File	247
Reverting to the Startup Configuration File	248
Performing a Configuration Replace Operation with the configure confirm Command	248
Performing a Configuration Rollback Operation	248
Feature History for Configuration Replace and Configuration Rollback	249

CHAPTER 11**Software Maintenance Upgrade 251**

Restrictions for Software Maintenance Upgrade	251
Information About Software Maintenance Upgrade	251
SMU Overview	251
SMU Workflow	252
SMU Package	252
SMU Reload	252
How to Manage Software Maintenance Updates	252

Installing an SMU Package	252
Managing an SMU Package	253
Configuration Examples for Software Maintenance Upgrade	254
Additional References for Software Maintenance Upgrade	258
Feature History for Software Maintenance Upgrade	258

CHAPTER 12

Working with the Flash File System	261
Finding Feature Information	261
Information About the Flash File System	261
Displaying Available File Systems	262
Setting the Default File System	264
Displaying Information About Files on a File System	264
Changing Directories and Displaying the Working Directory	266
Creating Directories	266
Removing Directories	267
Copying Files	267
Copying Files from One Device in a Stack to Another Device in the Same Stack	268
Deleting Files	269
Creating, Displaying and Extracting Files	270
Additional References for Flash File System	272
Feature History for Flash File System	273

CHAPTER 13

Factory Reset	275
Prerequisites for Performing Factory Reset	275
Limitations for Performing Factory Reset	275
Information About Factory Reset	275
How to Perform Factory Reset	276
Feature History and Information for Factory Reset	277

CHAPTER 14

Configuring Secure Storage	279
Information About Secure Storage	279
Enabling Secure Storage	279
Disabling Secure Storage	280
Verifying the Status of Encryption	281

Feature Information for Secure Storage 281

CHAPTER 15

Conditional Debug and Radioactive Tracing 283

Finding Feature Information 283

Introduction to Conditional Debugging 283

Introduction to Radioactive Tracing 284

How to Configure Conditional Debug and Radioactive Tracing 284

 Conditional Debugging and Radioactive Tracing 284

 Location of Tracefiles 284

 Configuring Conditional Debugging 285

 Radioactive Tracing for L2 Multicast 287

 Recommended Workflow for Trace files 287

 Copying tracefiles off the box 287

Monitoring Conditional Debugging 288

Configuration Examples for Conditional Debugging 288

Additional References for Conditional Debugging and Radioactive Tracing 289

Feature History for Conditional Debugging and Radioactive Tracing 290

CHAPTER 16

Troubleshooting the Software Configuration 291

Information About Troubleshooting the Software Configuration 291

 Software Failure on a Switch 291

 Lost or Forgotten Password on a Device 291

 Power over Ethernet Ports 292

 Disabled Port Caused by Power Loss 292

 Disabled Port Caused by False Link-Up 292

 Ping 293

 Layer 2 Traceroute 293

 Layer 2 Traceroute Guidelines 293

 IP Traceroute 294

 Debug Commands 295

 System Report 295

 Onboard Failure Logging on the Switch 297

 Fan Failures 298

 Possible Symptoms of High CPU Utilization 298

How to Troubleshoot the Software Configuration	298
Recovering from a Software Failure	298
Recovering from a Lost or Forgotten Password	302
Procedure with Password Recovery Enabled	304
Procedure with Password Recovery Disabled	305
Preventing Switch Stack Problems	307
Preventing Autonegotiation Mismatches	308
Troubleshooting SFP Module Security and Identification	308
Monitoring SFP Module Status	309
Executing Ping	309
Monitoring Temperature	309
Monitoring the Physical Path	309
Executing IP Traceroute	310
Redirecting Debug and Error Message Output	310
Using the show platform forward Command	310
Using the show debug command	311
Configuring OBFL	311
Verifying Troubleshooting of the Software Configuration	312
Displaying OBFL Information	312
Example: Verifying the Problem and Cause for High CPU Utilization	313
Scenarios for Troubleshooting the Software Configuration	314
Scenarios to Troubleshoot Power over Ethernet (PoE)	314
Configuration Examples for Troubleshooting Software	316
Example: Pinging an IP Host	316
Example: Performing a Traceroute to an IP Host	317
Feature History for Troubleshooting Software Configuration	318



CHAPTER 1

Administering the Device

- [Information About Administering the Device, on page 1](#)
- [How to Administer the Device, on page 10](#)
- [Configuration Examples for Device Administration, on page 39](#)
- [Additional References for Device Administration, on page 42](#)
- [Feature History for Device Administration, on page 42](#)

Information About Administering the Device

System Time and Date Management

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



Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Configuration Fundamentals Command Reference* on Cisco.com.

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 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.

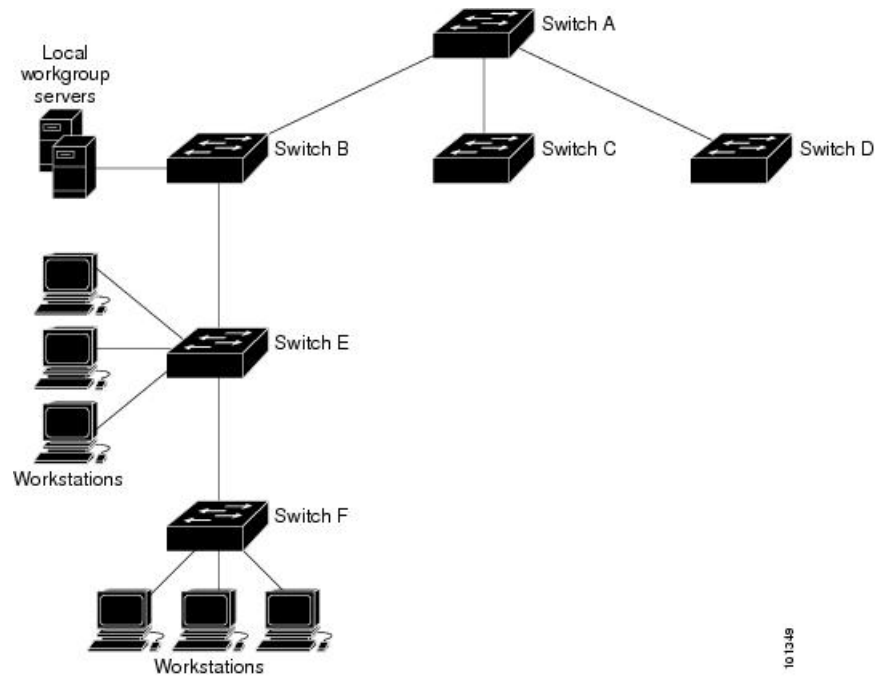
The communications between devices running NTP (known as associations) are usually statically configured; each device is given the IP address of all devices with which it should form associations. Accurate timekeeping is possible by exchanging NTP messages between each pair of devices with an association. 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.

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.

Cisco's 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 Figure shows a typical network example using NTP. Device A is the primary NTP, with the **Device B**, C, and D configured in NTP server mode, in server association with Device A. Device E is configured as an NTP peer to the upstream and downstream device, Device B and Device F, respectively.

Figure 1: Typical NTP Network Configuration



If the network is isolated from the Internet, Cisco's implementation of 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 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.

Poll-Based NTP Associations

Networking devices running NTP can be configured to operate in variety of association modes when synchronizing time with reference time sources. A networking device can obtain time information on a network in two ways—by polling host servers and by listening to NTP broadcasts. This section focuses on the poll-based association modes. Broadcast-based NTP associations are discussed in the *Broadcast-Based NTP Associations* section.

The following are the two most commonly used poll-based association modes:

- Client mode
- Symmetric active mode

The client and the symmetric active modes should be used when NTP is required to provide a high level of time accuracy and reliability.

When a networking device is operating in the client mode, it polls its assigned time-serving hosts for the current time. The networking device will then pick a host from among all the polled time servers to synchronize with. Because the relationship that is established in this case is a client-host relationship, the host will not capture or use any time information sent by the local client device. This mode is most suited for file-server and workstation clients that are not required to provide any form of time synchronization to other local clients. Use the **ntp server** command to individually specify the time server that you want your networking device to consider synchronizing with and to set your networking device to operate in the client mode.

When a networking device is operating in the symmetric active mode, it polls its assigned time-serving hosts for the current time and it responds to polls by its hosts. Because this is a peer-to-peer relationship, the host will also retain time-related information of the local networking device that it is communicating with. This mode should be used when a number of mutually redundant servers are interconnected via diverse network paths. Most stratum 1 and stratum 2 servers on the Internet adopt this form of network setup. Use the **ntp peer** command to individually specify the time serving hosts that you want your networking device to consider synchronizing with and to set your networking device to operate in the symmetric active mode.

The specific mode that you should set for each of your networking devices depends primarily on the role that you want them to assume as a timekeeping device (server or client) and the device's proximity to a stratum 1 timekeeping server.

A networking device engages in polling when it is operating as a client or a host in the client mode or when it is acting as a peer in the symmetric active mode. Although polling does not usually place a burden on memory and CPU resources such as bandwidth, an exceedingly large number of ongoing and simultaneous polls on a system can seriously impact the performance of a system or slow the performance of a given network. To avoid having an excessive number of ongoing polls on a network, you should limit the number of direct, peer-to-peer or client-to-server associations. Instead, you should consider using NTP broadcasts to propagate time information within a localized network.

Broadcast-Based NTP Associations

Broadcast-based NTP associations should be used when time accuracy and reliability requirements are modest and if your network is localized and has more than 20 clients. Broadcast-based NTP associations are also recommended for use on networks that have limited bandwidth, system memory, or CPU resources.

A networking device operating in the broadcast client mode does not engage in any polling. Instead, it listens for NTP broadcast packets that are transmitted by broadcast time servers. Consequently, time accuracy can be marginally reduced because time information flows only one way.

Use the **ntp broadcast client** command to set your networking device to listen for NTP broadcast packets propagated through a network. For broadcast client mode to work, the broadcast server and its clients must be located on the same subnet. You must enable the time server that transmits NTP broadcast packets on the interface of the given device by using the **ntp broadcast** command.

Authoritative NTP Server

An authoritative NTP server is a time server that can distribute time in the network. Other devices can configure it as a time server. You can configure a Cisco Catalyst 9000 Series Switch to act as an authoritative NTP server, enabling it to distribute time even when it is not synchronized to an outside time source. Use the **ntp master** command, in global configuration mode, to configure the device to be an authoritative NTP server.



Caution Use the **ntp master** command with caution. Usage of this command can override valid time sources, especially if a low stratum number is configured. Configuring multiple devices in the same network with the **ntp master** command can cause instability in timekeeping if the devices do not agree on the time.

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 Access Group

The access list-based restriction scheme allows you to grant or deny certain access privileges to an entire network, a subnet within a network, or a host within a subnet. To define an NTP access group, use the **ntp access-group** command in global configuration mode.

The access group options are scanned in the following order, from least restrictive to the most restrictive:

1. **ipv4** —Configures IPv4 access lists.
2. **ipv6** —Configures IPv6 access lists.
3. **peer** —Allows time requests and NTP control queries, and allows the system to synchronize itself to a system whose address passes the access list criteria.
4. **serve** —Allows time requests and NTP control queries, but does not allow the system to synchronize itself to a system whose address passes the access list criteria.
5. **serve-only** —Allows only time requests from a system whose address passes the access list criteria.
6. **query-only** —Allows only NTP control queries from a system whose address passes the access list criteria.

If the source IP address matches the access lists for more than one access type, the first type is granted access. If no access groups are specified, all access types are granted access to all systems. If any access groups are specified, only the specified access types will be granted access.

For details on NTP control queries, see RFC 1305 (NTP Version 3).

The encrypted NTP authentication scheme should be used when a reliable form of access control is required. Unlike the access list-based restriction scheme that is based on IP addresses, the encrypted authentication scheme uses authentication keys and an authentication process to determine if NTP synchronization packets sent by designated peers or servers on a local network are deemed as trusted before the time information that they carry along with them is accepted.

The authentication process begins from the moment an NTP packet is created. Cryptographic checksum keys are generated using the message digest algorithm 5 (MD5) and are embedded into the NTP synchronization packet that is sent to a receiving client. Once a packet is received by a client, its cryptographic checksum key is decrypted and checked against a list of trusted keys. If the packet contains a matching authentication key, the time-stamp information that is contained within the packet is accepted by the receiving client. NTP synchronization packets that do not contain a matching authenticator key are ignored.



Note In large networks, where many trusted keys must be configured, the Range of Trusted Key Configuration feature enables configuring multiple keys simultaneously.

It is important to note that the encryption and decryption processes used in NTP authentication can be very CPU-intensive and can seriously degrade the accuracy of the time that is propagated within a network. If your network setup permits a more comprehensive model of access control, you should consider the use of the access list-based form of control.

After NTP authentication is properly configured, your networking device will synchronize with and provide synchronization only to trusted time sources.

NTP Services on a Specific Interface

Network Time Protocol (NTP) services are disabled on all interfaces by default. NTP is enabled globally when any NTP commands are entered. You can selectively prevent NTP packets from being received through a specific interface by using the **ntp disable** command in interface configuration mode.

Source IP Address for NTP Packets

When the system sends an NTP packet, the source IP address is normally set to the address of the interface through which the NTP packet is sent. Use the **ntp source interface** command in global configuration mode to configure a specific interface from which the IP source address will be taken.

This interface will be used for the source address for all packets sent to all destinations. If a source address is to be used for a specific association, use the **source** keyword in the **ntp peer** or **ntp server** command.

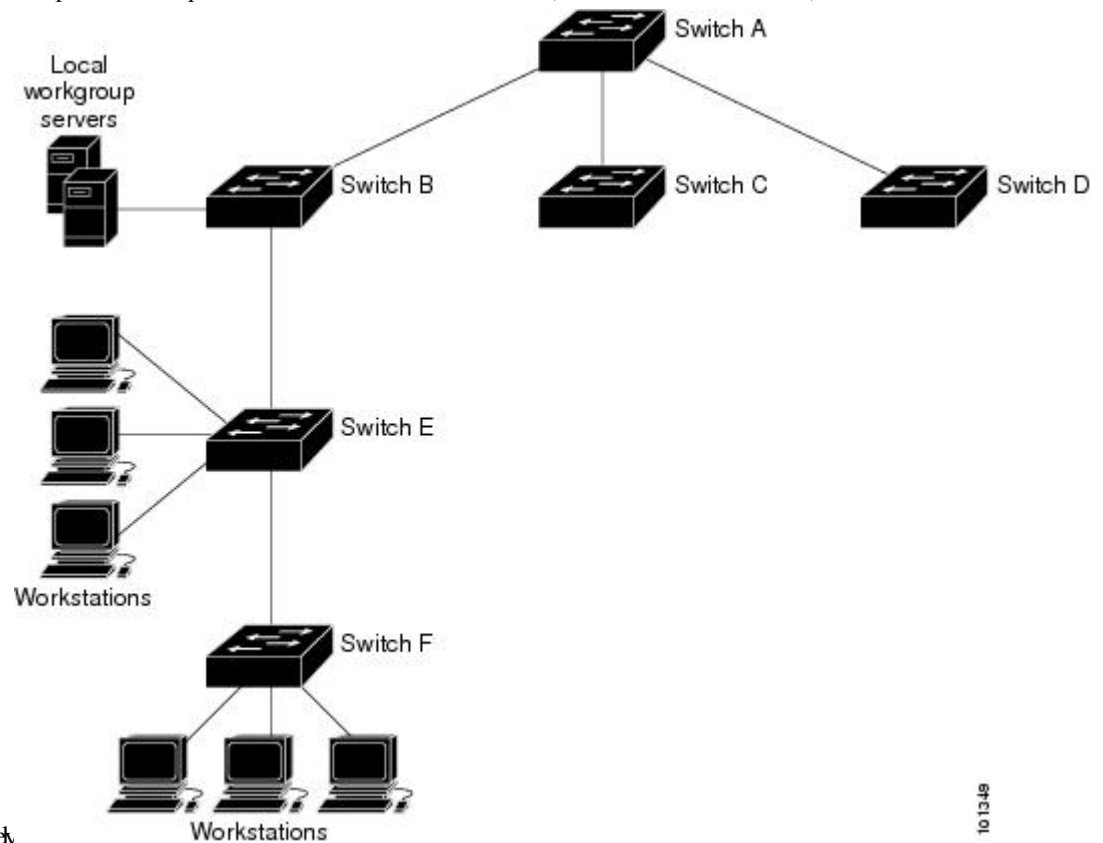
NTP Implementation

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.

Figure 2: Typical NTP Network Configuration

The following figure shows a typical network example using NTP. Switch A is the primary NTP, 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.

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.

System Name and Prompt

You configure the system name on the Device 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.

For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.4* and the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.4*.

Stack System Name and Prompt

If you are accessing a stack member through the active switch, you must use the **session** *stack-member-number* privileged EXEC command. The stack member number range is from 1 through 8. 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 device, 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 1: 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.



Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.4*.

Default Banner Configuration

The MOTD and login banners are not configured.

MAC Address Table

The MAC address table contains address information that the device 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 device 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 device 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).



Note For complete syntax and usage information for the commands used in this section, see the command reference for this release.

MAC Address Table Creation

With multiple MAC addresses supported on all ports, you can connect any port on the device to other network devices. The device 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 device 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 device maintains an address table for each VLAN, and STP can accelerate the aging interval on a per-VLAN basis.

The device sends packets between any combination of ports, based on the destination address of the received packet. Using the MAC address table, the device 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 device 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 Device 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 Device joins a switch stack, that Device 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 2: 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.

For CLI procedures, see the Cisco IOS Release 12.4 documentation on *Cisco.com*.

How to Administer the Device

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 device can synchronize, you do not need to manually set the system clock.

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.

Follow these steps to set the system clock:

SUMMARY STEPS

1. **enable**
2. 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	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	Use one of the following: <ul style="list-style-type: none"> • clock set <i>hh:mm:ss day month year</i> • clock set <i>hh:mm:ss month day year</i> Example: Device# clock set 13:32:00 23 March 2013	Manually set the system clock using one of these formats: <ul style="list-style-type: none"> • <i>hh:mm:ss</i>—Specifies the time in hours (24-hour format), minutes, and seconds. The time specified is relative to the configured time zone. • <i>day</i>—Specifies the day by date in the month. • <i>month</i>—Specifies the month by name. • <i>year</i>—Specifies the year (no abbreviation).

Configuring the Time Zone

Follow these steps to manually configure the time zone:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **clock timezone** *zone hours-offset [minutes-offset]*
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	clock timezone <i>zone hours-offset [minutes-offset]</i> Example: Device(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. <ul style="list-style-type: none"> • <i>zone</i>—Enters the name of the time zone to be displayed when standard time is in effect. The default is UTC. • <i>hours-offset</i>—Enters the hours offset from UTC. • (Optional) <i>minutes-offset</i>—Enters the minutes offset from UTC. This available where the local time zone is a percentage of an hour different from UTC.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

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. **enable**

2. **configure terminal**
3. **clock summer-time zone date date month year hh:mm date month year hh:mm [offset]**
4. **clock summer-time zone recurring [week day month hh:mm week day month hh:mm [offset]]**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	clock summer-time zone date date month year hh:mm date month year hh:mm [offset] Example: Device(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 4	clock summer-time zone recurring [week day month hh:mm week day month hh:mm [offset]] Example: Device(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. <ul style="list-style-type: none"> • <i>zone</i>—Specifies the name of the time zone (for example, PDT) to be displayed when summer time is in effect. • (Optional) <i>week</i>— Specifies the week of the month (1 to 4, first, or last). • (Optional) <i>day</i>—Specifies the day of the week (Sunday, Monday...).

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) <i>month</i>—Specifies the month (January, February...). • (Optional) <i>hh:mm</i>—Specifies the time (24-hour format) in hours and minutes. • (Optional) <i>offset</i>—Specifies the number of minutes to add during summer time. The default is 60.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Follow these steps if summer time in your area does not follow a recurring pattern (configure the exact date and time of the next summer time events):

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **clock summer-time zone date [month date year hh:mm month date year hh:mm [offset]] or clock summer-time zone date [date month year hh:mm date month year hh:mm [offset]]**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	clock summer-time zone date [<i>month date year hh:mm month date year hh:mm [offset]</i>] or clock summer-time zone date [<i>date month year hh:mm date month year hh:mm [offset]</i>]	Configures summer time to start on the first date and end on the second date. Summer time is disabled by default. <ul style="list-style-type: none"> • For <i>zone</i>, specify the name of the time zone (for example, PDT) to be displayed when summer time is in effect. • (Optional) For <i>week</i>, specify the week of the month (1 to 5 or last). • (Optional) For <i>day</i>, specify the day of the week (Sunday, Monday...). • (Optional) For <i>month</i>, specify the month (January, February...). • (Optional) For <i>hh:mm</i>, specify the time (24-hour format) in hours and minutes. • (Optional) For <i>offset</i>, specify the number of minutes to add during summer time. The default is 60.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring NTP

These following sections provide configuration information on NTP:

Default NTP Configuration

shows the default NTP configuration.

Table 3: Default NTP Configuration

Feature	Default Setting
NTP authentication	Disabled. No authentication key is specified.
NTP peer or server associations	None configured.
NTP broadcast service	Disabled; no interface sends or receives NTP broadcast packets.
NTP access restrictions	No access control is specified.
NTP packet source IP address	The source address is set by the outgoing interface.

NTP is enabled on all interfaces by default. All interfaces receive NTP packets.

Configuring NTP Authentication

To configure NTP authentication, perform this procedure:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **[no] ntp authenticate**
4. **[no] ntp authentication-key *number* md5 *value***
5. **[no] ntp trusted-key *key-number***
6. **[no] ntp server *ip-address* key *key-id* [prefer]**
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	<p>[no] ntp authenticate</p> <p>Example:</p> <pre>Device(config)# ntp authenticate</pre>	<p>Enables NTP authentication.</p> <p>Use the no form of this command to disable NTP authentication</p>
Step 4	<p>[no] ntp authentication-key <i>number</i> md5 <i>value</i></p> <p>Example:</p> <pre>Device(config)# ntp authentication-key 42 md5 aNiceKey</pre>	<p>Defines the authentication keys.</p> <ul style="list-style-type: none"> • Each key has a key number, a type, and a value. <p>Use the no form of this command to remove authentication key.</p>
Step 5	<p>[no] ntp trusted-key <i>key-number</i></p> <p>Example:</p> <pre>Device(config)# ntp trusted-key 42</pre>	<p>Defines trusted authentication keys that a peer NTP device must provide in its NTP packets for this device to synchronize to it.</p> <p>Use the no form of this command to disable trusted authentication.</p>
Step 6	<p>[no] ntp server <i>ip-address</i> key <i>key-id</i> [prefer]</p> <p>Example:</p> <pre>Device(config)# ntp server 172.16.22.44 key 42</pre>	<p>Allows the software clock to be synchronized by an NTP time server.</p> <ul style="list-style-type: none"> • <i>ip-address</i>: The IP address of the time server providing the clock synchronization. • <i>key-id</i>: Authentication key defined with the ntp authentication-key command. • prefer: Sets this peer as the preferred one that provides synchronization. This keyword reduces clock hop among peers. <p>Use the no form of this command to remove a server association.</p>
Step 7	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Exits global configuration mode and returns to privileged EXEC mode.</p>

Configuring Poll-Based NTP Associations

To configure poll-based NTP associations, perform this procedure:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **[no] ntp peer** *ip-address* [**version number**] [**key key-id**] [**source interface**] [**prefer**]
4. **[no] ntp server** [**vrf vrf-name**] *ip-address* [**version number**] [**key key-id**] [**source interface**] [**prefer**]
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	[no] ntp peer <i>ip-address</i> [version number] [key key-id] [source interface] [prefer] Example: Device(config)# ntp peer 172.16.22.44 version 2	Configures the device system clock to synchronize a peer or to be synchronized by a peer (peer association). <ul style="list-style-type: none"> • <i>ip-address</i>: The IP address of the peer providing or being provided, the clock synchronization. • <i>number</i>: NTP version number. The range is 1 to 3. By default, version 3 is selected. • <i>key-id</i>: Authentication key defined with the ntp authentication-key command. • <i>interface</i>: The interface from which to pick the IP source address. By default, the source IP address is taken from the outgoing interface. • prefer: Sets this peer as the preferred one that provides synchronization. This keyword reduces switching back and forth between peers. Use the no form of this command to remove a peer association.
Step 4	[no] ntp server [vrf vrf-name] <i>ip-address</i> [version number] [key key-id] [source interface] [prefer] Example: Device(config)# ntp server 172.16.22.44 version 2	Configures the device's system clock to be synchronized by a time server (server association). <ul style="list-style-type: none"> • <i>vrf-name</i>: The virtual routing and forwarding (VRF) address of the server providing the clock synchronization.

	Command or Action	Purpose
		<p>Note Before you configure this command, the VRF must be configured.</p> <ul style="list-style-type: none"> • <i>ip-address</i>: The IP address of the time server providing the clock synchronization. • <i>number</i>: NTP version number. The range is 1 to 3. By default, version 3 is selected. • <i>key-id</i>: Authentication key defined with the ntp authentication-key command. • <i>interface</i>: The interface from which to pick the IP source address. By default, the source IP address is taken from the outgoing interface. • prefer: Sets this peer as the preferred one that provides synchronization. This keyword reduces clock hop among peers. <p>Use the no form of this command to remove a server association.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.

Configuring Broadcast-Based NTP Associations

To configure broadcast-based NTP associations, perform this procedure:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **[no] ntp broadcast** [**version** *number*] [**key** *key-id*] [*destination-address*]
5. **[no] ntp broadcast client**
6. **exit**
7. **[no] ntp broadcastdelay** *microseconds*
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p>	<p>Enables privileged EXEC mode.</p> <p>Enter your password if prompted.</p>

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/1	Configures an interface and enters interface configuration mode.
Step 4	[no] ntp broadcast [version <i>number</i>] [key <i>key-id</i>] [destination-address] Example: Device(config-if)# ntp broadcast version 2	Enables the interface to send NTP broadcast packets to a peer. <ul style="list-style-type: none"> • <i>number</i>: NTP version number. The range is 1 to 3. By default, version 3 is used. • <i>key-id</i>: Authentication key. • <i>destination-address</i>: IP address of the peer that is synchronizing its clock to this switch. <p>Use the no form of this command to disable the interface from sending NTP broadcast packets.</p>
Step 5	[no] ntp broadcast client Example: Device(config-if)# ntp broadcast client	Enables the interface to receive NTP broadcast packets. Use the no form of this command to disable the interface from receiving NTP broadcast packets.
Step 6	exit Example: Device(config-if)# exit	Returns to privileged EXEC mode.
Step 7	[no] ntp broadcastdelay <i>microseconds</i> Example: Device(config)# ntp broadcastdelay 100	(Optional) Change the estimated round-trip delay between the device and the NTP broadcast server The default is 3000 microseconds. The range is from 1 to 999999. Use the no form of this command to disable the interface from receiving NTP broadcast packets.

	Command or Action	Purpose
Step 8	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.

Configuring NTP Access Restrictions

You can control NTP access on two levels as described in these sections:

Creating an Access Group and Assigning a Basic IP Access List

To create an access group and assign a basic IP access list, perform this procedure:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **[no] ntp access-group {query-only | serve-only | serve | peer} access-list-number**
4. **access-list access-list-number permit source [source-wildcard]**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	[no] ntp access-group {query-only serve-only serve peer} access-list-number Example: <pre>Device(config)# ntp access-group peer 99</pre>	Create an access group, and apply a basic IP access list.. <ul style="list-style-type: none"> • query-only: NTP control queries. • serve-only: Time requests. • serve: Allows time requests and NTP control queries, but does not allow the device to synchronize to the remote device. • peer: Allows time requests and NTP control queries and allows the device to synchronize to the remote device.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • <i>access-list-number</i>: IP access list number. The range is from 1 to 99. <p>Use the no form of this command to remove access control to the switch NTP services.</p>
Step 4	<p>access-list <i>access-list-number</i> permit <i>source</i> [<i>source-wildcard</i>]</p> <p>Example:</p> <pre>Device(config)# access-list 99 permit 172.20.130.5</pre>	<p>Create the access list.</p> <ul style="list-style-type: none"> • <i>access-list-number</i>: IP access list number. The range is from 1 to 99. • permit: Permits access if the conditions are matched. • <i>source</i>: IP address of the device that is permitted access to the device. • <i>source-wildcard</i>: Wildcard bits to be applied to the source. <p>Note When creating an access list, remember 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.</p> <p>Use the no form of this command to remove authentication key.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>

Disabling NTP Services on a Specific Interface

To disable NTP packets from being received on an interface, perform this procedure:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **[no] ntp disable**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p>	<p>Enables privileged EXEC mode.</p> <p>Enter your password if prompted.</p>

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/1	Enters global configuration mode.
Step 4	[no] ntp disable Example: Device(config-if)# ntp disable	Disables NTP packets from being received on the interface. Use the no form of this command to re-enable receipt of NTP packets on an interface.
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Configuring a System Name

Follow these steps to manually configure a system name:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **hostname *name***
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	hostname name Example: Device(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 4	end Example: remote-users(config)# end remote-users#	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Setting Up DNS

If you use the device 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.

Follow these steps to set up your switch to use the DNS:

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **ip domain-name** *name*
4. **ip name-server** *server-address1* [*server-address2* ... *server-address6*]
5. **ip domain-lookup** [*nsap* | *source-interface interface*]
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip domain-name <i>name</i> Example: Device(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 device 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 4	ip name-server <i>server-address1</i> [<i>server-address2</i> ... <i>server-address6</i>] Example: Device(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 device sends DNS queries to the primary server first. If that query fails, the backup servers are queried.
Step 5	ip domain-lookup [<i>nsap</i> <i>source-interface interface</i>] Example: Device(config)# ip domain-lookup	(Optional) Enables DNS-based hostname-to-address translation on your device. This feature is enabled by default. 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

	Command or Action	Purpose
		identify your devices by using the global Internet naming scheme (DNS).
Step 6	end Example: Device(config) # end	Returns to privileged EXEC mode.
Step 7	show running-config Example: Device# show running-config	Verifies your entries.
Step 8	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

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 device

Follow these steps to configure a MOTD login banner:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **banner motd *c message c***
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	banner motd c message c Example: <pre>Device(config)# banner motd # This is a secure site. Only authorized users are allowed. For access, contact technical support. #</pre>	Specifies the message of the day. <i>c</i> —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. <i>message</i> —Enters a banner message up to 255 characters. You cannot use the delimiting character in the message.
Step 4	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 6	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

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.

Follow these steps to configure a login banner:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **banner login c message c**
4. **end**
5. **show running-config**

6. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	banner login c message c Example: Device(config)# banner login \$ Access for authorized users only. Please enter your username and password. \$	Specifies the login message. <p><i>c</i>— 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.</p> <p><i>message</i>—Enters a login message up to 255 characters. You cannot use the delimiting character in the message.</p>
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Managing the MAC Address Table

Changing the Address Aging Time

Follow these steps to configure the dynamic address table aging time:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `mac address-table aging-time [0 | 10-1000000] [routed-mac | vlan vlan-id]`
4. `end`
5. `show running-config`
6. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	mac address-table aging-time [0 10-1000000] [routed-mac vlan vlan-id] Example: Device(config)# <code>mac address-table aging-time 500 vlan 2</code>	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. <i>vlan-id</i> —Valid IDs are 1 to 4094.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.

	Command or Action	Purpose
Step 6	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring MAC Address Change Notification Traps

Follow these steps to configure the switch to send MAC address change notification traps to an NMS host:

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	snmp-server host <i>host-addr</i> <i>community-string</i> <i>notification-type</i> { informs traps } { version { 1 2c 3 } } { vrf <i>vrf instance name</i> } Example: <pre>Device(config)# snmp-server host 172.20.10.10 traps private mac-notification</pre>	Specifies the recipient of the trap message. <ul style="list-style-type: none"> • 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
		<ul style="list-style-type: none"> • 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.
Step 4	snmp-server enable traps mac-notification change Example: <pre>Device(config)# snmp-server enable traps mac-notification change</pre>	Enables the device to send MAC address change notification traps to the NMS.
Step 5	mac address-table notification change Example: <pre>Device(config)# mac address-table notification change</pre>	Enables the MAC address change notification feature.
Step 6	mac address-table notification change [interval value] [history-size value] Example: <pre>Device(config)# mac address-table notification change interval 123 Device(config)# mac address-table notification change history-size 100</pre>	<p>Enters the trap interval time and the history table size.</p> <ul style="list-style-type: none"> • (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 7	interface interface-id Example: <pre>Device(config)# interface gigabitethernet1/0/2</pre>	Enters interface configuration mode, and specifies the Layer 2 interface on which to enable the SNMP MAC address notification trap.
Step 8	snmp trap mac-notification change {added removed} Example: <pre>Device(config-if)# snmp trap mac-notification change added</pre>	<p>Enables the MAC address change notification trap on the interface.</p> <ul style="list-style-type: none"> • Enables the trap when a MAC address is added on this interface.

	Command or Action	Purpose
		<ul style="list-style-type: none"> Enables the trap when a MAC address is removed from this interface.
Step 9	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 10	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 11	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

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.

Follow these steps to configure the device to send MAC address-move notification traps to an NMS host:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server host *host-addr* {traps | informs} {version {1 | 2c | 3}} *community-string notification-type***
4. **snmp-server enable traps mac-notification move**
5. **mac address-table notification mac-move**
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted.

	Command or Action	Purpose
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	<p>snmp-server host <i>host-addr</i> {traps informs} {version {1 2c 3}} <i>community-string notification-type</i></p> <p>Example:</p> <pre>Device(config)# snmp-server host 172.20.10.10 traps private mac-notification</pre>	<p>Specifies the recipient of the trap message.</p> <ul style="list-style-type: none"> • <i>host-addr</i>—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. • <i>community-string</i>—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. • <i>notification-type</i>—Uses the mac-notification keyword.
Step 4	<p>snmp-server enable traps mac-notification move</p> <p>Example:</p> <pre>Device(config)# snmp-server enable traps mac-notification move</pre>	Enables the device to send MAC address move notification traps to the NMS.
Step 5	<p>mac address-table notification mac-move</p> <p>Example:</p> <pre>Device(config)# mac address-table notification mac-move</pre>	Enables the MAC address move notification feature.
Step 6	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.

	Command or Action	Purpose
Step 8	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

To disable MAC address-move notification traps, use the **no snmp-server enable traps mac-notification move** global configuration command. To disable the MAC address-move notification feature, use the **no mac address-table notification mac-move** global configuration command.

You can verify your settings by entering the **show mac address-table notification mac-move** privileged EXEC commands.

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.

Follow these steps to configure the switch to send MAC address table threshold notification traps to an NMS host:

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	<p>snmp-server host <i>host-addr</i> {traps / informs} {version {1 2c 3}} <i>community-string notification-type</i></p> <p>Example:</p> <pre>Device(config)# snmp-server host 172.20.10.10 traps private mac-notification</pre>	<p>Specifies the recipient of the trap message.</p> <ul style="list-style-type: none"> • <i>host-addr</i>—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. • <i>community-string</i>—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. • <i>notification-type</i>—Uses the mac-notification keyword.
Step 4	<p>snmp-server enable traps mac-notification threshold</p> <p>Example:</p> <pre>Device(config)# snmp-server enable traps mac-notification threshold</pre>	Enables MAC threshold notification traps to the NMS.
Step 5	<p>mac address-table notification threshold</p> <p>Example:</p> <pre>Device(config)# mac address-table notification threshold</pre>	Enables the MAC address threshold notification feature.
Step 6	<p>mac address-table notification threshold [limit <i>percentage</i>] [interval <i>time</i>]</p> <p>Example:</p> <pre>Device(config)# mac address-table notification threshold interval 123 Device(config)# mac address-table notification threshold limit 78</pre>	<p>Enters the threshold value for the MAC address threshold usage monitoring.</p> <ul style="list-style-type: none"> • (Optional) limit <i>percentage</i>—Specifies the percentage of the MAC address table use; valid values are from 1 to 100 percent. The default is 50 percent. • (Optional) interval <i>time</i>—Specifies the time between notifications; valid values are greater than or equal to 120 seconds. The default is 120 seconds.
Step 7	<p>end</p> <p>Example:</p>	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config)# end	
Step 8	show running-config Example: Device# show running-config	Verifies your entries.
Step 9	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Adding and Removing Static Address Entries

Follow these steps to add a static address:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mac address-table static *mac-addr* vlan *vlan-id* interface *interface-id***
4. **show running-config**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	mac address-table static <i>mac-addr</i> vlan <i>vlan-id</i> interface <i>interface-id</i> Example: Device(config)# mac address-table	Adds a static address to the MAC address table. <ul style="list-style-type: none"> • <i>mac-addr</i>—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.

	Command or Action	Purpose
	<pre>static c2f3.220a.12f4 vlan 4 interface gigabitethernet 1/0/1</pre>	<ul style="list-style-type: none"> • <i>vlan-id</i>—Specifies the VLAN for which the packet with the specified MAC address is received. Valid VLAN IDs are 1 to 4094. • <i>interface-id</i>—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 4	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 5	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Unicast MAC Address Filtering

Follow these steps to configure the Device to drop a source or destination unicast static address:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mac address-table static *mac-addr* vlan *vlan-id* drop**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	mac address-table static mac-addr vlan vlan-id drop Example: <pre>Device(config)# mac address-table static c2f3.220a.12f4 vlan 4 drop</pre>	Enables unicast MAC address filtering and configure the device to drop a packet with the specified source or destination unicast static address. <ul style="list-style-type: none"> • <i>mac-addr</i>—Specifies a source or destination unicast MAC address (48-bit). Packets with this MAC address are dropped. • <i>vlan-id</i>—Specifies the VLAN for which the packet with the specified MAC address is received. Valid VLAN IDs are 1 to 4094.
Step 4	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 6	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Monitoring and Maintaining Administration of the Device

Command	Purpose
clear mac address-table dynamic	Removes all dynamic entries.
clear mac address-table dynamic address <i>mac-address</i>	Removes a specific MAC address.
clear mac address-table dynamic interface <i>interface-id</i>	Removes all addresses on the specified physical port or port channel.
clear mac address-table dynamic vlan <i>vlan-id</i>	Removes all addresses on a specified VLAN.

Command	Purpose
show clock [<i>detail</i>]	Displays the time and date configuration.
show ip igmp snooping groups	Displays the Layer 2 multicast entries for all VLANs or the specified VLAN.
show mac address-table address <i>mac-address</i>	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 <i>interface-name</i>	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 <i>vlan-id</i>	Displays the MAC address table information for the specified VLAN.

Configuration Examples for Device Administration

Example: Setting the System Clock

This example shows how to manually set the system clock:

```
Device# 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:

```
Device(config)# clock summer-time PDT recurring PST date
20 March 2013 2:00 3 November 2013 2:00
```

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

```
Device(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:

```
Device(config)# banner motd #
This is a secure site. Only authorized users are allowed.
For access, contact technical support.
#
Device(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:

```
Device(config)# banner login $
Access for authorized users only. Please enter your username and password.
$
Device(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:

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

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



Note You cannot associate the same static MAC address to multiple interfaces. If the command is executed again with a different interface, the static MAC address is overwritten on the new interface.

```
Device(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:

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

Additional References for Device Administration

Related Documents

Related Topic	Document Title
System management commands	<i>Command Reference (Catalyst 9300 Series Switches)</i>

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
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>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.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History for Device Administration

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Everest 16.5.1a	Device Administration	The device administration allows to configure the system time and date, system name, a login banner, and set up the DNS.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 2

Boot Integrity Visibility

- [Information About Boot Integrity Visibility, on page 43](#)
- [Verifying the Software Image and Hardware, on page 43](#)
- [Verifying Platform Identity and Software Integrity, on page 44](#)
- [Additional References for Boot Integrity Visibility, on page 46](#)
- [Feature History for Boot Integrity Visibility, on page 47](#)

Information About Boot Integrity Visibility

Boot Integrity Visibility allows Cisco's platform identity and software integrity information to be visible and actionable. Platform identity provides the platform's manufacturing installed identity. Software integrity exposes boot integrity measurements that can be used to assess whether the platform has booted trusted code.

During the boot process, the software creates a checksum record of each stage of the bootloader activities.

You can retrieve this record and compare it with a Cisco-certified record to verify if your software image is genuine. If the checksum values do not match, you may be running a software image that is either not certified by Cisco or has been altered by an unauthorized party.



Note Boot Integrity Visibility is supported only on the active supervisor. It does not support high availability scenarios.

Verifying the Software Image and Hardware

This task describes how to retrieve the checksum record that was created during a switch bootup. Enter the following commands in privileged EXEC mode.



Note On executing the following commands, you might see the message **% Please Try After Few Seconds** displayed on the CLI. This does not indicate a CLI failure, but indicates setting up of underlying infrastructure required to get the required output. We recommend waiting for a few minutes and then try the command again.

The messages **% Error retrieving SUDI certificate** and **% Error retrieving integrity data** signify a real CLI failure.

SUMMARY STEPS

1. **show platform sudi certificate** [**sign** [**nonce** *nonce*]]
2. **show platform integrity** [**sign** [**nonce** *nonce*]]

DETAILED STEPS

	Command or Action	Purpose
Step 1	show platform sudi certificate [sign [nonce <i>nonce</i>]] Example: Device# show platform sudi certificate sign nonce 123	Displays checksum record for the specific SUDI. <ul style="list-style-type: none"> • (Optional) sign - Show signature • (Optional) nonce - Enter a nonce value
Step 2	show platform integrity [sign [nonce <i>nonce</i>]] Example: Device# show platform integrity sign nonce 123	Displays checksum record for boot stages. <ul style="list-style-type: none"> • (Optional) sign - Show signature • (Optional) nonce - Enter a nonce value

Verifying Platform Identity and Software Integrity

Verifying Platform Identity

The following example displays the Secure Unique Device Identity (SUDI) chain in PEM format. The first certificate is the Cisco Root CA 2048 and the second is the Cisco subordinate CA (ACT2 SUDI CA). Both certificates can be verified to match those published on <https://www.cisco.com/security/pki/>. The third is the SUDI certificate.

```

Device# show platform sudi certificate sign nonce 123
-----BEGIN CERTIFICATE-----
MIIDQzCCAiugAwIBAgIQX/h7KCTU3I1CoxW1aMmt/zANBgkqhkiG9w0BAQUFADA1
MRYwFAYDVQQKEw1DaXNjbyBTeXN0ZW1zMRswGQYDVQQDExJDaXNjbyBSb290IENB
IDIwNDgwHhcNMDQwNTE0MjAxNzEyWhcNMjkwNTE0MjAyNTQyWjA1MRYwFAYDVQQK
Ew1DaXNjbyBTeXN0ZW1zMRswGQYDVQQDExJDaXNjbyBSb290IENBIDIwNDgwggEg
MA0GCSqGSIb3DQEBAQUAA4IBDQAwggEIAoIBAQCwmrmrp68Kd6ficba0ZmKUeIhH
xmJVhEAYv8CrLqUccda8bnuoqrpu0hWISEWdovyD0My5jOAmahBKeN8hF570YQXJ
FcjPFto1YYmUQ6iEqDGYeJu5Tm8sUxJsZR2tKys7McQr/4NEb7Y9JHcJ6r8qqB9q
VvYgDxFU14F1pyXOWWqCZe+36ufijXWlLvLdT6ZeYpzPEApk0E5tzivMW/VgpSdh
jWn0f84bcN5wGyDwbs2mAag8EtKpP6BrXruOIIt6ke01a06g58QBdKhTCytKmg9l
Eg6CTY5j/e/rmxrbU6YTYK/CfdfHbBcl1HP7R2RQgYCUTOG/rksc35LtLgXfAgED
o1EwTzALBgNVHQ8EBAMCAYYwDwYDVR0TAQH/BAUwAwEB/zAdBgNVHQ4EFgQUJ/PI
FR5umgIJFq0roIlgX9p7L6owEAYJKwYBBAGCNxUBBAMCAQAwDQYJKoZIhvcNAQEF
BQADggEBAJ2dhISjQa18dwy3U8pORFbi71R803UXHOjgkxhLtv5MOhmBvrbW7hmW
Yqpa02TB9k5UM8Z3/sUcuVdJcr18JOagxEu5sv4dEX+5wW4q+ffY0vhN4TauYuX
cB7w4ovXsNgOnbFp1iqRe6lJT37mjpxYgyc81WhJdTsd9i7rp77rMKSH0T8Lasz
Bvt9YAretIpjsJyp8qS5UwGH0GikJ3+r/+n6yUA4iGe00caEb1fJU9u6ju7AQ7L4
CYNu/2bPPu8Xs1gYJQk0XuPL1hS27PKSb3TkL4Eq1ZKR4OCXPDJoBYVL0fdX41Id

```

```
kxpUnwVwwEpxYB5DC2Ae/qPOgRnhCzU=
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIEPDCCAYsgAwIBAgIKYQlufQAAAAAADANBgkqhkiG9w0BAQUFADA1MRYwFAFD
VQQKEw1DaXNjbyBTeXN0ZW1zMRSwGQYDVQQDExJDaXNjbyBsb290IENBTDIwNDgw
HhcnMTEwNjMwMTc1NjU3WHcnMjkwNTE0MjYNTQyYwJmMQ4wDAYDVQQKEwVDAxNj
bzEVM2MGA1UEAaxMMQUNUMiBTUVURJiENBMTiBiIjANBgkqhkiG9w0BAQEFAAOCAQ8A
MIIBCgKCAQEAOm513THixA9tN/hS5qR/6UZRpdd+9aE2JbFknJht6gfHKd477Aks
5XAtUs5oxDYvt/zEbs1Zq3+LR6qrqKQVU6JYvH05UYLBqCj38s76NLk53905WzP
9pRcmRCPUX+a6tHF/QRuOiJ44mdeDYzo3qPCpxzprWJDPclM4iYKHumMQmqm+
xghHiooWS80BOcdiyeEbeP5rZ7qRuewKmp11TiI3WdBNJZjnfpj66F+P4SaDkGb
BXGgJ130Vef+EyFwLrFjj97fL2+8oauV43Qrvnf3d/GfQXj7ew+z/sX1XtEOjSXJ
URsYMEj53Rdd9tJwhky8neapsz+s+r+kdVQIDAQABo4IBWjCCAVYwCwYDVR0PBAQD
AgHGMB0GA1UdDgQWBRI2PHxwDVM7t8cwmTr7i4MAP4fzAfBgNVHSMGDAWgBQN
88gVHM6aAgkWrSuglWf2nsVqjBDBgNVHR8EPDA6MDIqNqA0hJodHRWoi8vd3d3
LmNpc2NvLmNvbS9zZW51cm10eS9wa2kvY3JsL2NyY2EyMDQ4LmNybDBQBGRBGEF
BQcBAQREMEIwQAYIKwYBBQUHMAKGNH0dHA6Ly93d3cuY2l2Y28uY29tL3NlY3VyaXR5
aXR5L3BraS9jZlJ0cy9jcmNhmjA0OC5jZXIwXAYDVR0gBFUwUzBRBgorBgEEAQKv
AQwAMEMwQQYIKwYBBQUHAgEWNW0dHA6Ly93d3cuY2l2Y28uY29tL3NlY3VyaXR5
L3BraS9w2xpY211cy9pbmRlcC5odG1sMBIGA1UdEwEB/wQIMAYBAf8CAQAwdQYJ
KoZiHvcNAQEFBQADggEBAGhlqclr9tx4hzWgDERm371yeuEmqIifi9b9+GbsJbi
ZHc/CcCl0lJu0a9zTXA9w47H9/t61eduGxb4WeLxcwCiUgvFtCa51lkl28nBcKY
/4dwlex+7amATUQ04QggIE67wVlPu6bgAE3Ja/nRS3xKYSnj8H5TehimBSv6TEci
i5jUhOwryAK4dVo8hCjkjEku3ufBTJapnv89g90E+H3VKM4L+/KdkUO+52djFKn
hy147d7cZR4DY4LIuFM2P1As8YyjoZnpK/urSRI14WdIlplRlnH7KND15618yfVP
0IFJZBGrooCRBjOSwFv8cpWCbmWdPacQT2nwIjTfy8c=
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIDezCCAmOgAwIBAgIEAc+JiTANBgkqhkiG9w0BAQsFADANMQ4wDAYDVQQKEwVD
aXNjbyEVMBMGA1UEAaxMMQUNUMiBTUVURJiENBMB4XDTE3MDg3OjEwNDMzOV0eXDTI3
MDg3OjEwNDMzOV0wZzEmMCQGA1UEBRMduE1EOKM5MzAwLTI0VGVggU046RkNXMjEz
NEwWMEMxMjIwY28uY29tL3NlY3VyaXR5L3BraS9w2xpY211cy9pbmRlcC5odG1sMBIGA1
UEBMBAGf8EAgA1UdEQRGMESgQqYJKwYBBAEFQIDoDUTM0NoaXBjRjR1VWUwPmVzNEZRT0xSbkpwsUUxaGNpQXhNMQ0F4Tnpvd05Eb3hNeUfiY1FjPTANBgkq
hkiG9w0BAQsFAAOCQAQEASXX+iZLMvHQIR1/s1Pobm0Kp/bYeHsgdTRQPRHbcM1HH
ROfjJdaJMHCspB17XtcLkNNFowyUEkjoepYHjpxxhkeGIqgD6xt4rW6v/058Haw6
QbAhJFGZriVxFObvW20VQ4ezYaGogA+0I2GZqD/ZggUY6zsvVwKme6inoEgXcYap
5GqF4weEoty9u+OKqr3ppWU4751xnNm/h+WHbNtunL6r7wZfe5dFQIxR5QP5gwRa
svpSsCoKm6PiwIUhw25CvtZ9NTg0tu5t5D7aVcxLeR8XbAlpjfgxw/RtSsjNse3+
ZkQgJUESqlxwzxcGULY+vDINyRQ/sP6y7cT+niT00A==
-----END CERTIFICATE-----
```

Signature version: 1
Signature:

Verifying Software Integrity

The following example displays the checksum record for the boot stages. The hash measurements are displayed for each of the three stages of software successively booted. These hashes can be compared against Cisco-provided reference values. An option to sign the output gives a verifier the ability to ensure the output is genuine and is not altered. A nonce can be provided to protect against replay attacks.

Feature History for Boot Integrity Visibility

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Fuji 16.8.1a	Boot Integrity Visibility	Boot Integrity Visibility allows Cisco's platform identity and software integrity information to be visible and actionable. Platform identity provides the platform's manufacturing installed identity.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 3

Performing Device Setup Configuration

- [Restrictions for Software Install, on page 49](#)
- [Information About Performing Device Setup Configuration, on page 49](#)
- [How to Perform Device Setup Configuration, on page 59](#)
- [Monitoring Device Setup Configuration, on page 74](#)
- [Configuration Examples for Performing Device Setup, on page 78](#)
- [Additional References For Performing Device Setup, on page 93](#)
- [Feature History for Performing Device Setup Configuration, on page 93](#)

Restrictions for Software Install

- Subpackage installation is not supported.

Information About Performing Device Setup Configuration

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

Device Boot Process

To start your device, you need to follow the procedures in the hardware installation guide for installing and powering on the device and setting up the initial device configuration.

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

- 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 device.

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.

Before you can assign device 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 device 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 Install Overview

The Software Install feature provides a uniform experience across different types of upgrades, such as full image install, Software Maintenance Upgrade (SMU), In-Service Software Upgrade (ISSU) and In-Service Model Update (data model package).

The Software Install feature facilitates moving from one version of the software to another version in install mode. Use the **install** command in privileged EXEC mode to install or upgrade a software image. You can also downgrade to a previous version of the software image, using the install mode.

The method that you use to upgrade Cisco IOS XE software depends on whether the switch is running in install mode or in bundle mode. In bundle mode or consolidated boot mode, a .bin image file is used from a local or remote location to boot the device. In the install boot mode, the boot loader uses the packages.conf file to boot up the device.

The following software install features are supported on your switch:

- Software bundle installation on a standalone switch.
- Software rollback to a previously installed package set.
- Emergency installation in the event that no valid installed packages reside on the boot flash.

Software Boot Modes

Your device supports two modes to boot the software packages:

Installed Boot Mode

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

```
Switch: boot flash:packages.conf
```




Note We recommend that you use the install mode for Cisco Catalyst 9200 Series Switches.



Note The packages.conf file for particular release is created on following the install workflow described in the section, *Installing a Software Package*.

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.

Bundle Boot Mode

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

```
switch: boot flash:cat9k_iosxe.16.05.01a.SPA.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.

Changing the Boot Mode

To change a device running in bundle boot mode to install mode, set the boot variable to flash:packages.conf, and execute the **install add file flash:cat9k_2.bin activate commit** command. After the command is executed, the device reboots in install boot mode.

Installing the Software Package

You can install the software package on a device by using the **install add**, **install activate**, and **install commit** commands in privileged EXEC mode.

The **install add** command copies the software package from a local or remote location to the device. The location can be FTP, HTTP, HTTPS, or TFTP. The command extracts individual components of the .bin file into sub-packages and packages.conf file. It also validates the file to ensure that the image file is specific to the platform.

For the **install activate** command to work, the package must be available in the device bootflash. When this command is configured, previously added packages from the .bin file get activated, and the system reloads.

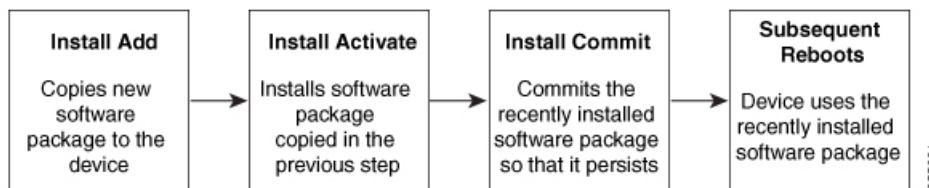
Enable the **install commit** command to make updates persistent over reloads.

Installing an update replaces any previously installed software image. At any time, only one image is installed on the device.

The following flow chart explains how the software install works:

Figure 3: Committing a Software Package

Process with Install Commit



3165291



Note The **install activate** command reloads the device with the new image.

Terminating a Software Install

You can terminate the activation of a software image in the following ways:

- Using the **install activate auto-abort-timer** command. When the device reloads after activating a new image, the auto-abort-timer is triggered. If the timer expires before issuing the **install commit** command, then the installation process is terminated; the device reloads again and boots up with the previous version of the software image.

Use the **install auto-abort-timer stop** command to stop this timer.

- Using the **install abort** command. This command rolls back to the version that was running before installing the new software. Use this command before issuing the **install commit** command.

Devices Information Assignment

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

Use the device 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.

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 device receives the dynamically assigned IP address and reads the configuration file.

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

Default Switch Information

Table 4: 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 Device.
Telnet password	No password is defined.
Cluster command switch functionality	Disabled.
Cluster name	No cluster name 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 device can act as both a DHCP client and a DHCP server.

During DHCP-based autoconfiguration, your device (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 device. However, you need to configure the DHCP server for various lease options associated with IP addresses.

If you want to use DHCP to relay the configuration file location on the network, you might also need to configure a Trivial File Transfer Protocol (TFTP) server and a Domain Name System (DNS) server.



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 device can be on the same LAN or on a different LAN than the device. If the DHCP server is running on a different LAN, you should configure a DHCP relay device between your device 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.

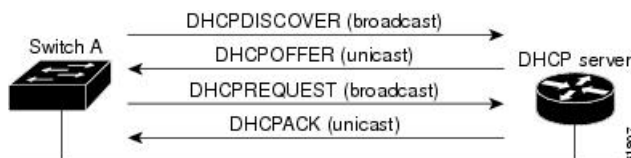
DHCP-based autoconfiguration replaces the BOOTP client functionality on your device.

DHCP Client Request Process

When you boot up your device, the DHCP client is invoked and requests configuration information from a DHCP server when the configuration file is not present on the device. 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 4: DHCP Client and Server Message Exchange



The client, Device 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 device 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 device accepts replies from a BOOTP server and configures itself, the device broadcasts, instead of unicasts, TFTP requests to obtain the device configuration file.

The DHCP hostname option allows a group of devices to obtain hostnames and a standard configuration from the central management DHCP server. A client (device) includes in its DHCPDISCOVER 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.

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 devices in a network. Simultaneous image and configuration

upgrade for all switches in the network helps ensure that each new device added to a network receives the same image and configuration.

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 devices in your network from a DHCP server. The downloaded configuration file becomes the running configuration of the device. It does not overwrite the bootup configuration saved in the flash, until you reload the device.

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 devices in your network. The device (or devices) downloading the new configuration and the new image can be blank (or only have a default factory configuration loaded).

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.)

To enable a DHCP auto-image update on the device, 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 device in your network, the auto-image update feature starts. The downloaded configuration file is saved in the running configuration of the device, and the new image is downloaded and installed on the device. When you reboot the device, the configuration is stored in the saved configuration on the device.

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 device by the device hardware address.

- If you want the device 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 device) (required)
- If you want the device 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 device 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 device is not configured. If the router IP address or the TFTP server name are not found, the device might send broadcast, instead of unicast, TFTP requests. Unavailability of other lease options does not affect autoconfiguration.
- The device can act as a DHCP server. By default, the Cisco IOS DHCP server and relay agent features are enabled on your device but are not configured. (These features are not operational.)

Purpose of the TFTP Server

Based on the DHCP server configuration, the device attempts to download one or more configuration files from the TFTP server. If you configured the DHCP server to respond to the device 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 device 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 device 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 device'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 device 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 device configuration file).
- The `network-config` or the `cisconet.cfg` file (known as the default configuration files).
- The `router-config` or the `ciscottr.cfg` file (These files contain commands common to all devices. 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 device, or if it is to be accessed by the device 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 device.

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 device. If it is on a different LAN, the device 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 device obtains its configuration information in these ways:

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

The device receives its IP address, subnet mask, TFTP server address, and the configuration filename from the DHCP server. The device 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 device, but the TFTP server address is not provided in the DHCP reply (one-file read method).

The device receives its IP address, subnet mask, and the configuration filename from the DHCP server. The device 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 device and provided in the DHCP reply. The configuration filename is not provided (two-file read method).

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

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

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

If the device cannot read the network-config, cisco.net.cfg, or the hostname file, it reads the router-config file. If the device cannot read the router-config file, it reads the ciscotr.cfg file.



Note The device 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 device, you enter the boot loader mode only through the console connection configured for 9600 bps. Unplug the device power cord, and press the **Mode** button while reconnecting the power cord. You can release the **Mode** button after the system LED begins flashing green and remains solid. The boot loader device prompt then appears.

The device 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.

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 5: 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. A reset is required for the new value to take effect.
IP_ADDRESS	Specifies the IP address and the subnet mask for the associated IP subnet of the switch.
DEFAULT_GATEWAY	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 device at a later time (for example, late at night or during the weekend when the device is used less), or you can synchronize a reload network-wide (for example, to perform a software upgrade on all devices 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 effect 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 device is configured for manual booting, do not reload it from a virtual terminal. This restriction prevents the device from entering the boot loader mode and then taking it from the remote user's control.

If you modify your configuration file, the device 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 Device Setup Configuration

Using DHCP to download a new image and a new configuration to a device requires that you configure at least two devices. One device acts as a DHCP and TFTP server and the second device (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 device in the network so that it can support the autoconfiguration of a new device.

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: Device# configure terminal	Enters global configuration mode.
Step 2	ip dhcp pool <i>poolname</i> Example: Device(config)# ip dhcp pool pool	Creates a name for the DHCP server address pool, and enters DHCP pool configuration mode.
Step 3	boot <i>filename</i> Example: Device(dhcp-config)# boot config-boot.text	Specifies the name of the configuration file that is used as a boot image.
Step 4	network <i>network-number mask prefix-length</i> Example: Device(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 <i>address</i> Example: Device(dhcp-config)# default-router 10.10.10.1	Specifies the IP address of the default router for a DHCP client.
Step 6	option 150 <i>address</i> Example:	Specifies the IP address of the TFTP server.

	Command or Action	Purpose
	Device(dhcp-config)# option 150 10.10.10.1	
Step 7	exit Example: Device(dhcp-config)# exit	Returns to global configuration mode.
Step 8	tftp-server flash:filename.text Example: Device(config)# tftp-server flash:config-boot.text	Specifies the configuration file on the TFTP server.
Step 9	interface interface-id Example: Device(config)# interface gigabitethernet1/0/4	Specifies the address of the client that will receive the configuration file.
Step 10	no switchport Example: Device(config-if)# no switchport	Puts the interface into Layer 3 mode.
Step 11	ip address address mask Example: Device(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: Device(config-if)# end	Returns to privileged EXEC mode.

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

This task describes DHCP autoconfiguration to configure TFTP and DHCP settings on an existing device 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 device. In the text file, put the name of the image that you want to download (for example, `cat9k_iosxe.16.xx.xx.SPA.bin`).

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.txt*
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: Device# configure terminal	Enters global configuration mode.
Step 2	ip dhcp pool <i>poolname</i> Example: Device(config)# ip dhcp pool pool1	Creates a name for the DHCP server address pool and enter DHCP pool configuration mode.
Step 3	boot <i>filename</i> Example: Device(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	<p>network <i>network-number mask prefix-length</i></p> <p>Example:</p> <pre>Device(dhcp-config)# network 10.10.10.0 255.255.255.0</pre>	<p>Specifies the subnet network number and mask of the DHCP address pool.</p> <p>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 (/).</p>
Step 5	<p>default-router <i>address</i></p> <p>Example:</p> <pre>Device(dhcp-config)# default-router 10.10.10.1</pre>	Specifies the IP address of the default router for a DHCP client.
Step 6	<p>option 150 <i>address</i></p> <p>Example:</p> <pre>Device(dhcp-config)# option 150 10.10.10.1</pre>	Specifies the IP address of the TFTP server.
Step 7	<p>option 125 <i>hex</i></p> <p>Example:</p> <pre>Device(dhcp-config)# option 125 hex 0000.0009.0a05.08661.7574.6f69.6e73.7461.6c6c.5f64.686370</pre>	Specifies the path to the text file that describes the path to the image file.
Step 8	<p>copy tftp flash <i>filename.txt</i></p> <p>Example:</p> <pre>Device(config)# copy tftp flash image.bin</pre>	Uploads the text file to the device.
Step 9	<p>copy tftp flash <i>imagename.bin</i></p> <p>Example:</p> <pre>Device(config)# copy tftp flash image.bin</pre>	Uploads the tar file for the new image to the device.
Step 10	<p>exit</p> <p>Example:</p> <pre>Device(dhcp-config)# exit</pre>	Returns to global configuration mode.
Step 11	<p>tftp-server flash: <i>config.text</i></p> <p>Example:</p>	Specifies the Cisco IOS configuration file on the TFTP server.

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

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: Device# configure terminal	Enters global configuration mode.
Step 2	boot host dhcp Example: Device(conf)# boot host dhcp	Enables autoconfiguration with a saved configuration.
Step 3	boot host retry timeout <i>timeout-value</i> Example: Device(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 <i>warning-message</i> ^C Example: Device(conf)# banner config-save ^C Caution - Saving Configuration File to NVRAM May Cause You to No longer Automatically Download Configuration Files at Reboot^C	(Optional) Creates warning messages to be displayed when you try to save the configuration file to NVRAM.
Step 5	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config-if)# end	
Step 6	show boot Example: Device# show boot	Verifies the configuration.

Manually Assigning IP Information to Multiple SVIs

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

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: Device# configure terminal	Enters global configuration mode.
Step 2	interface vlan <i>vlan-id</i> Example: Device(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 <i>ip-address subnet-mask</i> Example: Device(config-vlan)# ip address 10.10.10.2 255.255.255.0	Enters the IP address and subnet mask.

	Command or Action	Purpose
Step 4	exit Example: Device(config-vlan)# exit	Returns to global configuration mode.
Step 5	ip default-gateway ip-address Example: Device(config)# ip default-gateway 10.10.10.1	<p>Enters the IP address of the next-hop router interface that is directly connected to the device where a default gateway is being configured. The default gateway receives IP packets with unresolved destination IP addresses from the device.</p> <p>Once the default gateway is configured, the device has connectivity to the remote networks with which a host needs to communicate.</p> <p>Note When your device is configured to route with IP, it does not need to have a default gateway set.</p> <p>Note The device capwap relays on default-gateway configuration to support routed access point join the device.</p>
Step 6	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 7	show interfaces vlan vlan-id Example: Device# show interfaces vlan 99	Verifies the configured IP address.
Step 8	show ip redirects Example: Device# show ip redirects	Verifies the configured default gateway.

Modifying the Device 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 device 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: <i>file-url</i> Example: Switch(config) # boot flash:config.text	Specifies the configuration file to load during the next boot cycle. <i>file-url</i> —The path (directory) and the configuration filename. Filenames and directory names are case-sensitive.
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: Device# configure terminal	Enters global configuration mode.
Step 2	boot manual Example: Device(config)# boot manual	Enables the switch to manually boot up during the next boot cycle.
Step 3	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 4	show boot Example: Device# 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 <i>switch:</i> prompt. To boot up the system, use the boot filesystem:/file-url boot loader command. <ul style="list-style-type: none"> • <i>filesystem:</i>—Uses flash: for the system board flash device. Switch: boot flash: • For <i>file-url</i>—Specifies the path (directory) and the name of the bootable image. Filenames and directory names are case-sensitive.

	Command or Action	Purpose
Step 5	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Booting the Device in Installed Mode

Installing a Software Package

You can install, activate, and commit a software package using a single command or using separate commands. This task shows how to use the **install add file activate commit** command for installing a software package.

SUMMARY STEPS

1. **enable**
2. **install add file tftp: *filename* [activate commit]**
3. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	install add file tftp: <i>filename</i> [activate commit] Example: <pre>Device# install add file tftp://172.16.0.1//tftpboot/folder1/ cat9k_iosxe.16.06.01.SPA.bin activate commit</pre>	Copies the software install package from a remote location (via FTP, HTTP, HTTPS, TFTP) to the device, performs a compatibility check for the platform and image versions, activates the software package, and makes the package persistent across reloads. <ul style="list-style-type: none"> • This command extracts the individual components of the .bin file into sub-packages and packages.conf file. • The device reloads after executing this command.
Step 3	exit Example: <pre>Device# exit</pre>	Exits privileged EXEC mode and returns to user EXEC mode.

Managing the Update Package

SUMMARY STEPS

1. **enable**
2. **install add file tftp: *filename***

3. **install activate** [auto-abort-timer]
4. **install abort**
5. **install commit**
6. **install rollback to committed**
7. **install remove** {file filesystem: filename | inactive}
8. **show install summary**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	install add file tftp: filename Example: Device# install add file tftp://172.16.0.1/tftpboot/folder1/ cat9k_iosxe.16.06.01.SPA.bin	Copies the software install package from a remote location (via FTP, HTTP, HTTPS, TFTP) to the device, and performs a compatibility check for the platform and image versions. <ul style="list-style-type: none"> • This command extracts the individual components of the .bin file into sub-packages and packages.conf file.
Step 3	install activate [auto-abort-timer] Example: Device# install activate	Activates the added software install package, and reloads the device. <ul style="list-style-type: none"> • When doing a full software install, do not provide a package filename. • The auto-abort-timer keyword, automatically rolls back the software image activation. The automatic timer is triggered after the new image is activated. If the timer expires prior to the issuing of the install commit command, then the install process is automatically terminated. The device reloads, and boots up with a previous version of the software image.
Step 4	install abort Example: Device# install abort	(Optional) Terminates the software install activation, and rolls back to the version that was running before current installation procedure. <ul style="list-style-type: none"> • You can use this command only when the image is in an activated state; and not when the image is in a committed state.
Step 5	install commit Example: Device# install commit	Makes the changes persistent over reload. <ul style="list-style-type: none"> • The install commit command completes the new image installation. Changes are persistent across reloads until the auto-abort timer expires.

	Command or Action	Purpose
Step 6	install rollback to committed Example: Device# install rollback to committed	(Optional) Rolls back the update to the last committed version.
Step 7	install remove {file filesystem: filename inactive} Example: Device# install remove inactive	(Optional) Deletes all unused and inactive software installation files.
Step 8	show install summary Example: Device# show install summary	Displays information about the active package. <ul style="list-style-type: none"> The output of this command varies according to the install commands that are configured.

Booting the Device in Bundle Mode

There are several methods by which you can boot the device—either by copying the bin file from the TFTP server and then boot the device, or by booting the device 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 device from the TFTP server in the bundle mode.

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch:BOOT=<source path of .bin file> Example: switch:BOOT=tftp://10.0.0.2/cat9k_iosxe.16.05.01a.SPA.bin	Sets the boot parameters.
Step 2	boot Example: switch: boot	Boots the device.
Step 3	show version	Verifies that the device is in the BUNDLE mode.

Configuring a Scheduled Software Image Reload

This task describes how to configure your device 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 at** *hh: mm* *[month day | day month]* *[text]*
5. **reload cancel**
6. **show reload**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	copy running-config startup-config Example: Device# copy running-config startup-config	Saves your device configuration information to the startup configuration before you use the reload command.
Step 3	reload in <i>[hh:]mm</i> <i>[text]</i> Example: Device# 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 at <i>hh: mm</i> <i>[month day day month]</i> <i>[text]</i> Example: Device(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 device 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 device. To schedule reloads across several devices to occur simultaneously, the time on each device must be synchronized with NTP.
Step 5	reload cancel Example: Device(config)# reload cancel	Cancels a previously scheduled reload.
Step 6	show reload Example: show reload	Displays information about a previously scheduled reload or identifies if a reload has been scheduled on the device.

Monitoring Device Setup Configuration

Examples: Displaying Software Bootup in Install Mode

This example displays software bootup in install mode:

```
switch: boot flash:packages.conf
Attempting to boot from [flash:packages.conf]
Located packages.conf
#

validate_package: SHA-1 hash:
    expected 340D5091:2872A0DD:03E9068C:3FDBECAB:69786462
    calculated 340D5091:2872A0DD:03E9068C:3FDBECAB:69786462
Image parsed from conf file is cat9k-rpboot.16.05.01a.SPA.pkg
#####

Waiting for 120 seconds for other switches to boot
#####
Switch number is 1

                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 [Everest], Catalyst L3 Switch Software (CAT9K_IOSXE), Version 16.5.1a,
RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2017 by Cisco Systems, Inc.
Compiled Tue 30-May-17 00:36 by mcpre

Cisco IOS-XE software, Copyright (c) 2005-2017 by cisco Systems, Inc.
All rights reserved. Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0. The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY. You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0. For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.

FIPS: Flash Key Check : Begin
FIPS: Flash Key Check : End, Not Found, FIPS Mode Not Enabled
```


This product contains cryptographic features and is subject to United States and local country laws governing import, export, transfer and use. Delivery of Cisco cryptographic products does not imply third-party authority to import, export, distribute or use encryption. Importers, exporters, distributors and users are responsible for compliance with U.S. and local country laws. By using this product you agree to comply with applicable laws and regulations. If you are unable to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at: <http://www.cisco.com/wwl/export/crypto/tool/stqrg.html>

If you require further assistance please contact us by sending email to export@cisco.com.

```
cisco C9300-48P (X86) processor with 818597K/6147K bytes of memory.
Processor board ID FCW2049G03S
2048K bytes of non-volatile configuration memory.
8388608K bytes of physical memory.
1638400K bytes of Crash Files at crashinfo:.
11264000K bytes of Flash at flash:.
0K bytes of WebUI ODM Files at webui:.
```

```
Base Ethernet MAC Address       : 04:6c:9d:01:3b:80
Motherboard Assembly Number    : 73-17956-04
Motherboard Serial Number     : FOC20465ABU
Model Revision Number         : P4B
Motherboard Revision Number   : 04
Model Number                  : C9300-48P
System Serial Number          : FCW2049G03S
```

```
%INIT: waited 0 seconds for NVRAM to be available
```

```
Defaulting CPP : Policer rate for all classes will be set to their defaults
```

```
Press RETURN to get started!
```

This example displays software bootup in bundle mode:

```
switch: boot flash:cat9k_iosxe.16.05.01a.SPA.bin
```

```
Attempting to boot from [flash:cat9k_iosxe.16.05.01a.SPA.bin]
Located cat9k_iosxe.16.05.01a.SPA.bin
```

```
#####
Warning: ignoring ROMMON var "BOOT_PARAM"
```

```
Waiting for 120 seconds for other switches to boot
```

```
#####
Switch number is 3
```

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 [Everest], Catalyst L3 Switch Software (CAT9K_IOSXE), Version 16.5.1a,
RELEASE SOFTWARE (fc2)
Technical Support: <http://www.cisco.com/techsupport>
Copyright (c) 1986-2017 by Cisco Systems, Inc.
Compiled Tue 30-May-17 00:36 by mcpre

Cisco IOS-XE software, Copyright (c) 2005-2017 by cisco Systems, Inc.
All rights reserved. Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0. The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY. You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0. For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.

FIPS: Flash Key Check : Begin
FIPS: Flash Key Check : End, Not Found, FIPS Mode Not Enabled

This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
<http://www.cisco.com/wvl/export/crypto/tool/stqrg.html>

If you require further assistance please contact us by sending email to
export@cisco.com.

cisco C9300-24U (X86) processor with 818597K/6147K bytes of memory.
Processor board ID FCW2111G00X
2048K bytes of non-volatile configuration memory.
8388608K bytes of physical memory.
1638400K bytes of Crash Files at crashinfo:.
11264000K bytes of Flash at flash:.
15633392K bytes of USB Flash at usbflash0:.
0K bytes of WebUI ODM Files at webui:.

Base Ethernet MAC Address	: 04:6c:9d:1e:2a:80
Motherboard Assembly Number	: 73-17954-05
Motherboard Serial Number	: FOC21094MWL
Model Revision Number	: PP
Motherboard Revision Number	: 05
Model Number	: C9300-24U
System Serial Number	: FCW2111G00X

%INIT: waited 0 seconds for NVRAM to be available

Defaulting CPP : Policer rate for all classes will be set to their defaults

Press RETURN to get started!

Example: Emergency Installation

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

```
switch: emergency-install
tftp://223.255.254.254/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin
WARNING: The system partition (bootflash:) will be erased during the system recovery install
process.
Are you sure you want to proceed? [y] y/n [n]: y
Starting system recovery
(tftp://223.255.254.254/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin) ...
Attempting to boot from [sda9:cat9k-recovery.SSA.bin]
Located cat9k-recovery.SSA.bin
#####

Warning: ignoring ROMMON var "BOOT_PARAM"

PLATFORM_TYPE C9300 speed 9600

Booting Recovery Image 16.5.1a

Initiating Emergency Installation of bundle
tftp://223.255.254.254/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin

Downloading bundle tftp://223.255.254.254/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin...
curl_vrf=2
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
 100 485M  100 485M    0     0 5143k      0  0:01:36  0:01:36 ---:---:-- 5256k
 100 485M  100 485M    0     0 5143k      0  0:01:36  0:01:36 ---:---:-- 5143k

Validating bundle tftp://223.255.254.254/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin...
Installing bundle tftp://223.255.254.254/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin...
Verifying bundle tftp://223.255.254.254/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin...
Package cat9k-cc_srdriver.16.05.01a.SPA.pkg /temp//stage/cat9k-cc_srdriver.16.05.01a.SPA.pkg
is Digitally Signed
Package cat9k-espbase.16.05.01a.SPA.pkg /temp//stage/cat9k-espbase.16.05.01a.SPA.pkg is
Digitally Signed
Package cat9k-guestshell.16.05.01a.SPA.pkg /temp//stage/cat9k-guestshell.16.05.01a.SPA.pkg
is Digitally Signed
Package cat9k-rpbase.16.05.01a.SPA.pkg /temp//stage/cat9k-rpbase.16.05.01a.SPA.pkg is
Digitally Signed
Package cat9k-sipbase.16.05.01a.SPA.pkg /temp//stage/cat9k-sipbase.16.05.01a.SPA.pkg is
Digitally Signed
Package cat9k-sipspace.16.05.01a.SPA.pkg /temp//stage/cat9k-sipspace.16.05.01a.SPA.pkg is
Digitally Signed
Package cat9k-srdriver.16.05.01a.SPA.pkg /temp//stage/cat9k-srdriver.16.05.01a.SPA.pkg is
Digitally Signed
Package cat9k-webui.16.05.01a.SPA.pkg /temp//stage/cat9k-webui.16.05.01a.SPA.pkg is Digitally
Signed
Package cat9k-wlc.16.05.01a.SPA.pkg /temp//stage/cat9k-wlc.16.05.01a.SPA.pkg is Digitally
```

```

Signed
Package /cat9k-rpboot.16.05.01a.SPA.pkg /temp//rpboot/cat9k-rpboot.16.05.01a.SPA.pkg is
Digitally Signed
Preparing flash....
Flash filesystem unmounted successfully /dev/sdb3
Syncing device....
Emergency Install successful... Rebooting
Will reboot now

Initializing Hardware...

System Bootstrap, Version 16.5.2r, RELEASE SOFTWARE (P)
Compiled Wed 05/31/2017 15:58:35.22 by rel

Current image running:
Primary Rommon Image

Last reset cause: SoftwareReload
C9300-24U platform with 8388608 Kbytes of main memory

```

Configuration Examples for Performing Device Setup

Example: Managing an Update Package

The following example shows how to add a software package file:

```

Device# install add file flash:cat9k_iosxe.16.06.02.SPA.bin activate commit

install_add_activate_commit: START Mon Oct 30 19:54:51 UTC 2017

System configuration has been modified.
Press Yes(y) to save the configuration and proceed.
Press No(n) for proceeding without saving the configuration.
Press Quit(q) to exit, you may save configuration and re-enter the command. [y/n/q]y
Building configuration...

[OK]Modified configuration has been saved

*Oct 30 19:54:55.633: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 19:54:55 install_engine.sh:

%INSTALL-5-INSTALL_START_INFO: Started install one-shot
flash:cat9k_iosxe.16.06.02.SPA.bininstall_add_activate_commit: Adding PACKAGE

This operation requires a reload of the system. Do you want to proceed?
Please confirm you have changed boot config to flash:packages.conf [y/n]y

--- Starting initial file syncing ---
Info: Finished copying flash:cat9k_iosxe.16.06.02.SPA.bin to the selected switch(es)
Finished initial file syncing

--- Starting Add ---
Performing Add on all members
  [1] Add package(s) on switch 1
  [1] Finished Add on switch 1
Checking status of Add on [1]
Add: Passed on [1]
Finished Add

install_add_activate_commit: Activating PACKAGE

```

```

Following packages shall be activated:
/flash/cat9k-wlc.16.06.02.SPA.pkg
/flash/cat9k-webui.16.06.02.SPA.pkg
/flash/cat9k-srdriver.16.06.02.SPA.pkg
/flash/cat9k-sipspa.16.06.02.SPA.pkg
/flash/cat9k-sipbase.16.06.02.SPA.pkg
/flash/cat9k-rpboot.16.06.02.SPA.pkg
/flash/cat9k-rpbase.16.06.02.SPA.pkg
/flash/cat9k-guestshell.16.06.02.SPA.pkg
/flash/cat9k-espbase.16.06.02.SPA.pkg
/flash/cat9k-cc_srdriver.16.06.02.SPA.pkg

This operation requires a reload of the system. Do you want to proceed? [y/n]y
--- Starting Activate ---
Performing Activate on all members
  [1] Activate package(s) on switch 1
  [1] Finished Activate on switch 1
Checking status of Activate on [1]
Activate: Passed on [1]
Finished Activate

--- Starting Commit ---
Performing Commit on all members

*Oct 30 19:57:41.145: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 19:57:41 rollback_timer.sh:

%INSTALL-5-INSTALL_AUTO_ABORT_TIMER_PROGRESS: Install auto abort timer will expire in 7200
seconds [1]
Commit package(s) on switch 1
  [1] Finished Commit on switch 1
Checking status of Commit on [1]
Commit: Passed on [1]
Finished Commit

Install will reload the system now!
SUCCESS: install_add_activate_commit Mon Oct 30 19:57:48 UTC 2017

Device#
*Oct 30 19:57:48.384: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 19:57:48 install_engine.sh:

%INSTALL-5-INSTALL_COMPLETED_INFO: Completed install one-shot PACKAGE
flash:cat9k_iosxe.16.06.02.SPA.bin

Chassis 1 reloading, reason - Reload command

```

The following is sample output from the **show install summary** command after adding a software package file to the device:

```

Device# show install summary

[ R0 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
             C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type  St   Filename/Version
-----
IMG   I    16.6.1.0
IMG   C    16.6.2.0

```

The following example shows how to activate an added software package file:

Example: Managing an Update Package

```

Device# install activate

install_activate: START Mon Oct 30 20:14:20 UTC 2017
install_activate: Activating PACKAGE

*Oct 30 20:14:21.379: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 20:14:21 install_engine.sh:

%INSTALL-5-INSTALL_START_INFO: Started install activateFollowing packages shall be activated:
/flash/cat9k-wlc.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
/flash/cat9k-webui.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
/flash/cat9k-srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
/flash/cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
/flash/cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
/flash/cat9k-rpboot.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
/flash/cat9k-rpbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
/flash/cat9k-guestshell.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
/flash/cat9k-espbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
/flash/cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg

This operation requires a reload of the system. Do you want to proceed? [y/n]y
--- Starting Activate ---
Performing Activate on all members
[1] Activate package(s) on switch 1
--- Starting list of software package changes ---
Old files list:
  Removed cat9k-cc_srdriver.16.06.02.SPA.pkg
  Removed cat9k-espbase.16.06.02.SPA.pkg
  Removed cat9k-guestshell.16.06.02.SPA.pkg
  Removed cat9k-rpbase.16.06.02.SPA.pkg
  Removed cat9k-rpboot.16.06.02.SPA.pkg
  Removed cat9k-sipbase.16.06.02.SPA.pkg
  Removed cat9k-sipspa.16.06.02.SPA.pkg
  Removed cat9k-srdriver.16.06.02.SPA.pkg
  Removed cat9k-webui.16.06.02.SPA.pkg
  Removed cat9k-wlc.16.06.02.SPA.pkg
New files list:
  Added cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Added cat9k-espbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Added cat9k-guestshell.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Added cat9k-rpbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Added cat9k-rpboot.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Added cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Added cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Added cat9k-srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Added cat9k-webui.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Added cat9k-wlc.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Finished list of software package changes
[1] Finished Activate on switch 1
Checking status of Activate on [1]
Activate: Passed on [1]
Finished Activate

*Oct 30 20:15:56.572: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 20:15:56 rollback_timer.sh:
%INSTALL-5-INSTALL_AUTO_ABORT_TIMER_PROGRESS: Install auto abort timer will expire in 7200
seconds
Install will reload the system now!
SUCCESS: install_activate Mon Oct 30 20:16:01 UTC 2017

Device#
*Oct 30 20:16:01.935: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 20:16:01
install_engine.sh: %INSTALL-5-INSTALL_COMPLETED_INFO: Completed install activate PACKAGE
Chassis 1 reloading, reason - Reload command

```

The following sample output from the **show install summary** command displays the status of the software package as active and uncommitted:

```
Device# show install summary

[ R0 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
             C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type  St  Filename/Version
-----
IMG   I   16.6.2.0
IMG   U   16.6.1.0
Device#
```

The following example shows how to execute the **install commit** command:

```
Device# install commit
install_commit: START Fri Jun 23 21:24:45 IST 2017
install_commit: Committing PACKAGE

--- Starting Commit ---
Performing Commit on Active/Standby
  [R0] Commit package(s) on R0
  [R0] Finished Commit on R0
Checking status of Commit on [R0]
Commit: Passed on [R0]
Finished Commit

SUCCESS: install_commit  Fri Jun 23 21:24:48 IST 2017

Device#
```

The following example shows how to rollback an update package to the base package:

```
Device# install rollback to committed

install_rollback: START Mon Oct 30 20:53:33 UTC 2017

This operation requires a reload of the system. Do you want to proceed? [y/n]

*Oct 30 20:53:34.713: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 20:53:34
install_engine.sh: %INSTALL-5-INSTALL_START_INFO: Started install rollback

--- Starting Rollback ---
Performing Rollback on all members
[1] Rollback package(s) on switch 1
--- Starting rollback impact ---
Changes that are part of this rollback
Current   : rp 0 0  rp_boot      cat9k-rpboot.16.06.02.prd9.SPA.pkg
Current   : rp 1 0  rp_boot      cat9k-rpboot.16.06.02.prd9.SPA.pkg
Replacement: rp 0 0  rp_boot      cat9k-rpboot.16.06.02.SPA.pkg
Replacement: rp 1 0  rp_boot      cat9k-rpboot.16.06.02.SPA.pkg
Current   : cc 0 0  cc_srdriver   cat9k-cc_srdriver.16.06.02.prd9.SPA.pkg
Current   : cc 0 0  cc            cat9k-sipbase.16.06.02.prd9.SPA.pkg
Current   : cc 0 0  cc_spa        cat9k-sipspa.16.06.02.prd9.SPA.pkg
Current   : cc 1 0  cc_srdriver   cat9k-cc_srdriver.16.06.02.prd9.SPA.pkg
Current   : cc 1 0  cc            cat9k-sipbase.16.06.02.prd9.SPA.pkg
Current   : cc 1 0  cc_spa        cat9k-sipspa.16.06.02.prd9.SPA.pkg
Current   : cc 10 0 cc            cat9k-sipbase.16.06.02.prd9.SPA.pkg
Current   : cc 10 0 cc_spa        cat9k-sipspa.16.06.02.prd9.SPA.pkg
Current   : cc 10 0 cc_srdriver   cat9k-cc_srdriver.16.06.02.prd9.SPA.pkg
Current   : cc 2 0  cc_srdriver   cat9k-cc_srdriver.16.06.02.prd9.SPA.pkg
```

Example: Managing an Update Package

```

Current      : cc 2 0  cc          cat9k-sipbase.16.06.02.pr99.SPA.pkg
Current      : cc 2 0  cc_spa      cat9k-sipspace.16.06.02.pr99.SPA.pkg
Current      : cc 3 0  cc_srdriver  cat9k-cc_srdriver.16.06.02.pr99.SPA.pkg
Current      : cc 3 0  cc          cat9k-sipbase.16.06.02.pr99.SPA.pkg
Current      : cc 3 0  cc_spa      cat9k-sipspace.16.06.02.pr99.SPA.pkg
Current      : cc 4 0  cc_srdriver  cat9k-cc_srdriver.16.06.02.pr99.SPA.pkg
Current      : cc 4 0  cc          cat9k-sipbase.16.06.02.pr99.SPA.pkg
Current      : cc 4 0  cc_spa      cat9k-sipspace.16.06.02.pr99.SPA.pkg
Current      : cc 5 0  cc_srdriver  cat9k-cc_srdriver.16.06.02.pr99.SPA.pkg
Current      : cc 5 0  cc          cat9k-sipbase.16.06.02.pr99.SPA.pkg
Current      : cc 5 0  cc_spa      cat9k-sipspace.16.06.02.pr99.SPA.pkg
Current      : cc 6 0  cc_srdriver  cat9k-cc_srdriver.16.06.02.pr99.SPA.pkg
Current      : cc 6 0  cc          cat9k-sipbase.16.06.02.pr99.SPA.pkg
Current      : cc 6 0  cc_spa      cat9k-sipspace.16.06.02.pr99.SPA.pkg
Current      : cc 7 0  cc_srdriver  cat9k-cc_srdriver.16.06.02.pr99.SPA.pkg
Current      : cc 7 0  cc          cat9k-sipbase.16.06.02.pr99.SPA.pkg
Current      : cc 7 0  cc_spa      cat9k-sipspace.16.06.02.pr99.SPA.pkg
Current      : cc 8 0  cc_srdriver  cat9k-cc_srdriver.16.06.02.pr99.SPA.pkg
Current      : cc 8 0  cc          cat9k-sipbase.16.06.02.pr99.SPA.pkg
Current      : cc 8 0  cc_spa      cat9k-sipspace.16.06.02.pr99.SPA.pkg
Current      : cc 9 0  cc_srdriver  cat9k-cc_srdriver.16.06.02.pr99.SPA.pkg
Current      : cc 9 0  cc          cat9k-sipbase.16.06.02.pr99.SPA.pkg
Current      : cc 9 0  cc_spa      cat9k-sipspace.16.06.02.pr99.SPA.pkg
Current      : fp 0 0  fp          cat9k-espbase.16.06.02.pr99.SPA.pkg
Current      : fp 1 0  fp          cat9k-espbase.16.06.02.pr99.SPA.pkg
Current      : rp 0 0  guestshell  cat9k-guestshell.16.06.02.pr99.SPA.pkg
Current      : rp 0 0  rp_base      cat9k-rpbase.16.06.02.pr99.SPA.pkg
Current      : rp 0 0  rp_daemons  cat9k-rpbase.16.06.02.pr99.SPA.pkg
Current      : rp 0 0  rp_iosd      cat9k-rpbase.16.06.02.pr99.SPA.pkg
Current      : rp 0 0  rp_security  cat9k-rpbase.16.06.02.pr99.SPA.pkg
Current      : rp 0 0  rp_webui     cat9k-webui.16.06.02.pr99.SPA.pkg
Current      : rp 0 0  rp_wlc       cat9k-wlc.16.06.02.pr99.SPA.pkg
Current      : rp 0 0  srdriver     cat9k-srdriver.16.06.02.pr99.SPA.pkg
Current      : rp 1 0  guestshell  cat9k-guestshell.16.06.02.pr99.SPA.pkg
Current      : rp 1 0  rp_base      cat9k-rpbase.16.06.02.pr99.SPA.pkg
Current      : rp 1 0  rp_daemons  cat9k-rpbase.16.06.02.pr99.SPA.pkg
Current      : rp 1 0  rp_iosd      cat9k-rpbase.16.06.02.pr99.SPA.pkg
Current      : rp 1 0  rp_security  cat9k-rpbase.16.06.02.pr99.SPA.pkg
Current      : rp 1 0  rp_webui     cat9k-webui.16.06.02.pr99.SPA.pkg
Current      : rp 1 0  rp_wlc       cat9k-wlc.16.06.02.pr99.SPA.pkg
Current      : rp 1 0  srdriver     cat9k-srdriver.16.06.02.pr99.SPA.pkg
Replacement: cc 0 0  cc_srdriver  cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 0 0  cc          cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 0 0  cc_spa      cat9k-sipspace.16.06.02.SPA.pkg
Replacement: cc 1 0  cc_srdriver  cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 1 0  cc          cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 1 0  cc_spa      cat9k-sipspace.16.06.02.SPA.pkg
Replacement: cc 10 0  cc          cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 10 0  cc_spa      cat9k-sipspace.16.06.02.SPA.pkg
Replacement: cc 10 0  cc_srdriver  cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 2 0  cc_srdriver  cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 2 0  cc          cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 2 0  cc_spa      cat9k-sipspace.16.06.02.SPA.pkg
Replacement: cc 3 0  cc_srdriver  cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 3 0  cc          cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 3 0  cc_spa      cat9k-sipspace.16.06.02.SPA.pkg
Replacement: cc 4 0  cc_srdriver  cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 4 0  cc          cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 4 0  cc_spa      cat9k-sipspace.16.06.02.SPA.pkg
Replacement: cc 5 0  cc_srdriver  cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 5 0  cc          cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 5 0  cc_spa      cat9k-sipspace.16.06.02.SPA.pkg
Replacement: cc 6 0  cc_srdriver  cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 6 0  cc          cat9k-sipbase.16.06.02.SPA.pkg

```



```

Replacement: cc 6 0 cc_spa cat9k-sipspace.16.06.02.SPA.pkg
Replacement: cc 7 0 cc_srdriver cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 7 0 cc cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 7 0 cc_spa cat9k-sipspace.16.06.02.SPA.pkg
Replacement: cc 8 0 cc_srdriver cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 8 0 cc cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 8 0 cc_spa cat9k-sipspace.16.06.02.SPA.pkg
Replacement: cc 9 0 cc_srdriver cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 9 0 cc cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 9 0 cc_spa cat9k-sipspace.16.06.02.SPA.pkg
Replacement: fp 0 0 fp cat9k-espbase.16.06.02.SPA.pkg
Replacement: fp 1 0 fp cat9k-espbase.16.06.02.SPA.pkg
Replacement: rp 0 0 guestshell cat9k-guestshell.16.06.02.SPA.pkg
Replacement: rp 0 0 rp_base cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 0 0 rp_daemons cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 0 0 rp_iosd cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 0 0 rp_security cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 0 0 rp_webui cat9k-webui.16.06.02.SPA.pkg
Replacement: rp 0 0 rp_wlc cat9k-wlc.16.06.02.SPA.pkg
Replacement: rp 0 0 srdriver cat9k-srdriver.16.06.02.SPA.pkg
Replacement: rp 1 0 guestshell cat9k-guestshell.16.06.02.SPA.pkg
Replacement: rp 1 0 rp_base cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 1 0 rp_daemons cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 1 0 rp_iosd cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 1 0 rp_security cat9k-rpbase.16.06.02.SPA.pkg

```

Chassis 1 reloading, reason - Reload command

```

Replacement: rp 1 0 rp_webui cat9k-webui.16.06.02.SPA.pkg
Replacement: rp 1 0 rp_wlc cat9k-wlc.16.06.02.SPA.pkg
Replacement: rp 1 0 srdriver cat9k-srdriver.16.06.02.SPA.pkg
Finished rollback impact

```

```

[1] Finished Rollback on switch 1
Checking status of Rollback on [1]
Rollback: Passed on [1]
Finished Rollback

```

```

Install will reload the system now!
SUCCESS: install_rollback Mon Oct 30 20:54:23 UTC 2017

```

```

Device#
*Oct 30 20:54:23.576: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 20:54:23
install_engine.sh: %INSTALL-5-INSTALL_COMPLETED_INFO: Completed install rollback PACKAGE
*Oct 30 20:54:25.416: %STACKMGR-1-RELOAD: Switch 1 R0/0: stack_mgr:
Reloading due to reason Reload command Oct 30 20:54:31.615 FP0/0: %PMAN-5-EXITACTION:
Process manager is exiting: reload fp action requested
Oct 30 20:54

```

The following is sample output from the **install remove inactive** command:

```

Device# install remove inactive

install_remove: START Mon Oct 30 19:51:48 UTC 2017
Cleaning up unnecessary package files
  Scanning boot directory for packages ... done.
  Preparing packages list to delete ...
done.

```

```

The following files will be deleted:
[switch 1]:
/flash/cat9k-cc_srdriver.16.06.02.SPA.pkg
/flash/cat9k-espbase.16.06.02.SPA.pkg
/flash/cat9k-guestshell.16.06.02.SPA.pkg
/flash/cat9k-rpbase.16.06.02.SPA.pkg

```

Example: Managing an Update Package

```

/flash/cat9k-rpboot.16.06.02.SPA.pkg
/flash/cat9k-sipbase.16.06.02.SPA.pkg
/flash/cat9k-sipspsa.16.06.02.SPA.pkg
/flash/cat9k-srdriver.16.06.02.SPA.pkg
/flash/cat9k-webui.16.06.02.SPA.pkg
/flash/cat9k-wlc.16.06.02.SPA.pkg
/flash/packages.conf

Do you want to remove the above files? [y/n]y
[switch 1]:
Deleting file flash:cat9k-cc_srdriver.16.06.02.SPA.pkg ... done.
Deleting file flash:cat9k-espsbase.16.06.02.SPA.pkg ... done.
Deleting file flash:cat9k-guestshell.16.06.02.SPA.pkg ... done.
Deleting file flash:cat9k-rpbase.16.06.02.SPA.pkg ... done.
Deleting file flash:cat9k-rpboot.16.06.02.SPA.pkg ... done.
Deleting file flash:cat9k-sipbase.16.06.02.SPA.pkg ... done.
Deleting file flash:cat9k-sipspsa.16.06.02.SPA.pkg ... done.
Deleting file flash:cat9k-srdriver.16.06.02.SPA.pkg ... done.
Deleting file flash:cat9k-webui.16.06.02.SPA.pkg ... done.
Deleting file flash:cat9k-wlc.16.06.02.SPA.pkg ... done.
Deleting file flash:packages.conf ... done.
SUCCESS: Files deleted.
--- Starting Post_Remove_Cleanup ---
Performing Post_Remove_Cleanup on all members
  [1] Post_Remove_Cleanup package(s) on switch 1
  [1] Finished Post_Remove_Cleanup on switch 1
Checking status of Post_Remove_Cleanup on [1]
Post_Remove_Cleanup: Passed on [1]
Finished Post_Remove_Cleanup

SUCCESS: install_remove  Mon Oct 30 19:52:25 UTC 2017
Device#

```

The following is sample output from the **install abort** command:

```

Device# install abort

/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
install_abort: START Mon Oct 30 20:27:32 UTC 2017
install_abort: Abort type PACKAGE subtype NONE smutype NONE

This install abort would require a reload. Do you want to proceed? [y/n]

*Oct 30 20:27:33.189: %INSTALL-5-INSTALL_START_INFO: Switch 1 R0/0: install_engine: Started
install abort
--- Starting Abort ---
Performing Abort on all members
/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
[1] Abort package(s) on switch 1
  --- Starting rollback impact ---
  Changes that are part of this rollback
  Current      : rp 0 0  rp_boot
cat9k-rpboot.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current      : rp 1 0  rp_boot
cat9k-rpboot.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Replacement: rp 0 0  rp_boot          cat9k-rpboot.16.06.02.SPA.pkg

```

```

Replacement: rp 1 0 rp_boot cat9k-rpboot.16.06.02.SPA.pkg
Current : cc 0 0 cc_srdriver
cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 0 0 cc
cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 0 0 cc_spa
cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 1 0 cc_srdriver
cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 1 0 cc
cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 1 0 cc_spa
cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 10 0 cc
cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 10 0 cc_spa
cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 10 0 cc_srdriver
cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 2 0 cc_srdriver
cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 2 0 cc
cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 2 0 cc_spa
cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 3 0 cc_srdriver
cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 3 0 cc
cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 3 0 cc_spa
cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 4 0 cc_srdriver
cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 4 0 cc
cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 4 0 cc_spa
cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 5 0 cc_srdriver
cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 5 0 cc
cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 5 0 cc_spa
cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 6 0 cc_srdriver
cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 6 0 cc
cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 6 0 cc_spa
cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 7 0 cc_srdriver
cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 7 0 cc
cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 7 0 cc_spa
cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 8 0 cc_srdriver
cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 8 0 cc
cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 8 0 cc_spa
cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 9 0 cc_srdriver
cat9k-cc_srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
Current : cc 9 0 cc

```

Example: Managing an Update Package

```

cat9k-sipbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current : cc 9 0 cc_spa
cat9k-sipspa.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current : fp 0 0 fp
cat9k-esppbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current : fp 1 0 fp
cat9k-esppbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current : rp 0 0 guestshell
cat9k-guestshell.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current : rp 0 0 rp_base
cat9k-rpbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current : rp 0 0 rp_daemons
cat9k-rpbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current : rp 0 0 rp_iosd
cat9k-rpbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current : rp 0 0 rp_security
cat9k-rpbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current : rp 0 0 rp_webui
cat9k-webui.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current : rp 0 0 rp_wlc
cat9k-wlc.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current : rp 0 0 srdriver
cat9k-srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current : rp 1 0 guestshell
cat9k-guestshell.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current : rp 1 0 rp_base
cat9k-rpbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current : rp 1 0 rp_daemons
cat9k-rpbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current : rp 1 0 rp_iosd
cat9k-rpbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current : rp 1 0 rp_security
cat9k-rpbase.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current : rp 1 0 rp_webui
cat9k-webui.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current : rp 1 0 rp_wlc
cat9k-wlc.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Current : rp 1 0 srdriver
cat9k-srdriver.BLD_POLARIS_DEV_LATEST_20171029_082249.SSA.pkg
  Replacement: cc 0 0 cc_srdriver cat9k-cc_srdriver.16.06.02.SPA.pkg
  Replacement: cc 0 0 cc cat9k-sipbase.16.06.02.SPA.pkg
  Replacement: cc 0 0 cc_spa cat9k-sipspa.16.06.02.SPA.pkg
  Replacement: cc 1 0 cc_srdriver cat9k-cc_srdriver.16.06.02.SPA.pkg
  Replacement: cc 1 0 cc cat9k-sipbase.16.06.02.SPA.pkg
  Replacement: cc 1 0 cc_spa cat9k-sipspa.16.06.02.SPA.pkg
  Replacement: cc 10 0 cc cat9k-sipbase.16.06.02.SPA.pkg
  Replacement: cc 10 0 cc_spa cat9k-sipspa.16.06.02.SPA.pkg
  Replacement: cc 10 0 cc_srdriver cat9k-cc_srdriver.16.06.02.SPA.pkg
  Replacement: cc 2 0 cc_srdriver cat9k-cc_srdriver.16.06.02.SPA.pkg
  Replacement: cc 2 0 cc cat9k-sipbase.16.06.02.SPA.pkg
  Replacement: cc 2 0 cc_spa cat9k-sipspa.16.06.02.SPA.pkg
  Replacement: cc 3 0 cc_srdriver cat9k-cc_srdriver.16.06.02.SPA.pkg
  Replacement: cc 3 0 cc cat9k-sipbase.16.06.02.SPA.pkg
  Replacement: cc 3 0 cc_spa cat9k-sipspa.16.06.02.SPA.pkg
  Replacement: cc 4 0 cc_srdriver cat9k-cc_srdriver.16.06.02.SPA.pkg
  Replacement: cc 4 0 cc cat9k-sipbase.16.06.02.SPA.pkg
  Replacement: cc 4 0 cc_spa cat9k-sipspa.16.06.02.SPA.pkg
  Replacement: cc 5 0 cc_srdriver cat9k-cc_srdriver.16.06.02.SPA.pkg
  Replacement: cc 5 0 cc cat9k-sipbase.16.06.02.SPA.pkg
  Replacement: cc 5 0 cc_spa cat9k-sipspa.16.06.02.SPA.pkg
  Replacement: cc 6 0 cc_srdriver cat9k-cc_srdriver.16.06.02.SPA.pkg
  Replacement: cc 6 0 cc cat9k-sipbase.16.06.02.SPA.pkg
  Replacement: cc 6 0 cc_spa cat9k-sipspa.16.06.02.SPA.pkg
  Replacement: cc 7 0 cc_srdriver cat9k-cc_srdriver.16.06.02.SPA.pkg

```

```

Replacement: cc 7 0 cc cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 7 0 cc_spa cat9k-sipspace.16.06.02.SPA.pkg
Replacement: cc 8 0 cc_srdriver cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 8 0 cc cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 8 0 cc_spa cat9k-sipspace.16.06.02.SPA.pkg
Replacement: cc 9 0 cc_srdriver cat9k-cc_srdriver.16.06.02.SPA.pkg
Replacement: cc 9 0 cc cat9k-sipbase.16.06.02.SPA.pkg
Replacement: cc 9 0 cc_spa cat9k-sipspace.16.06.02.SPA.pkg
Replacement: fp 0 0 fp cat9k-espbase.16.06.02.SPA.pkg
Replacement: fp 1 0 fp cat9k-espbase.16.06.02.SPA.pkg
Replacement: rp 0 0 guestshell cat9k-guestshell.16.06.02.SPA.pkg
Replacement: rp 0 0 rp_base cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 0 0 rp_daemons cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 0 0 rp_iosd cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 0 0 rp_security cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 0 0 rp_webui cat9k-webui.16.06.02.SPA.pkg
Replacement: rp 0 0 rp_wlc cat9k-wlc.16.06.02.SPA.pkg
Replacement: rp 0 0 srdriver cat9k-srdriver.16.06.02.SPA.pkg
Replacement: rp 1 0 guestshell cat9k-guestshell.16.06.02.SPA.pkg
Replacement: rp 1 0 rp_base cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 1 0 rp_daemons cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 1 0 rp_iosd cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 1 0 rp_security cat9k-rpbase.16.06.02.SPA.pkg
Replacement: rp 1 0 rp_webui cat9k-webui.16.06.02.SPA.pkg
Replacement: rp 1 0 rp_wlc cat9k-wlc.16.06.02.SPA.pkg
Replacement: rp 1 0 srdriver cat9k-srdriver.16.06.02.SPA.pkg
Finished rollback impact
[1] Finished Abort on switch 1
Checking status of Abort on [1]
Abort: Passed on [1]
Finished Abort

```

```

/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function
[1]: Performing MCU Upgrade Service
/usr/binos/conf/provfunc.sh: line 8792: $!_log_file: ambiguous redirect
SUCCESS: MCU Upgrade Service finished
Install will reload the system now!
SUCCESS: install_abort Mon Oct 30 20:28:21 UTC 2017
/usr/binos/conf/chasutils.sh: line 428: chasfs_is_dominica: readonly function

```

The following is sample output from the **install activate auto-abort-timer** command:

```

Device# install activate auto-abort-timer 30

install_activate: START Mon Oct 30 20:42:28 UTC 2017
install_activate: Activating PACKAGE

*Oct 30 20:42:29.149: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 20:42:29 install_engine.sh:

%INSTALL-5-INSTALL_START_INFO: Started install activateFollowing packages shall be activated:
/flash/cat9k-wlc.16.06.02.prd9.SPA.pkg
/flash/cat9k-webui.16.06.02.prd9.SPA.pkg
/flash/cat9k-srdriver.16.06.02.prd9.SPA.pkg
/flash/cat9k-sipspace.16.06.02.prd9.SPA.pkg
/flash/cat9k-sipbase.16.06.02.prd9.SPA.pkg
/flash/cat9k-rpboot.16.06.02.prd9.SPA.pkg
/flash/cat9k-rpbase.16.06.02.prd9.SPA.pkg
/flash/cat9k-guestshell.16.06.02.prd9.SPA.pkg
/flash/cat9k-espbase.16.06.02.prd9.SPA.pkg
/flash/cat9k-cc_srdriver.16.06.02.prd9.SPA.pkg

This operation requires a reload of the system. Do you want to proceed? [y/n]
--- Starting Activate ---
Performing Activate on all members

```

```

[1] Activate package(s) on switch 1
--- Starting list of software package changes ---
Old files list:
  Removed cat9k-cc_srdriver.16.06.02.SPA.pkg
  Removed cat9k-espbase.16.06.02.SPA.pkg
  Removed cat9k-guestshell.16.06.02.SPA.pkg
  Removed cat9k-rpbase.16.06.02.SPA.pkg
  Removed cat9k-rpboot.16.06.02.SPA.pkg
  Removed cat9k-sipbase.16.06.02.SPA.pkg
  Removed cat9k-sipspa.16.06.02.SPA.pkg
  Removed cat9k-srdriver.16.06.02.SPA.pkg
  Removed cat9k-webui.16.06.02.SPA.pkg
  Removed cat9k-wlc.16.06.02.SPA.pkg
New files list:
  Added cat9k-cc_srdriver.16.06.02.prd9.SPA.pkg
  Added cat9k-espbase.16.06.02.prd9.SPA.pkg
  Added cat9k-guestshell.16.06.02.prd9.SPA.pkg
  Added cat9k-rpbase.16.06.02.prd9.SPA.pkg
  Added cat9k-rpboot.16.06.02.prd9.SPA.pkg
  Added cat9k-sipbase.16.06.02.prd9.SPA.pkg
  Added cat9k-sipspa.16.06.02.prd9.SPA.pkg
  Added cat9k-srdriver.16.06.02.prd9.SPA.pkg
  Added cat9k-webui.16.06.02.prd9.SPA.pkg
  Added cat9k-wlc.16.06.02.prd9.SPA.pkg
  Finished list of software package changes
[1] Finished Activate on switch 1
Checking status of Activate on [1]
Activate: Passed on [1]
Finished Activate

*Oct 30 20:43:39.249: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 20:43:39 rollback_timer.sh:

%INSTALL-5-INSTALL_AUTO_ABORT_TIMER_PROGRESS: Install auto abort timer will expire in 1800
seconds
Install will reload the system now!
SUCCESS: install_activate Mon Oct 30 20:43:44 UTC 2017

Device#
*Oct 30 20:43:44.615: %IOSXE-5-PLATFORM: Switch 1 R0/0: Oct 30 20:43:44 install_engine.sh:

%INSTALL-5-INSTALL_COMPLETED_INFO: Completed install activate PACKAGE
Chassis 1 reloading, reason - Reload command

```

Verifying Software Install

SUMMARY STEPS

1. enable
2. show install log
3. show install summary
4. show install package *filesystem: filename*
5. show install active
6. show install inactive
7. show install committed
8. show install uncommitted

DETAILED STEPS

Step 1 enable

Example:

```
Device> enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 show install log

Example:

```
Device# show install log
```

Displays information about all the software install operations that was performed since boot-up of the device.

```
Device# show install log
```

```
[0|install_op_boot]: START Sun Jun 11 15:01:37 Universal 2017
[0|install_op_boot]: END SUCCESS Sun Jun 11 15:01:44 Universal 2017
[1|install_commit]: START Mon Jun 12 07:27:31 UTC 2017
[1|install_commit(INFO, )]: Releasing transaction lock...
[1|install_commit(CONSOLE, )]: Committing PACKAGE
[remote|install_commit]: START Mon Jun 12 07:28:08 UTC 2017
[remote|install_commit(INFO, )]: Releasing transaction lock...
[remote|install_commit]: END SUCCESS Mon Jun 12 07:28:41 UTC 2017
[1|install_commit(INFO, )]: [1 2 3]: Performing Commit
SUCCESS: Commit finished
[1|install_commit(INFO, )]: install_commit: START Mon Jun 12 07:28:08 UTC 2017
SUCCESS: install_commit Mon Jun 12 07:28:41 UTC 2017
[1|install_commit(INFO, )]: Remote output from switch 2
[1|install_commit(INFO, )]: install_commit: START Mon Jun 12 07:28:12 UTC 2017
SUCCESS: install_commit Mon Jun 12 07:28:44 UTC 2017
[1|install_commit(INFO, )]: install_commit: START Mon Jun 12 07:28:12 UTC 2017
SUCCESS: install_commit Mon Jun 12 07:28:45 UTC 2017
[1|install_commit]: END SUCCESS Mon Jun 12 07:28:47 UTC 2017
```

Step 3 show install summary

Example:

```
Device# show install summary
```

Displays information about the image versions and their corresponding install state for all members/field-replaceable unit (FRU).

- The output of this command differs based on the **install** command that is executed.

```
Device# show install summary
```

```
[ R0 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
             C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type  St   Filename/Version
-----
IMG   I    16.6.2.0
```

```
IMG C 16.6.1.0
```

```
Device#
```

Step 4 show install package *filesystem: filename*

Example:

```
Device# show install package flash:cat9k_iosxe.16.06.01.SPA.bin
```

Displays information about the specified software install package file.

```
Device# show install package flash:cat9k_iosxe.16.06.01.SPA.bin
```

```
Package: cat9k_iosxe.16.06.01.SPA.bin
Size: 333806196
Timestamp: Sun Jun 11 14:47:23 2017 UTC
Canonical path: /flash/cat9k_iosxe.16.06.01.SPA.bin
```

```
Raw disk-file SHA1sum:
 5e9ef6ed1f7472b35eddd61df300e44b14b65ec4
Header size: 1000 bytes
Package type: 10002
Package flags: 0
Header version: 3
```

```
Internal package information:
Name: cc_srdriver
BuildTime:
ReleaseDate: Sun-27-Aug-17-09:05
BootArchitecture: none
RouteProcessor: cat9k
Platform: CAT9K
User: mcpre
PackageName: cc_srdriver
Build: BLD_V166_THROTTLE_LATEST_20170827_090555
CardTypes:
```

```
Package is not bootable.
Device#
```

Step 5 show install active

Example:

```
Device# show install active
```

Displays information about the active software install package.

```
Device# show install active
[ R0 ] Active Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
             C - Activated & Committed, D - Deactivated & Uncommitted
```

```
-----
Type St Filename/Version
-----
```

```
IMG C 16.6.2.0
```

Step 6 show install inactive

Example:


```
Device# show install inactive
```

Displays information about the inactive packages.

```
Device# show install inactive
```

```
[ R0 ] Inactive Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
           C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type  St  Filename/Version
-----
IMG   I   16.7.1.0
Device#
```

Step 7 show install committed

Example:

```
Device# show install committed
```

Displays information about committed packages.

```
Device# show install committed
[ R0 ] Committed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
           C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type  St  Filename/Version
-----
IMG   C   16.6.1.0
Device#
```

Step 8 show install uncommitted

Example:

```
Device# show install uncommitted
```

Displays information about uncommitted packages.

```
Device# show install uncommitted
[ R0 ] Uncommitted Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
           C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type  St  Filename/Version
-----
IMG   U   16.6.2.0
Device#
```

Example: Configuring a Device as a DHCP Server

```
Device# configure terminal
Device(config)# ip dhcp pool pool1
Device(dhcp-config)# network 10.10.10.0 255.255.255.0
Device(dhcp-config)# boot config-boot.text
```

Example: Configuring DHCP Auto-Image Update

```

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

```

Example: Configuring DHCP Auto-Image Update

```

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

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

```

Example: Configuring a Device 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:

```

Device# configure terminal
Device(config)# boot host dhcp
Device(config)# boot host retry timeout 300
Device(config)# banner config-save ^C Caution - Saving Configuration File to NVRAM May Cause
  You to No longer Automatically Download Configuration Files at Reboot^C
Device(config)# vlan 99
Device(config-vlan)# interface vlan 99
Device(config-if)# no shutdown
Device(config-if)# end
Device# 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)
Device#

```

Examples: Scheduling Software Image Reload

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

```

Device# 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 device at a future time:

```

Device# 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 Device Setup

Related Documents

Related Topic	Document Title
Device setup commands Boot loader commands	<i>Command Reference (Catalyst 9300 Series Switches)</i>
Hardware installation	<i>Cisco Catalyst 9300 Series Switches Hardware Installation Guide</i>

Feature History for Performing Device Setup Configuration

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Everest 16.5.1a	Device Setup Configuration	A device setup configuration can be performed, including auto configuration of IP address assignments and DHCP.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 4

Configuring Smart Licensing

- Prerequisites for Configuring Smart Licensing, on page 95
- Introduction to Smart Licensing, on page 95
- Connecting to CSSM, on page 96
- Linking Existing Licenses to CSSM, on page 98
- Configuring a Connection to CSSM and Setting Up the License Level, on page 98
- Registering a Device on CSSM, on page 110
- Monitoring Smart Licensing Configuration, on page 115
- Configuration Examples for Smart Licensing, on page 116
- Additional References, on page 122
- Feature Information for Smart Licensing, on page 123

Prerequisites for Configuring Smart Licensing

You must have the following in [CSSM](#):

- Cisco Smart Account
- One or more Virtual Account
- User role with proper access rights
- You should have accepted the Smart Software Licensing Agreement on CSSM to register devices.
- Network reachability to <https://tools.cisco.com>.

Introduction to Smart Licensing

Cisco Smart Licensing is a flexible licensing model that provides you with an easier, faster, and more consistent way to purchase and manage software across the Cisco portfolio and across your organization. And it's secure – you control what users can access. With Smart Licensing you get:

- Easy Activation: Smart Licensing establishes a pool of software licenses that can be used across the entire organization—no more PAKs (Product Activation Keys).

- **Unified Management:** My Cisco Entitlements (MCE) provides a complete view into all of your Cisco products and services in an easy-to-use portal, so you always know what you have and what you are using.
- **License Flexibility:** Your software is not node-locked to your hardware, so you can easily use and transfer licenses as needed.

To use Smart Licensing, you must first set up a Smart Account on Cisco Software Central (software.cisco.com).

For a more detailed overview on Cisco Licensing, go to cisco.com/go/licensingguide.



Note Licenses are managed as *Smart licenses* from Cisco IOS XE Fuji 16.9.1 and later. Right-to-Use licenses are deprecated from Cisco IOS XE Fuji 16.9.1.

Overview of CSSM

CSSM enables you to manage all your Cisco smart software licenses from one centralized portal. With CSSM, you can organize and view your licenses in groups called virtual accounts (collections of licenses and product instances).

You can access the CSSM on <https://software.cisco.com/#>, by clicking the **Smart Software Licensing** link under the **License** tab.



Note Use a Chrome 32.0, Firefox 25.0, or Safari 6.0.5 web browser to access CSSM. Also, ensure that Javascript 1.5 or a later version is enabled in your browser.

Use the CSSM to do the following tasks:

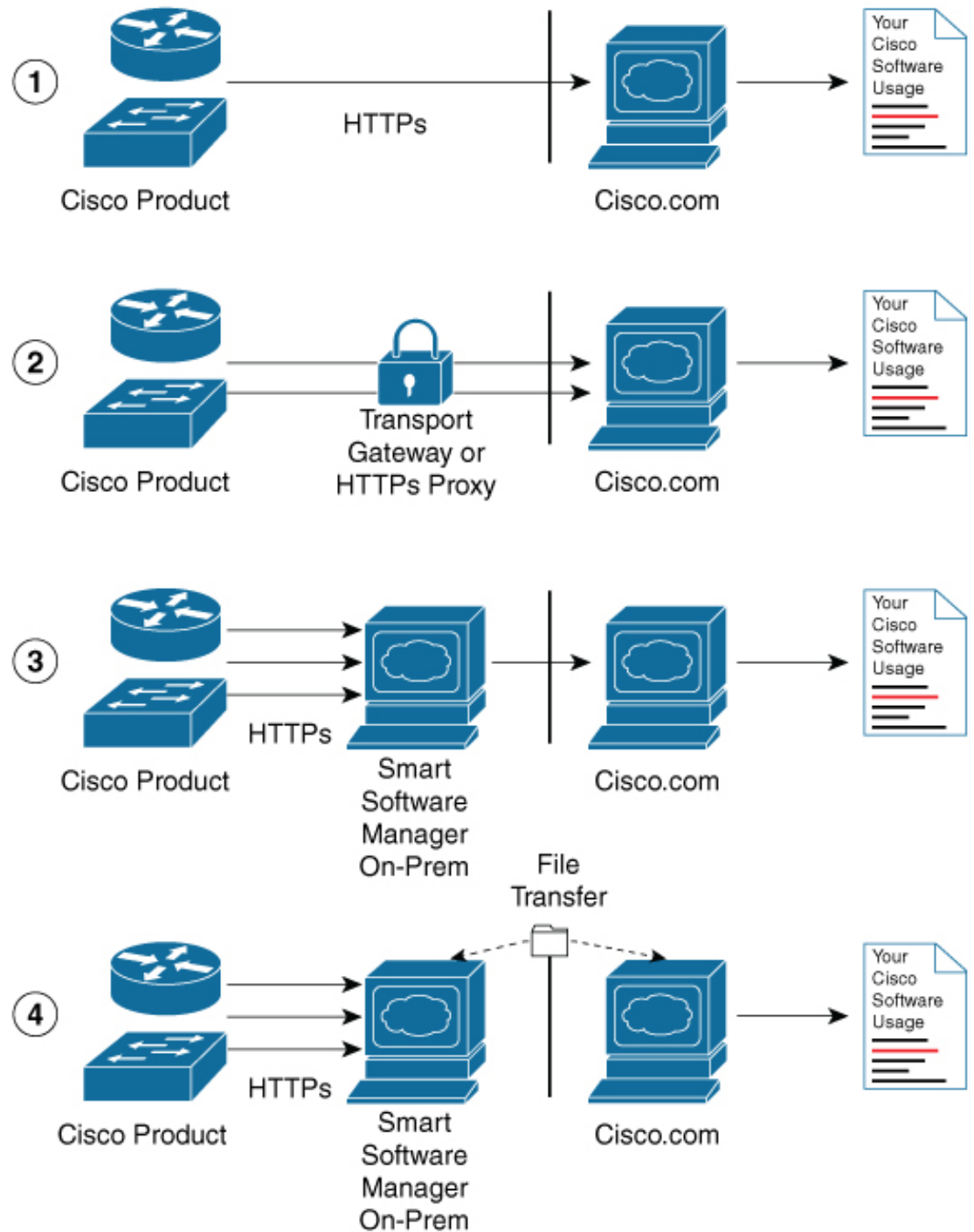
- Create, manage, or view virtual accounts.
- Create and manage Product Instance Registration Tokens.
- Transfer licenses between virtual accounts or view licenses.
- Transfer, remove, or view product instances.
- Run reports against your virtual accounts.
- Modify your email notification settings.
- View overall account information.

CSSM Help describes the procedures for carrying out these tasks.

Connecting to CSSM

The following illustration shows the various options available to connect to CSSM:

Figure 5: Connection Options



1. Direct cloud access: In this method, Cisco products send usage information directly over the internet to Cisco.com; no additional components are needed for the connection.

356271

2. Direct cloud access through an HTTPs proxy: In this method, Cisco products send usage information over the internet through a proxy server - either a Call Home Transport Gateway or an off-the-shelf proxy (such as Apache) to Cisco.com.
3. Mediated access through a connected on-premises collector: In this method, Cisco products send usage information to a locally-connected collector, which acts as a local license authority. Periodically, this information is exchanged to keep the databases synchronized.
4. Mediated access through a disconnected on-premises collector: In this method, Cisco products send usage information to a local disconnected collector, which acts as a local license authority. Exchange of human-readable information takes place occasionally (maybe once a month) to keep the databases synchronized.

Options 1 and 2 provide an easy connection option, and options 3 and 4 provide a secure environment connection option. Cisco Smart Software Manager On-Prem (formerly known as Cisco Smart Software Manager satellite) provides support for options 3 and 4.

Linking Existing Licenses to CSSM

The following section is required for those licenses that were purchased without a Cisco Smart Account. These licenses will not be available in CSSM after you have upgraded to Cisco IOS XE Fuji 16.9.1. You are requested to contact the Cisco Global Licensing Operations (GLO) team with the following email template. Fill the template with the appropriate information to request linking of your existing licenses to your Cisco Smart Account in CSSM.

Email Template:

To: licensing@cisco.com

Subject: Request for Linking Existing Licenses to Cisco Smart Account

Email Text:

Cisco.com ID: #####

Smart virtual account name: #####

Smart account domain ID (domain in the form of "xyz.com"): #####

List of UDIs:

List of licenses with count:

Proof of purchase (*Please attach your proof of purchase along with this mail*)

Configuring a Connection to CSSM and Setting Up the License Level

The following sections provide information about how to set up a connection to CSSM and set up the license level.

Setting Up a Connection to CSSM

The following steps show how to set up a Layer 3 connection to CSSM to verify network reachability. Skip this section if you already have Layer 3 connectivity to CSSM.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **{ip | ipv6} name-server** *server-address 1* [*server-address 2*] [*server-address 3*] [*server-address 4*] [*server-address 5*] [*server-address 6*]
4. **ip name-server vrf Mgmt-vrf** *server-address 1* [*server-address 2*] [*server-address 3*] [*server-address 4*] [*server-address 5*] [*server-address 6*]
5. **ip domain lookup source-interface** *interface-type interface-number*
6. **ip domain name** *example.com*
7. **ip host** *tools.cisco.com ip-address*
8. **interface** *vlan_id*
9. **ntp server** *ip-address* [*version number*] [**key** *key-id*] [**prefer**]
10. **switchport access vlan** *vlan_id*
11. **ip route** *ip-address ip-mask subnet mask*
12. **license smart transport callhome**
13. **ip http client source-interface** *interface-type interface-number*
14. **exit**
15. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	{ip ipv6} name-server <i>server-address 1</i> [<i>server-address 2</i>] [<i>server-address 3</i>] [<i>server-address 4</i>] [<i>server-address 5</i>] [<i>server-address 6</i>] Example: Device(config)# ip name-server 209.165.201.1 209.165.200.225 209.165.201.14 209.165.200.230	Configures Domain Name System (DNS).
Step 4	ip name-server vrf Mgmt-vrf <i>server-address 1</i> [<i>server-address 2</i>] [<i>server-address 3</i>] [<i>server-address 4</i>] [<i>server-address 5</i>] [<i>server-address 6</i>] Example:	(Optional) Configures DNS on the VRF interface. Note You should configure this command as an alternative to the ip name-server command.

	Command or Action	Purpose
	<pre>Device(config)# ip name-server vrf Mgmt-vrf 209.165.201.1 209.165.200.225 209.165.201.14 209.165.200.230</pre>	
Step 5	<p>ip domain lookup source-interface <i>interface-type interface-number</i></p> <p>Example:</p> <pre>Device(config)# ip domain lookup source-interface Vlan100</pre>	(Optional) Configures the source interface for the DNS domain lookup.
Step 6	<p>ip domain name <i>example.com</i></p> <p>Example:</p> <pre>Device(config)# ip domain name example.com</pre>	Configures the domain name.
Step 7	<p>ip host <i>tools.cisco.com ip-address</i></p> <p>Example:</p> <pre>Device(config)# ip host tools.cisco.com 209.165.201.30</pre>	(Optional) Configures static hostname-to-address mappings in the DNS hostname cache if automatic DNS mapping is not available.
Step 8	<p>interface <i>vlan_id</i></p> <p>Example:</p> <pre>Device(config)# interface Vlan100 Device(config-if)# ip address 192.0.2.10 255.255.255.0 Device(config-if)# exit</pre>	Configures a Layer 3 interface.
Step 9	<p>ntp server <i>ip-address [version number] [key key-id] [prefer]</i></p> <p>Example:</p> <pre>Device(config)# ntp server 198.51.100.100 version 2 prefer</pre>	Forms a server association with the specified system. Note The ntp server command is mandatory to ensure that the device time is synchronized with CSSM.
Step 10	<p>switchport access vlan <i>vlan_id</i></p> <p>Example:</p> <pre>Device(config)# interface GigabitEthernet1/0/1 Device(config-if)# switchport access vlan 100 Device(config-if)# switchport mode access Device(config-if)# exit Device(config)#</pre>	(Optional) Enables the VLAN for which this access port carries traffic and sets the interface as a nontrunking nontagged single-VLAN Ethernet interface. Note This step is to be configured only if the switchport access mode is required.
Step 11	<p>ip route <i>ip-address ip-mask subnet mask</i></p> <p>Example:</p> <pre>Device(config)# ip route 192.0.2.0 255.255.255.255 192.0.2.1</pre>	Configures a route on the device. Note You can configure either a static route or a dynamic route.
Step 12	license smart transport callhome	Enables the transport mode as Call Home.

	Command or Action	Purpose
	Example: Device(config)# <code>license smart transport callhome</code>	Note The <code>license smart transport callhome</code> command is mandatory.
Step 13	ip http client source-interface <i>interface-type</i> <i>interface-number</i> Example: Device(config)# <code>ip http client source-interface Vlan100</code>	Configures a source interface for the HTTP client. Note The <code>ip http client source-interface interface-type interface-number</code> command is mandatory.
Step 14	exit Example: Device(config)# <code>exit</code>	(Optional) Exits global configuration mode and returns to privileged EXEC mode.
Step 15	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring the Call Home Service for Direct Cloud Access



Note By default, the CiscoTAC-1 profile is already set up on the device. Use the `show call-home profile all` command to check the profile status.

The Call Home service provides email-based and web-based notification of critical system events to CSSM.

To configure and enable the Call Home service, perform this procedure:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `call-home`
4. `no http secure server-identity-check`
5. `contact-email-address email-address`
6. `profile CiscoTAC-1`
7. `destination transport-method http`
8. `destination address http url`
9. `active`
10. `no destination transport-method email`
11. `exit`
12. `exit`
13. `service call-home`
14. `exit`
15. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	call-home Example: Device (config)# call-home	Enters Call Home configuration mode.
Step 4	no http secure server-identity-check Example: Device (config-call-home)# no http secure server-identity-check	Disables server identity check when HTTP connection is established.
Step 5	contact-email-address <i>email-address</i> Example: Device (config-call-home)# contact-email-addr username@example.com	Assigns customer's email address. You can enter up to 200 characters in email address format with no spaces.
Step 6	profile CiscoTAC-1 Example: Device (config-call-home)# profile CiscoTAC-1	By default, the CiscoTAC-1 profile is inactive. To use this profile with the Call Home service, you must enable the profile.
Step 7	destination transport-method http Example: Device (config-call-home-profile)# destination transport-method http	Enables the Call Home service via HTTP.
Step 8	destination address http <i>url</i> Example: Device (config-call-home-profile)# destination address http https://tools.cisco.com/its/service/oddce/services/DDCEService	Connects to CSSM.
Step 9	active Example: Device (config-call-home-profile)# active	Enables the destination profile.
Step 10	no destination transport-method email Example:	Disables the Call Home service via email.

	Command or Action	Purpose
	<code>Device(config-call-home-profile)# no destination transport-method email</code>	
Step 11	exit Example: <code>Device(config-call-home-profile)# exit</code>	Exits Call Home destination profile configuration mode and returns to Call Home configuration mode.
Step 12	exit Example: <code>Device(config-call-home)# exit</code>	Exits Call Home configuration mode and returns to global configuration mode.
Step 13	service call-home Example: <code>Device(config)# service call-home</code>	Enables the Call Home feature.
Step 14	exit Example: <code>Device(config)# exit</code>	Exits global configuration mode and returns to privileged EXEC mode.
Step 15	copy running-config startup-config Example: <code>Device# copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring the Call Home Service for Direct Cloud Access through an HTTPs Proxy Server

The Call Home service can be configured through an HTTPs proxy server. This configuration requires no user authentication to connect to CSSM.



Note Authenticated HTTPs proxy configurations are not supported.

To configure and enable the Call Home service through an HTTPs proxy, perform this procedure:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **call-home**
4. **contact-email-address** *email-address*
5. **http-proxy** *proxy-address* **proxy-port** *port-number*
6. **profile** CiscoTAC-1
7. **destination transport-method** http
8. **no destination transport-method** email

9. **profile** *name*
10. **reporting smart-licensing-data**
11. **destination transport-method** **http**
12. **destination address** **http** *url*
13. **active**
14. **exit**
15. **exit**
16. **service call-home**
17. **ip http client proxy-server** *proxy-address* **proxy-port** *port-number*
18. **exit**
19. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	call-home Example: Device(config)# call-home	Enters Call Home configuration mode.
Step 4	contact-email-address <i>email-address</i> Example: Device(config-call-home)# contact-email-addr sch-smart-licensing@cisco.com	Configures the default email address as sch-smart-licensing@cisco.com.
Step 5	http-proxy <i>proxy-address</i> proxy-port <i>port-number</i> Example: Device(config-call-home)# http-proxy 198.51.100.10 port 3128	Configures the proxy server information to the Call Home service.
Step 6	profile CiscoTAC-1 Example: Device(config-call-home)# profile CiscoTAC-1	By default, the CiscoTAC-1 profile is inactive. To use this profile with the Call Home service, you must enable the profile.
Step 7	destination transport-method http Example: Device(config-call-home-profile)# destination transport-method http	Enables the Call Home service via HTTP.

	Command or Action	Purpose
Step 8	no destination transport-method email Example: Device(config-call-home-profile)# no destination transport-method email	Disables the Call Home service via email.
Step 9	profile name Example: Device(config-call-home)# profile test1	Enters Call Home destination profile configuration mode for the specified destination profile name. If the specified destination profile does not exist, it is created.
Step 10	reporting smart-licensing-data Example: Device(config-call-home-profile)# reporting smart-licensing-data	Enables data sharing with the Call Home service via HTTP.
Step 11	destination transport-method http Example: Device(config-call-home-profile)# destination transport-method http	Enables the HTTP message transport method.
Step 12	destination address http url Example: Device(config-call-home-profile)# destination address http https://tools.cisco.com/its/service/odce/services/DDCEService	Connects to CSSM.
Step 13	active Example: Device(config-call-home-profile)# active	Enables the destination profile.
Step 14	exit Example: Device(config-call-home-profile)# exit	Exits Call Home destination profile configuration mode and returns to Call Home configuration mode.
Step 15	exit Example: Device(config-call-home)# exit	Exits Call Home configuration mode and returns to global configuration mode.
Step 16	service call-home Example: Device(config)# service call-home	Enables the Call Home feature.
Step 17	ip http client proxy-server proxy-address proxy-port port-number Example: Device(config)# ip http client proxy-server 198.51.100.10 port 3128	Enables the Call Home feature.

	Command or Action	Purpose
Step 18	exit Example: Device(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.
Step 19	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring the Call Home Service for Cisco Smart Software Manager On-Prem

For information about Cisco Smart Software Manager On-Prem (formerly known as Cisco Smart Software Manager satellite), see <https://www.cisco.com/c/en/us/buy/smart-accounts/software-manager-satellite.html>.

To configure the Call Home service for the Cisco Smart Software Manager On-Prem (formerly known as Cisco Smart Software Manager satellite), perform this procedure:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **call-home**
4. **no http secure server-identity-check**
5. **profile name**
6. **reporting smart-licensing-data**
7. **destination transport-method http**
8. **destination address http url**
9. **destination preferred-msg-format {long-text | short-text | xml}**
10. **active**
11. **exit**
12. **exit**
13. **ip http client source-interface interface-type interface-number**
14. **crypto pki trustpoint name**
15. **revocation-check none**
16. **end**
17. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	call-home Example: Device(config)# call-home	Enters Call Home configuration mode.
Step 4	no http secure server-identity-check Example: Device(config-call-home)# no http secure server-identity-check	Disables server identity check when HTTP connection is established.
Step 5	profile name Example: Device(config-call-home)# profile test1	Enters Call Home destination profile configuration mode for the specified destination profile name. If the specified destination profile does not exist, it is created.
Step 6	reporting smart-licensing-data Example: Device(config-call-home-profile)# reporting smart-licensing-data	Enables data sharing with the Call Home service via HTTP.
Step 7	destination transport-method http Example: Device(config-call-home-profile)# destination transport-method http	Enables the HTTP message transport method.
Step 8	destination address http url Example: Device(config-call-home-profile)# destination address http <code>https://209.165.201.15:443/Transportgateway/services/DeviceRequestHandler</code> OR Device(config-call-home-profile)# destination address http <code>http://209.165.201.15:80/Transportgateway/services/DeviceRequestHandler</code>	Configures the destination URL (CSSM) to which Call Home messages are sent. Note Ensure the IP address or the fully qualified domain name (FQDN) in the destination URL matches the IP address or the FQDN as configured for the Satellite Name on the Cisco Smart Software Manager On-Prem.
Step 9	destination preferred-msg-format {long-text short-text xml} Example: Device(config-call-home-profile)# destination preferred-msg-format xml	(Optional) Configures a preferred message format. The default is XML.
Step 10	active Example:	Enables the destination profile. By default, a profile is enabled when it is created.

	Command or Action	Purpose
	Device(config-call-home-profile)# active	
Step 11	exit Example: Device(config-call-home-profile)# exit	Exits Call Home destination profile configuration mode and returns to Call Home configuration mode.
Step 12	exit Example: Device(config-call-home)# exit	Exits Call Home configuration mode and returns to global configuration mode.
Step 13	ip http client source-interface <i>interface-type</i> <i>interface-number</i> Example: Device(config)# ip http client source-interface Vlan100	Configures a source interface for the HTTP client. Note The ip http client source-interface <i>interface-type</i> <i>interface-number</i> command is mandatory for a vrf interface.
Step 14	crypto pki trustpoint <i>name</i> Example: Device(config)# crypto pki trustpoint SLA-TrustPoint	(Optional) Declares the trustpoint and a given name and enters ca-trustpoint configuration mode.
Step 15	revocation-check none Example: Device(ca-trustpoint)# revocation-check none	(Optional) Specifies that certificate checking is ignored.
Step 16	end Example: Device(ca-trustpoint)# end	(Optional) Exits ca-trustpoint configuration mode and returns to privileged EXEC mode.
Step 17	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring the License Level

This procedure is optional. You can use this procedure to :

- Downgrade or upgrade licenses.
- Enable or disable an evaluation or extension license
- Clear an upgrade license

The required license level(s) needs to be configured on the device before registering. The following are the license levels available for Cisco Catalyst 9000 Series Switches:

Base licenses

- Network Essentials
- Network Advantage (includes Network Essentials)

Add-on licenses—These can be subscribed for a fixed term of three, five, or seven years.

- Digital Networking Architecture (DNA) Essentials
- DNA Advantage (includes DNA Essentials)

To configure the license levels, follow this procedure:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **license boot level** *license_level*
4. **exit**
5. **write memory**
6. **show version**
7. **reload**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	license boot level <i>license_level</i> Example: Device(config)# license boot level network-essentials	Activates the licenses on the switch.
Step 4	exit Example: Device(config)# exit	Returns to the privileged EXEC mode.
Step 5	write memory Example: Device# write memory	Saves the license information on the switch.
Step 6	show version Example:	Shows license-level information.

	Command or Action	Purpose
	Device# show version	
	Technology-package Current Type Technology-package Next reboot	
	network-essentials Smart License network-essentials None Subscription Smart License None	
Step 7	reload Example: Device# reload	Reloads the device.

Registering a Device on CSSM

To register a device on CSSM, you must do the following tasks:

1. Generate a unique token from the CSSM.
2. Register the device with the generated token.

On successful registration, the device will receive an identity certificate. This certificate is saved on your device and automatically used for all future communications with Cisco. CSSM will attempt to renew the registration information every 30 days..

Additionally, license usage data is collected and a report is sent to you every month. If required, you can configure your Call Home settings to filter out sensitive information (like hostname, username and password) from the usage report.



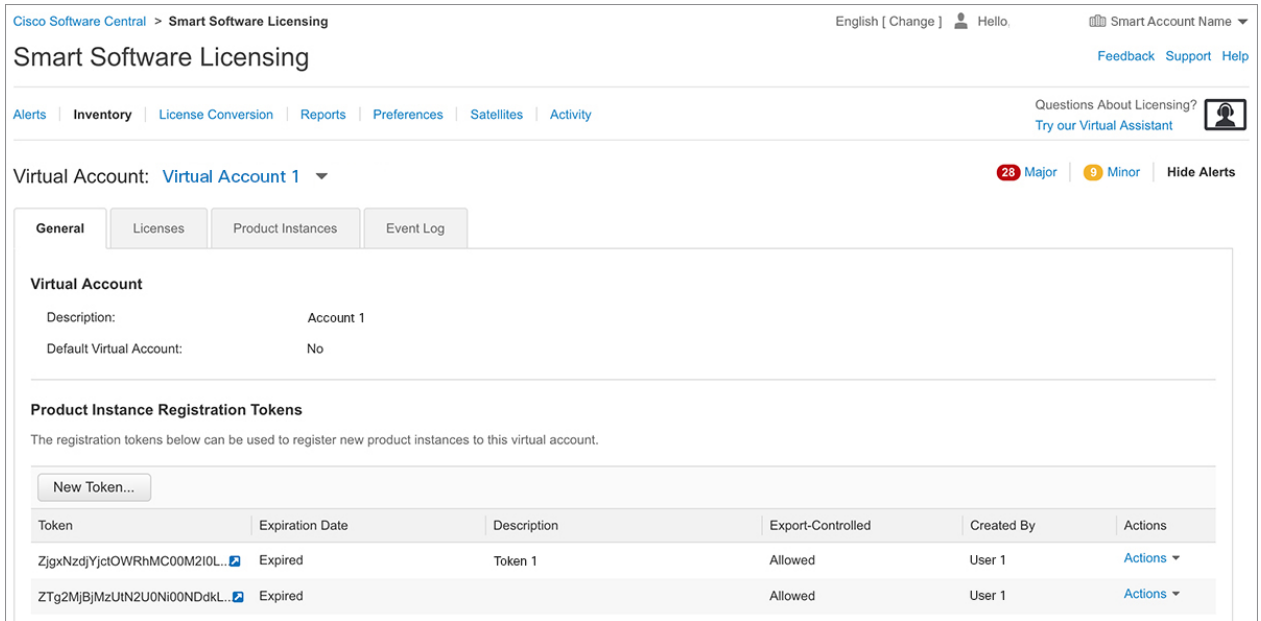
Note Downgrading a device from Cisco IOS XE Fuji 16.9.1 to any prior release will migrate the smart license to traditional license. All smart license information on the device will be removed. In case the device needs to be upgraded back to Cisco IOS XE Fuji 16.9.1, the license status will remain in evaluation mode until the device is registered again in CSSM.

Generating a New Token from CSSM

Tokens are generated to register new product instances to the virtual account.

-
- Step 1** Log in to CSSM from <https://software.cisco.com/#>.
You must log in to the portal using the username and password provided by Cisco.
- Step 2** Click the **Inventory** tab.
- Step 3** From the **Virtual Account** drop-down list, choose the required virtual account.
- Step 4** Click the **General** tab.

Step 5 Click **New Token**.

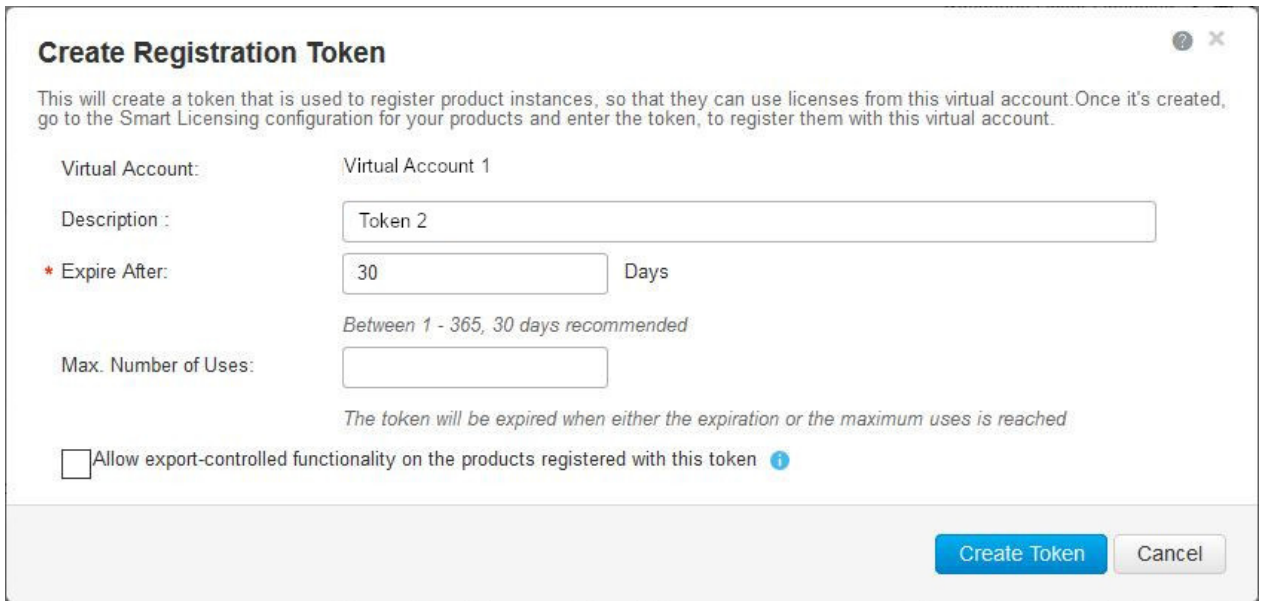


The **Create Registration Token** window is displayed.

Step 6 In the **Description** field, enter the token description.

Step 7 In the **Expire After** field, enter the number of days the token must be active.

Step 8 (Optional) In the **Max. Number of Uses** field, enter the maximum number of uses allowed after which the token expires.



Step 9 Check the **Allow export-controlled functionality on the products registered with this token** checkbox.

Enabling this checkbox ensures Cisco compliance with US and country-specific export policies and guidelines. For more information, see <https://www.cisco.com/c/en/us/about/legal/global-export-trade.html>.

Step 10 Click **Create Token** to create a token.

Step 11 After the token is created, click **Copy** to copy the newly created token.

Create Registration Token ? X

This will create a token that is used to register product instances, so that they can use licenses from this virtual account. Once it's created, go to the Smart Licensing configuration for your products and enter the token, to register them with this virtual account.

Virtual Account: Virtual Account 1

Description:

* Expire After: Days
Between 1 - 365, 30 days recommended

Max. Number of Uses:

The token will be expired when either the expiration or the maximum uses is reached

Allow export-controlled functionality on the products registered with this token i

Create Token
Cancel

Registering a Device with the New Token

To register a device with the new token, perform this procedure:

SUMMARY STEPS

1. enable
2. license smart register idtoken *token_ID*
3. write memory

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	license smart register idtoken <i>token_ID</i> Example: Device# license smart register idtoken \$T14UytrNXBzEs1ck8veUtWAG5abnZJOFdDa1FwbVRa%0Ab1FMbz0%3D%0A	Registers the device with the back-end server using the token generated from CSSM.
Step 3	write memory Example: Device# write memory	Saves the license information on the device.

Verifying the License Status After Registration

To verify the status of a license after registration, use the **show license all** command.

```
Device> enable
Device# show license all
Smart Licensing Status
=====

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: Smart Account Name
  Virtual Account: Virtual Account 1
  Export-Controlled Functionality: Allowed
  Initial Registration: SUCCEEDED on Jul 13 09:30:40 2018 EDT
  Last Renewal Attempt: None
  Next Renewal Attempt: Jan 09 09:30:40 2019 EDT
  Registration Expires: Jul 13 09:25:31 2019 EDT

License Authorization:
  Status: AUTHORIZED on Jul 13 09:30:45 2018 EDT
  Last Communication Attempt: SUCCEEDED on Jul 13 09:30:45 2018 EDT
  Next Communication Attempt: Aug 12 09:30:45 2018 EDT
  Communication Deadline: Oct 11 09:25:40 2018 EDT

Utility:
  Status: DISABLED

Data Privacy:
  Sending Hostname: yes
  Callhome hostname privacy: DISABLED
  Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED

Transport:
  Type: Callhome

License Usage
=====

C9300 DNA Advantage (C9300-24 DNA Advantage):
  Description: C9300-24P DNA Advantage
  Count: 3
  Version: 1.0
  Status: AUTHORIZED

C9300 Network Advantage (C9300-24 Network Advantage):
  Description: C9300-24P Network Advantage
  Count: 3
  Version: 1.0
  Status: AUTHORIZED

Product Information
=====
UDI: PID:C9300-24U,SN:FCW2125L046

HA UDI List:
  Active:PID:C9300-24U,SN:FCW2125L046
  Standby:PID:C9300-24U,SN:FCW2125L03U
  Member:PID:C9300-24U,SN:FCW2125G01T
```

```

Agent Version
=====
Smart Agent for Licensing: 4.4.13_rel/116
Component Versions: SA:(1_3_dev)1.0.15, SI:(dev22)1.2.1, CH:(rel5)1.0.3, PK:(dev18)1.0.3

Reservation Info
=====
License reservation: DISABLED

```

Canceling a Device's Registration in CSSM

When your device is taken off the inventory, shipped elsewhere for redeployment, or returned to Cisco for replacement using the return merchandise authorization (RMA) process, you can use the **deregister** command to cancel the registration of your device.

To cancel device registration, follow this procedure:

Before you begin

Layer 3 connection to CSSM must be available to successfully deregister the device.

SUMMARY STEPS

1. **enable**
2. **license smart deregister**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	license smart deregister Example: Device# license smart deregister	Cancels the device's registration, and sends the device into evaluation mode. All smart licensing entitlements and certificates on the corresponding platform are removed. The device product instance stored on CSSM is also removed.

Monitoring Smart Licensing Configuration

Use the following commands in privileged EXEC mode to monitor smart licensing configuration.

Table 6: Commands to Monitor Smart Licensing Configuration

Command	Purpose
show license status	<p>Displays the compliance status of smart licensing. The following is the list of possible statuses:</p> <ul style="list-style-type: none"> • Enabled: Indicates that smart licensing is enabled. • Waiting: Indicates the initial state after your device has made a license entitlement request. The device establishes communication with Cisco and successfully registers itself with the CSSM. • Registered: Indicates that your device is able to communicate with the CSSM, and is authorized to initiate requests for license entitlements. • Authorized: Indicates that your device is in Compliance status and is authorized to use the requested type and count of licenses. The Authorization status has a lifetime of 90 days. At the end of 30 days, the device will send a new entitlement authorization request to the CSSM to renew the authorization. • Out Of Compliance: Indicates that one or more of your licenses are out of compliance. You must buy additional licenses. • Eval Mode: You must register the device with the CSSM within 90 days (of device usage). Otherwise, your device's evaluation period will expire. • Evaluation Period Expired: At the end of 90 days, if your device has not registered, the device enters Evaluation Expired mode.
show license all	Displays all the entitlements in use. Additionally, it shows the associated licensing certificates, compliance status, UDI, and other details.
show tech-support license	Displays the detailed debug output.
show license usage	Displays the license usage information.

Command	Purpose
<code>show license summary</code>	Displays the summary of all the active licenses.

Configuration Examples for Smart Licensing

The following sections provide various Smart Licensing configuration examples.

Example: Viewing the Call Home Profile

Example

To display the Call Home profile, use the `show call-home profile all` command:

```
Device> enable
Device# show call-home profile all
Profile Name: CiscoTAC-1
  Profile status: ACTIVE
  Profile mode: Full Reporting
  Reporting Data: Smart Call Home, Smart Licensing
  Preferred Message Format: xml
  Message Size Limit: 3145728 Bytes
  Transport Method: http
  HTTP address(es): https://tools.cisco.com/its/service/odce/services/DDCEService
  Other address(es): default

  Periodic configuration info message is scheduled every 1 day of the month at 09:15
  Periodic inventory info message is scheduled every 1 day of the month at 09:00

Alert-group          Severity
-----
crash                debug
diagnostic           minor
environment          warning
inventory            normal

Syslog-Pattern      Severity
-----
APF-.-WLC_.*       warning
.*                  major
```

Example: Viewing the License Information Before Registering

Example

To display the license entitlements, use the `show license all` command:

```
Device> enable
Device# show license all
Load for five secs: 1%/0%; one minute: 0%; five minutes: 0%
```

No time source, 09:28:07.210 EDT Fri Jul 13 2018

Smart Licensing Status
=====

Smart Licensing is ENABLED

Registration:

Status: UNREGISTERED
Export-Controlled Functionality: Not Allowed

License Authorization:

Status: EVAL MODE
Evaluation Period Remaining: 68 days, 0 hours, 30 minutes, 5 seconds

Utility:

Status: DISABLED

Data Privacy:

Sending Hostname: yes
Callhome hostname privacy: DISABLED
Smart Licensing hostname privacy: DISABLED
Version privacy: DISABLED

Transport:

Type: Callhome

License Usage
=====

(C9300-24 DNA Advantage):

Description:
Count: 3
Version: 1.0
Status: EVAL MODE

(C9300-24 Network Advantage):

Description:
Count: 3
Version: 1.0
Status: EVAL MODE

Product Information
=====

UDI: PID:C9300-24U,SN:FCW2125L046

HA UDI List:

Active:PID:C9300-24U,SN:FCW2125L046
Standby:PID:C9300-24U,SN:FCW2125L03U
Member:PID:C9300-24U,SN:FCW2125G01T

Agent Version
=====

Smart Agent for Licensing: 4.4.13_rel/116
Component Versions: SA:(1_3_dev)1.0.15, SI:(dev22)1.2.1, CH:(rel15)1.0.3, PK:(dev18)1.0.3

Reservation Info
=====

License reservation: DISABLED

Example

To display the license usage information, use the **show license usage** command:

```
Device> enable
Device# show license usage
Load for five secs: 1%/0%; one minute: 0%; five minutes: 0%
No time source, 09:28:34.123 EDT Fri Jul 13 2018

License Authorization:
  Status: EVAL MODE
  Evaluation Period Remaining: 68 days, 0 hours, 29 minutes, 38 seconds

(C9300-24 DNA Advantage):
  Description:
  Count: 3
  Version: 1.0
  Status: EVAL MODE

(C9300-24 Network Advantage):
  Description:
  Count: 3
  Version: 1.0
  Status: EVAL MODE
```

Example

To display all the license summaries, use the **show license summary** command:

```
Device> enable
Device# show license summary
Load for five secs: 1%/0%; one minute: 0%; five minutes: 0%
No time source, 09:28:39.986 EDT Fri Jul 13 2018

Smart Licensing is ENABLED

Registration:
  Status: UNREGISTERED
  Export-Controlled Functionality: Not Allowed

License Authorization:
  Status: EVAL MODE
  Evaluation Period Remaining: 68 days, 0 hours, 29 minutes, 33 seconds

License Usage:
  License                               Entitlement tag                Count Status
  -----
                               (C9300-24 DNA Advantage)      3 EVAL MODE
                               (C9300-24 Network Advan...)  3 EVAL MODE
```

Example

To display the license status information, use the **show license status** command:

```
Device> enable
Device# show license status
```

```

Load for five secs: 1%/0%; one minute: 0%; five minutes: 0%
No time source, 09:28:37.683 EDT Fri Jul 13 2018

Smart Licensing is ENABLED

Utility:
  Status: DISABLED

Data Privacy:
  Sending Hostname: yes
  Callhome hostname privacy: DISABLED
  Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED

Transport:
  Type: Callhome

Registration:
  Status: UNREGISTERED
  Export-Controlled Functionality: Not Allowed

License Authorization:
  Status: EVAL MODE
  Evaluation Period Remaining: 68 days, 0 hours, 29 minutes, 35 seconds

```

Example: Registering a Device

Example

To register a device, use the **license smart register idtoken** command:

```

Device> enable
Device# license smart register idtoken
Tl4UytrNXBzbEs1ck8veUtWaG5abnZJOFdDa1FwbVRa%0Ab1RMbz0%3D%0A
Device# write memory

```

Example: Viewing the License Status After Registering

Example

To display the license entitlements, use the **show license all** command:

```

Device> enable
Device# show license all
Load for five secs: 0%/0%; one minute: 2%; five minutes: 1%
No time source, 09:31:16.387 EDT Fri Jul 13 2018

Smart Licensing Status
=====

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: Smart Account Name

```

```

Virtual Account: Virtual Account 1
Export-Controlled Functionality: Allowed
Initial Registration: SUCCEEDED on Jul 13 09:30:40 2018 EDT
Last Renewal Attempt: None
Next Renewal Attempt: Jan 09 09:30:40 2019 EDT
Registration Expires: Jul 13 09:25:31 2019 EDT

License Authorization:
  Status: AUTHORIZED on Jul 13 09:30:45 2018 EDT
  Last Communication Attempt: SUCCEEDED on Jul 13 09:30:45 2018 EDT
  Next Communication Attempt: Aug 12 09:30:45 2018 EDT
  Communication Deadline: Oct 11 09:25:40 2018 EDT

Utility:
  Status: DISABLED

Data Privacy:
  Sending Hostname: yes
  Callhome hostname privacy: DISABLED
  Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED

Transport:
  Type: Callhome

License Usage
=====

C9300 DNA Advantage (C9300-24 DNA Advantage):
  Description: C9300-24P DNA Advantage
  Count: 3
  Version: 1.0
  Status: AUTHORIZED

C9300 Network Advantage (C9300-24 Network Advantage):
  Description: C9300-24P Network Advantage
  Count: 3
  Version: 1.0
  Status: AUTHORIZED

Product Information
=====
UDI: PID:C9300-24U,SN:FCW2125L046

HA UDI List:
  Active:PID:C9300-24U,SN:FCW2125L046
  Standby:PID:C9300-24U,SN:FCW2125L03U
  Member:PID:C9300-24U,SN:FCW2125G01T

Agent Version
=====
Smart Agent for Licensing: 4.4.13_rel/116
Component Versions: SA:(1_3_dev)1.0.15, SI:(dev22)1.2.1, CH:(rel5)1.0.3, PK:(dev18)1.0.3

Reservation Info
=====
License reservation: DISABLED

```

Example

To display license usage information, use the **show license usage** command:

```

Device> enable
Device# show license usage
License Authorization:
  Status: AUTHORIZED on Jul 17 09:47:28 2018 EDT

C9300 DNA Advantage (C9300-24 DNA Advantage):
  Description: C9300-24P DNA Advantage
  Count: 3
  Version: 1.0
  Status: AUTHORIZED

C9300 Network Advantage (C9300-24 Network Advantage):
  Description: C9300-24P Network Advantage
  Count: 3
  Version: 1.0
  Status: AUTHORIZED

```

Example

To display all the license summaries, use the **show license summary** command:

```

Device> enable
Device# show license summary
Load for five secs: 1%/0%; one minute: 1%; five minutes: 1%
No time source, 09:32:13.746 EDT Fri Jul 13 2018

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: Smart Account Name
  Virtual Account: Virtual Account 1
  Export-Controlled Functionality: Allowed
  Last Renewal Attempt: None
  Next Renewal Attempt: Jan 09 09:30:40 2019 EDT

License Authorization:
  Status: AUTHORIZED
  Last Communication Attempt: SUCCEEDED
  Next Communication Attempt: Aug 12 09:30:44 2018 EDT

License Usage:

```

License	Entitlement tag	Count	Status
C9300 DNA Advantage	(C9300-24 DNA Advantage)	3	AUTHORIZED
C9300 Network Advantage	(C9300-24 Network Advan...)	3	AUTHORIZED

Example

To display the license status information, use the **show license status** command:

```

Device> enable
Device# show license status
Load for five secs: 0%/0%; one minute: 1%; five minutes: 1%
No time source, 09:32:00.191 EDT Fri Jul 13 2018

Smart Licensing is ENABLED

```

```

Utility:
  Status: DISABLED

Data Privacy:
  Sending Hostname: yes
  Callhome hostname privacy: DISABLED
  Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED

Transport:
  Type: Callhome

Registration:
  Status: REGISTERED
  Smart Account: Smart Account Name
  Virtual Account: Virtual Account 1
  Export-Controlled Functionality: Allowed
  Initial Registration: SUCCEEDED on Jul 13 09:30:40 2018 EDT
  Last Renewal Attempt: None
  Next Renewal Attempt: Jan 09 09:30:40 2019 EDT
  Registration Expires: Jul 13 09:25:31 2019 EDT

License Authorization:
  Status: AUTHORIZED on Jul 13 09:30:45 2018 EDT
  Last Communication Attempt: SUCCEEDED on Jul 13 09:30:45 2018 EDT
  Next Communication Attempt: Aug 12 09:30:45 2018 EDT
  Communication Deadline: Oct 11 09:25:40 2018 EDT

```

Additional References

Related Documents

Related Topic	Document Title
Cisco Smart Software Manager Help	Smart Software Manager Help
Cisco Smart Software Manager On-Prem	Cisco Smart Software Manager On-Prem

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>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.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for Smart Licensing

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

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

Table 7: Feature Information for Smart Licensing

Feature Name	Releases	Feature Information
Smart Licensing	Cisco IOS XE Fuji 16.9.1	Smart Licensing is a cloud-based, software license management solution that allows you to manage and track the status of your license, hardware and software usage trends.



CHAPTER 5

Configuring Application Visibility and Control in a Wired Network

- [Finding Feature Information, on page 125](#)
- [Information About Application Visibility and Control in a Wired Network, on page 125](#)
- [Supported AVC Class Map and Policy Map Formats, on page 126](#)
- [Restrictions for Wired Application Visibility and Control, on page 127](#)
- [How to Configure Application Visibility and Control, on page 129](#)
- [Monitoring Application Visibility and Control, on page 153](#)
- [Examples: Application Visibility and Control Configuration, on page 154](#)
- [Basic Troubleshooting - Questions and Answers, on page 163](#)
- [Additional References for Application Visibility and Control, on page 164](#)
- [Feature History for Application Visibility and Control in a Wired Network, on page 165](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and 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 feature information table at the end of this module.

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 Application Visibility and Control in a Wired Network

Application Visibility and Control (AVC) is a critical part of Cisco's efforts to evolve its Branch and Campus solutions from being strictly packet and connection based to being application-aware and application-intelligent. Application Visibility and Control (AVC) classifies applications using deep packet inspection techniques with the Network-Based Application Recognition (NBAR2) engine. AVC can be configured on wired access ports for standalone switches as well as for a switch stack. NBAR2 can be activated either explicitly on the interface by enabling protocol-discovery or implicitly by attaching a QoS policy that contains **match protocol** classifier.

Wired AVC Flexible NetFlow (FNF) can be configured on an interface to provide client, server and application statistics per interface. The record is similar to **application-client-server-stats** traffic monitor which is available in **application-statistics** and **application-performance** profiles in Easy Performance Monitor (Easy perf-mon or ezPM).

Supported AVC Class Map and Policy Map Formats

Supported AVC Class Map Format

Class Map Format	Class Map Example	Direction
match protocol <i>protocol name</i>	<code>class-map match-any NBAR-VOICE match protocol ms-lync-audio</code>	Both ingress and egress
Combination filters	<code>class-map match-any NBAR-VOICE match protocol ms-lync-audio match dscp ef</code>	Both ingress and egress

Supported AVC Policy Format

Policy Format	QoS Action
Egress policy based on match protocol filter	Mark and police
Ingress policy based on match protocol filter	Mark and police

The following table describes the detailed AVC policy format with an example:

AVC Policy Format	AVC Policy Example	Direction
Basic set	<code>policy-map MARKING-IN class NBAR-MM_CONFERENCING set dscp af41</code>	Ingress and egress
Basic police	<code>policy-map POLICING-IN class NBAR-MM_CONFERENCING police cir 600000 set dscp af41</code>	Ingress and egress
Basic set and police	<code>policy-map webex-policy class webex-class set dscp ef police 5000000</code>	Ingress and egress

AVC Policy Format	AVC Policy Example	Direction
Multiple set and police including default	<pre> policy-map webex-policy class webex-class set dscp af31 police 4000000 class class-webex-category set dscp ef police 6000000 class class-default set dscp <> </pre>	Ingress and egress
Hierarchical police	<pre> policy-map webex-policy class webex-class police 5000000 service-policy client-in-police-only policy-map client-in-police-only class webex-class police 100000 class class-webex-category set dscp ef police 200000 </pre>	Ingress and egress
Hierarchical set and police	<pre> policy-map webex-policy class class-default police 1500000 service policy client-up-child policy-map client-up-child class webex-class police 100000 set dscp ef class class-webex-category police 200000 set dscp af31 </pre>	

Restrictions for Wired Application Visibility and Control

- NBAR based QoS policy configuration is allowed only on wired physical ports. Policy configuration is not supported on virtual interfaces like VLAN and other logical interfaces.
- NBAR based QoS policy configuration is not supported on port-channel member ports and virtual interfaces like SVIs or sub-interfaces.
- NBAR based QoS policy configuration is supported on Layer 2 access and trunk ports and Layer 3 routed ports.
- NBAR and transmit (Tx) Switched Port Analyzer (SPAN) is not supported on the same interface.
- Only one of the NBAR based QoS mechanisms are allowed to be attached to any port at the same time, either protocol based or attributes based. Only the following two attributes are supported:
 - traffic-class
 - business-relevance

- The legacy WDAVC QoS limitations are still applicable:
 - Only marking and policing are supported.
 - Only physical interfaces are supported.
 - There is a delay in the QoS classification since the application classification is done offline (while the initial packet/s of the flow are meanwhile forwarded before the correct QoS classification).
- NBAR2 based match criteria **match protocol** will be allowed only with marking or policing actions. NBAR2 match criteria will not be allowed in a policy that has queuing features configured.
- ‘Match Protocol’: up to 255 concurrent different protocols in all policies (8 bits HW limitation).
- AVC is not supported on management port (Gig 0/0).
- IPv6 packet classification is not supported.
- Only IPv4 unicast(TCP/UDP) is supported.
- Web UI: You can configure application visibility and perform application monitoring from the Web UI. Application Control can only be done using the CLI. It is not supported on the Web UI.
To manage and check wired AVC traffic on the Web UI, you must first configure **ip http authentication local** and **ip nbar http-service** commands using the CLI.
- NBAR and ACL logging cannot be configured together on the same switch.
- Protocol-discovery, application-based QoS, and wired AVC FNF cannot be configured together at the same time on the same interface with the non-application-based FNF. However, these wired AVC features can be configured with each other. For example, protocol-discovery, application-based QoS and wired AVC FNF can be configured together on the same interface at the same time.
- AVC and Encrypted Traffic Analytics (ETA) cannot be configured together at the same time on the same interface.
- Attachment should be done only on physical Layer2 (Access/Trunk) and Layer3 ports. Uplink can be attached as long as it is a single uplink and is not part of a port channel.
- Performance: Each switch member is able to handle 500 connections per second (CPS) at less than 50% CPU utilization.
- Scale: Able to handle up to 10,000 bi-directional flows per 48 access ports and 5000 bi-directional flows per 24 access ports. (~200 flows per access port).
- Wired AVC allows only the fixed set of fields listed in the procedures of this chapter. Other combinations are not allowed. For a regular FNF flow monitor, other combinations are allowed (for the list of supported FNF fields, refer the "Configuring Flexible NetFlow" chapter of the *Network Management Configuration Guide*).
- For wired AVC traffic, four AVC flow monitors per direction, interface, and protocol (IPv4/6) are supported on the system.

How to Configure Application Visibility and Control

Configuring Application Visibility and Control in a Wired Network

To configure application visibility and control on wired ports, follow these steps:

Configuring Visibility :

- Activate NBAR2 engine by enabling protocol-discovery on the interface using the **ip nbar protocol-discovery** command in the interface configuration mode. See [Enabling Application Recognition on an interface, on page 129](#) .

Configuring Control : Configure QoS policies based on application by

1. Creating an AVC QoS policy. See [Creating AVC QoS Policy, on page 130](#) .
2. Applying AVC QoS policy to the interface. See [Applying a QoS Policy to the switch port, on page 132](#) .

Configuring application-based Flexible Netflow :

- Create a flow record by specifying key and non-key fields to the flow. See [Creating a Flow Record, on page 133](#) .
- Create a flow exporter to export the flow record. See [Creating a Flow Exporter, on page 145](#) .
- Create a flow monitor based on the flow record and the flow exporter. See [Creating a Flow Monitor, on page 146](#) .
- Attach the flow monitor to the interface. See [Associating Flow Monitor to an interface, on page 148](#) .

Protocol-Discovery, application-based QoS and application-based FNF are all independent features. They can be configured independently or together on the same interface at the same time.

Enabling Application Recognition on an interface

To enable application recognition on an interface, follow these steps:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip nbar protocol-discovery**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 2	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet 1/0/1</code>	Specifies the interface for which you are enabling protocol-discovery and enters interface configuration mode.
Step 3	ip nbar protocol-discovery Example: Device(config-if)# <code>ip nbar protocol-discovery</code>	Enables application recognition on the interface by activating NBAR2 engine.
Step 4	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.

Creating AVC QoS Policy

To create AVC QoS policy, perform these general steps:

1. Create a class map with match protocol filters.
2. Create a policy map.
3. Apply the policy map to the interface.

Creating a Class Map

You need to create a class map before configuring any match protocol filter. The QoS actions such as marking and policing can be applied to the traffic. The AVC match protocol filters are applied to the wired access ports. For more information about the protocols that are supported, see http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/qos_nbar/prot_lib/config_library/nbar-prot-pack-library.html.

SUMMARY STEPS

1. **configure terminal**
2. **class-map** *class-map-name*
3. **match protocol** *application-name*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	class-map <i>class-map-name</i> Example: Device(config)# class-map webex-class	Creates a class map.
Step 3	match protocol <i>application-name</i> Example: Device(config)# class-map webex-class Device(config-cmap)# match protocol webex-media	Specifies match to the application name.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

Creating a Policy Map

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** *policy-map-name*
3. **class** [*class-map-name* | **class-default**]
4. **police** *rate-bps burst-byte*
5. **set** { **dscp** *new-dscp* | **cos** *cos-value* }
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	policy-map <i>policy-map-name</i> Example: Device(config)# policy-map webex-policy	Creates a policy map by entering the policy map name, and enters policy-map configuration mode. By default, no policy maps are defined. The default behavior of a policy map is to set the DSCP to 0 if the packet is an IP packet and to set the CoS to 0 if the packet is tagged. No policing is performed.

	Command or Action	Purpose
		<p>Note To delete an existing policy map, use the no policy-map <i>policy-map-name</i> global configuration command.</p>
Step 3	<p>class [<i>class-map-name</i> class-default]</p> <p>Example:</p> <pre>Device(config-pmap) # class webex-class</pre>	<p>Defines a traffic classification, and enters policy-map class configuration mode.</p> <p>By default, no policy map and class maps are defined.</p> <p>If a traffic class has already been defined by using the class-map global configuration command, specify its name for <i>class-map-name</i> in this command.</p> <p>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 is included in the class-default class, all packets that have not already matched the other traffic classes will match class-default.</p> <p>Note To delete an existing class map, use the no class <i>class-map-name</i> policy-map configuration command.</p>
Step 4	<p>police <i>rate-bps</i> <i>burst-byte</i></p> <p>Example:</p> <pre>Device(config-pmap-c) # police 100000 80000</pre>	<p>Defines a policer for the classified traffic.</p> <p>By default, no policer is defined.</p> <ul style="list-style-type: none"> For <i>rate-bps</i>, specify an average traffic rate in bits per second (b/s). The range is 8000 to 10000000000. For <i>burst-byte</i>, specify the normal burst size in bytes. The range is 1000 to 512000000.
Step 5	<p>set { dscp <i>new-dscp</i> cos <i>cos-value</i> }</p> <p>Example:</p> <pre>Device(config-pmap-c) # set dscp 45</pre>	<p>Classifies IP traffic by setting a new value in the packet.</p> <ul style="list-style-type: none"> For dscp <i>new-dscp</i>, enter a new DSCP value to be assigned to the classified traffic. The range is 0 to 63.
Step 6	<p>end</p> <p>Example:</p> <pre>Device(config) # end</pre>	<p>Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.</p>

Applying a QoS Policy to the switch port

SUMMARY STEPS

1. **configure** terminal
2. **interface** *interface-id*
3. **service-policy input** *policymapname*

4. end

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface GigabitEthernet 1/0/1	Enters the interface configuration mode.
Step 3	service-policy input <i>polycymapname</i> Example: Device(config-if)# service-policy input MARKING_IN	Applies local policy to interface.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

Configuring Wired AVC Flexible Netflow

Creating a Flow Record

Wired AVC FNF supports two types of predefined flow records — Legacy Bidirectional flow records and Directional flow records (ingress and egress). A total of four different predefined flow records, two bidirectional flow records and two directional flow records, can be configured and associated with a flow monitor. The legacy bidirectional records are client/server application statistics records, and the new directional records are application-stats for input/output.

- [Bidirectional Flow Records, on page 133](#)
- [Directional Flow Records, on page 140](#)

Bidirectional Flow Records

Flow Record 1 - Bidirectional Flow Record

SUMMARY STEPS

1. **configure terminal**
2. **flow record** *flow_record_name*
3. **description** *description*
4. **match ipv4 version**
5. **match ipv4 protocol**
6. **match application name**

7. **match connection client ipv4 address**
8. **match connection server ipv4 address**
9. **match connection server transport port**
10. **match flow observation point**
11. **collect flow direction**
12. **collect connection initiator**
13. **collect connection new-connections**
14. **collect connection client counter packets long**
15. **collect connection client counter bytes network long**
16. **collect connection server counter packets long**
17. **collect connection server counter bytes network long**
18. **collect timestamp absolute first**
19. **collect timestamp absolute last**
20. **end**
21. **show flow record**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	flow record <i>flow_record_name</i> Example: Device(config)# <code>flow record fr-wdavic-1</code>	Enters flow record configuration mode.
Step 3	description <i>description</i> Example: Device(config-flow-record)# <code>description fr-wdavic-1</code>	(Optional) Creates a description for the flow record.
Step 4	match ipv4 version Example: Device (config-flow-record)# <code>match ipv4 version</code>	Specifies a match to the IP version from the IPv4 header.
Step 5	match ipv4 protocol Example: Device (config-flow-record)# <code>match ipv4 protocol</code>	Specifies a match to the IPv4 protocol.
Step 6	match application name Example: Device (config-flow-record)# <code>match application name</code>	Specifies a match to the application name. Note This action is mandatory for AVC support, as this allows the flow to be matched against the application.

	Command or Action	Purpose
Step 7	<p>match connection client ipv4 address</p> <p>Example:</p> <pre>Device (config-flow-record) # match connection client ipv4 address</pre>	Specifies a match to the IPv4 address of the client (flow initiator).
Step 8	<p>match connection server ipv4 address</p> <p>Example:</p> <pre>Device (config-flow-record) # match connection server ipv4 address</pre>	Specifies a match to the IPv4 address of the server (flow responder).
Step 9	<p>match connection server transport port</p> <p>Example:</p> <pre>Device (config-flow-record) # match connection server transport port</pre>	Specifies a match to the transport port of the server.
Step 10	<p>match flow observation point</p> <p>Example:</p> <pre>Device (config-flow-record) # match flow observation point</pre>	Specifies a match to the observation point ID for flow observation metrics.
Step 11	<p>collect flow direction</p> <p>Example:</p> <pre>Device (config-flow-record) # collect flow direction</pre>	<p>Specifies to collect the direction — Ingress or Egress — of the relevant side — Initiator or Responder — of the bi-directional flow that is specified by the initiator keyword in the collect connection initiator command in the step below. Depending on the value specified by the initiator keyword, the flow direction keyword takes the following values :</p> <ul style="list-style-type: none"> • 0x01 = Ingress Flow • 0x02 = Egress Flow <p>When the initiator keyword is set to initiator, the flow direction is specified from the initiator side of the flow. When the initiator keyword is set to responder, the flow direction is specified from the responder side of the flow. For wired AVC, the initiator keyword is always set to initiator.</p>
Step 12	<p>collect connection initiator</p> <p>Example:</p> <pre>Device (config-flow-record) # collect connection initiator</pre>	<p>Specifies to collect the side of the flow — Initiator or Responder — relevant to the direction of the flow specified by the collect flow direction command. The initiator keyword provides the following information about the direction of the flow :</p> <ul style="list-style-type: none"> • 0x01 = Initiator - the flow source is the initiator of the connection <p>For wired AVC, the initiator keyword is always set to initiator.</p>

	Command or Action	Purpose
Step 13	collect connection new-connections Example: Device (config-flow-record)# collect connection new-connections	Specifies to collect the number of connection initiations observed.
Step 14	collect connection client counter packets long Example: Device (config-flow-record)# collect connection client counter packets long	Specifies to collect the number of packets sent by the client.
Step 15	collect connection client counter bytes network long Example: Device (config-flow-record)# collect connection client counter bytes network long	Specifies to collect the total number of bytes transmitted by the client.
Step 16	collect connection server counter packets long Example: Device (config-flow-record)# collect connection server counter packets long	Specifies to collect the number of packets sent by the server.
Step 17	collect connection server counter bytes network long Example: Device (config-flow-record)# collect connection server counter bytes network long	Specifies to collect the total number of bytes transmitted by the server.
Step 18	collect timestamp absolute first Example: Device (config-flow-record)# collect timestamp absolute first	Specifies to collect the time, in milliseconds, when the first packet was seen in the flow.
Step 19	collect timestamp absolute last Example: Device (config-flow-record)# collect timestamp absolute last	Specifies to collect the time, in milliseconds, when the most recent packet was seen in the flow.
Step 20	end Example: Device (config)# end	Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.
Step 21	show flow record Example: Device # show flow record	Displays information about all the flow records.

Flow Record 2 - Bidirectional Flow Record

SUMMARY STEPS

1. **configure terminal**
2. **flow record** *flow_record_name*
3. **description** *description*
4. **match ipv4 version**
5. **match ipv4 protocol**
6. **match application name**
7. **match connection client ipv4 address**
8. **match connection client transport port**
9. **match connection server ipv4 address**
10. **match connection server transport port**
11. **match flow observation point**
12. **collect flow direction**
13. **collect connection initiator**
14. **collect connection new-connections**
15. **collect connection client counter packets long**
16. **collect connection client counter bytes network long**
17. **collect connection server counter packets long**
18. **collect connection server counter bytes network long**
19. **collect timestamp absolute first**
20. **collect timestamp absolute last**
21. **end**
22. **show flow record**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	flow record <i>flow_record_name</i> Example: Device(config)# flow record fr-wdavic-1	Enters flow record configuration mode.
Step 3	description <i>description</i> Example: Device(config-flow-record)# description fr-wdavic-1	(Optional) Creates a description for the flow record.
Step 4	match ipv4 version Example: Device (config-flow-record)# match ipv4 version	Specifies a match to the IP version from the IPv4 header.

	Command or Action	Purpose
Step 5	match ipv4 protocol Example: Device (config-flow-record)# match ipv4 protocol	Specifies a match to the IPv4 protocol.
Step 6	match application name Example: Device (config-flow-record)# match application name	Specifies a match to the application name. Note This action is mandatory for AVC support, as this allows the flow to be matched against the application.
Step 7	match connection client ipv4 address Example: Device (config-flow-record)# match connection client ipv4 address	Specifies a match to the IPv4 address of the client (flow initiator).
Step 8	match connection client transport port Example: Device (config-flow-record)# match connection client transport port	(Optional) Specifies a match to the connection port of the client as a key field for a flow record.
Step 9	match connection server ipv4 address Example: Device (config-flow-record)# match connection server ipv4 address	Specifies a match to the IPv4 address of the server (flow responder).
Step 10	match connection server transport port Example: Device (config-flow-record)# match connection server transport port	Specifies a match to the transport port of the server.
Step 11	match flow observation point Example: Device (config-flow-record)# match flow observation point	Specifies a match to the observation point ID for flow observation metrics.
Step 12	collect flow direction Example: Device (config-flow-record)# collect flow direction	Specifies to collect the direction — Ingress or Egress — of the relevant side — Initiator or Responder — of the bi-directional flow that is specified by the initiator keyword in the collect connection initiator command in the step below. Depending on the value specified by the initiator keyword, the flow direction keyword takes the following values : <ul style="list-style-type: none"> • 0x01 = Ingress Flow • 0x02 = Egress Flow When the initiator keyword is set to initiator, the flow direction is specified from the initiator side of the flow.

	Command or Action	Purpose
		When the initiator keyword is set to responder, the flow direction is specified from the responder side of the flow. For wired AVC, the initiator keyword is always set to initiator.
Step 13	collect connection initiator Example: Device (config-flow-record) # collect connection initiator	Specifies to collect the side of the flow — Initiator or Responder — relevant to the direction of the flow specified by the collect flow direction command. The initiator keyword provides the following information about the direction of the flow : <ul style="list-style-type: none"> • 0x01 = Initiator - the flow source is the initiator of the connection For wired AVC, the initiator keyword is always set to initiator.
Step 14	collect connection new-connections Example: Device (config-flow-record) # collect connection new-connections	Specifies to collect the number of connection initiations observed.
Step 15	collect connection client counter packets long Example: Device (config-flow-record) # collect connection client counter packets long	Specifies to collect the number of packets sent by the client.
Step 16	collect connection client counter bytes network long Example: Device (config-flow-record) # collect connection client counter bytes network long	Specifies to collect the total number of bytes transmitted by the client.
Step 17	collect connection server counter packets long Example: Device (config-flow-record) # collect connection server counter packets long	Specifies to collect the number of packets sent by the server.
Step 18	collect connection server counter bytes network long Example: Device (config-flow-record) # collect connection server counter bytes network long	Specifies to collect the total number of bytes transmitted by the server.
Step 19	collect timestamp absolute first Example: Device (config-flow-record) # collect timestamp absolute first	Specifies to collect the time, in milliseconds, when the first packet was seen in the flow.

	Command or Action	Purpose
Step 20	collect timestamp absolute last Example: Device (config-flow-record)# <code>collect timestamp absolute last</code>	Specifies to collect the time, in milliseconds, when the most recent packet was seen in the flow.
Step 21	end Example: Device (config)# <code>end</code>	Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.
Step 22	show flow record Example: Device # <code>show flow record</code>	Displays information about all the flow records.

Directional Flow Records

Flow Record 3 - Directional Flow Record - Ingress

SUMMARY STEPS

1. **configure terminal**
2. **flow record** *flow_record_name*
3. **description** *description*
4. **match ipv4 version**
5. **match ipv4 protocol**
6. **match ipv4 source address**
7. **match ipv4 destination address**
8. **match transport source-port**
9. **match transport destination-port**
10. **match interface input**
11. **match application name**
12. **collect interface output**
13. **collect counter bytes long**
14. **collect counter packets long**
15. **collect timestamp absolute first**
16. **collect timestamp absolute last**
17. **end**
18. **show flow record**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	flow record <i>flow_record_name</i> Example: Device(config)# flow record fr-wdavic-3	Enters flow record configuration mode.
Step 3	description <i>description</i> Example: Device(config-flow-record)# description flow-record-1	(Optional) Creates a description for the flow record.
Step 4	match ipv4 version Example: Device (config-flow-record)# match ipv4 version	Specifies a match to the IP version from the IPv4 header.
Step 5	match ipv4 protocol Example: Device (config-flow-record)# match ipv4 protocol	Specifies a match to the IPv4 protocol.
Step 6	match ipv4 source address Example: Device (config-flow-record)# match ipv4 source address	Specifies a match to the IPv4 source address as a key field.
Step 7	match ipv4 destination address Example: Device (config-flow-record)# match ipv4 destination address	Specifies a match to the IPv4 destination address as a key field.
Step 8	match transport source-port Example: Device (config-flow-record)# match transport source-port	Specifies a match to the transport source port as a key field.
Step 9	match transport destination-port Example: Device (config-flow-record)# match transport destination-port	Specifies a match to the transport destination port as a key field.
Step 10	match interface input Example: Device (config-flow-record)# match interface input	Specifies a match to the input interface as a key field.
Step 11	match application name Example: Device (config-flow-record)# match application name	Specifies a match to the application name. Note This action is mandatory for AVC support, as this allows the flow to be matched against the application.

	Command or Action	Purpose
Step 12	collect interface output Example: Device (config-flow-record)# collect interface output	Specifies to collect the output interface from the flows.
Step 13	collect counter bytes long Example: Device (config-flow-record)# collect counter bytes long	Specifies to collect the number of bytes in a flow.
Step 14	collect counter packets long Example: Device (config-flow-record)# collect counter packets long	Specifies to collect the number of packets in a flow.
Step 15	collect timestamp absolute first Example: Device (config-flow-record)# collect timestamp absolute first	Specifies to collect the time, in milliseconds, when the first packet was seen in the flow.
Step 16	collect timestamp absolute last Example: Device (config-flow-record)# collect timestamp absolute last	Specifies to collect the time, in milliseconds, when the most recent packet was seen in the flow.
Step 17	end Example: Device (config)# end	Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.
Step 18	show flow record Example: Device # show flow record	Displays information about all the flow records.

Flow Record 4 - Directional Flow Record - Egress

SUMMARY STEPS

1. **configure terminal**
2. **flow record** *flow_record_name*
3. **description** *description*
4. **match ipv4 version**
5. **match ipv4 protocol**
6. **match ipv4 source address**
7. **match ipv4 destination address**
8. **match transport source-port**

9. **match transport destination-port**
10. **match interface output**
11. **match application name**
12. **collect interface input**
13. **collect counter bytes long**
14. **collect counter packets long**
15. **collect timestamp absolute first**
16. **collect timestamp absolute last**
17. **end**
18. **show flow record**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	flow record <i>flow_record_name</i> Example: Device(config)# flow record fr-wdavic-4	Enters flow record configuration mode.
Step 3	description <i>description</i> Example: Device(config-flow-record)# description flow-record-1	(Optional) Creates a description for the flow record.
Step 4	match ipv4 version Example: Device (config-flow-record)# match ipv4 version	Specifies a match to the IP version from the IPv4 header.
Step 5	match ipv4 protocol Example: Device (config-flow-record)# match ipv4 protocol	Specifies a match to the IPv4 protocol.
Step 6	match ipv4 source address Example: Device (config-flow-record)# match ipv4 source address	Specifies a match to the IPv4 source address as a key field.
Step 7	match ipv4 destination address Example: Device (config-flow-record)# match ipv4 destination address	Specifies a match to the IPv4 destination address as a key field.

	Command or Action	Purpose
Step 8	match transport source-port Example: Device (config-flow-record)# match transport source-port	Specifies a match to the transport source port as a key field.
Step 9	match transport destination-port Example: Device (config-flow-record)# match transport destination-port	Specifies a match to the transport destination port as a key field.
Step 10	match interface output Example: Device (config-flow-record)# match interface output	Specifies a match to the output interface as a key field.
Step 11	match application name Example: Device (config-flow-record)# match application name	Specifies a match to the application name. Note This action is mandatory for AVC support, as this allows the flow to be matched against the application.
Step 12	collect interface input Example: Device (config-flow-record)# collect interface input	Specifies to collect the input interface from the flows.
Step 13	collect counter bytes long Example: Device (config-flow-record)# collect counter bytes long	Specifies to collect the number of bytes in a flow.
Step 14	collect counter packets long Example: Device (config-flow-record)# collect counter packets long	Specifies to collect the number of packets in a flow.
Step 15	collect timestamp absolute first Example: Device (config-flow-record)# collect timestamp absolute first	Specifies to collect the time, in milliseconds, when the first packet was seen in the flow.
Step 16	collect timestamp absolute last Example: Device (config-flow-record)# collect timestamp absolute last	Specifies to collect the time, in milliseconds, when the most recent packet was seen in the flow.

	Command or Action	Purpose
Step 17	end Example: Device(config)# end	Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.
Step 18	show flow record Example: Device # show flow record	Displays information about all the flow records.

Creating a Flow Exporter

You can create a flow exporter to define the export parameters for a flow.

SUMMARY STEPS

1. **configure terminal**
2. **flow exporter** *flow_exporter_name*
3. **description** *description*
4. **destination** { *hostname* | *ipv4-address* | *ipv6-address* }
5. **option application-table** [**timeout** *seconds*]
6. **end**
7. **show flow exporter**
8. **show flow exporter statistics**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	flow exporter <i>flow_exporter_name</i> Example: Device(config)# flow exporter flow-exporter-1	Enters flow exporter configuration mode.
Step 3	description <i>description</i> Example: Device(config-flow-exporter)# description flow-exporter-1	(Optional) Creates a description for the flow exporter.
Step 4	destination { <i>hostname</i> <i>ipv4-address</i> <i>ipv6-address</i> } Example: Device (config-flow-exporter)# destination 10.10.1.1	Specifies the hostname, IPv4 or IPv6 address of the system to which the exporter sends data.

	Command or Action	Purpose
Step 5	option application-table [timeout <i>seconds</i>] Example: Device (config-flow-exporter)# option application-table timeout 500	(Optional) Configures the application table option for the flow exporter. The timeout option configures the resend time in seconds for the flow exporter. The valid range is from 1 to 86400 seconds.
Step 6	end Example: Device(config)# end	Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.
Step 7	show flow exporter Example: Device # show flow exporter	Displays information about all the flow exporters.
Step 8	show flow exporter statistics Example: Device # show flow exporter statistics	Displays flow exporter statistics.

Creating a Flow Monitor

You can create a flow monitor and associate it with a flow record.

SUMMARY STEPS

1. **configure terminal**
2. **flow monitor** *monitor-name*
3. **description** *description*
4. **record** *record-name*
5. **exporter** *exporter-name*
6. **cache** { **entries** *number-of-entries* | **timeout** { **active** | **inactive** } | **type normal** }
7. **end**
8. **show flow monitor**
9. **show flow monitor** *flow-monitor-name*
10. **show flow monitor** *flow-monitor-name* **statistics**
11. **clear flow monitor** *flow-monitor-name* **statistics**
12. **show flow monitor** *flow-monitor-name* **cache format table**
13. **show flow monitor** *flow-monitor-name* **cache format record**
14. **show flow monitor** *flow-monitor-name* **cache format csv**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	flow monitor <i>monitor-name</i> Example: Device (config)# flow monitor flow-monitor-1	Creates a flow monitor and enters flow monitor configuration mode.
Step 3	description <i>description</i> Example: Device (config-flow-monitor)# description flow-monitor-1	(Optional) Creates a description for the flow monitor.
Step 4	record <i>record-name</i> Example: Device (config-flow-monitor)# record flow-record-1	Specifies the name of a record that was created previously.
Step 5	exporter <i>exporter-name</i> Example: Device (config-flow-monitor)# exporter flow-exporter-1	Specifies the name of an exporter that was created previously.
Step 6	cache { entries <i>number-of-entries</i> timeout { active inactive } type normal } Example: Device (config-flow-monitor)# cache timeout active 1800 Example: Device (config-flow-monitor)# cache timeout inactive 200 Example: Device (config-flow-monitor)# cache type normal	(Optional) Specifies to configure flow cache parameters. <ul style="list-style-type: none"> • entries <i>number-of-entries</i> — Specifies the maximum number of flow entries in the flow cache in the range from 16 to 65536. Note Only normal cache type is supported.
Step 7	end Example: Device(config)# end	Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.
Step 8	show flow monitor Example: Device # show flow monitor	Displays information about all the flow monitors.
Step 9	show flow monitor <i>flow-monitor-name</i> Example: Device # show flow monitor flow-monitor-1	Displays information about the specified wired AVC flow monitor.
Step 10	show flow monitor <i>flow-monitor-name</i> statistics Example: Device# show flow monitor flow-monitor-1 statistics	Displays statistics for wired AVC flow monitor.

Associating Flow Monitor to an interface

	Command or Action	Purpose
Step 11	clear flow monitor <i>flow-monitor-name</i> statistics Example: Device# clear flow monitor flow-monitor-1 statistics	Clears the statistics of the specified flow monitor. Use the show flow monitor flow-monitor-1 statistics command after using the clear flow monitor flow-monitor-1 statistics to verify that all the statistics have been reset.
Step 12	show flow monitor <i>flow-monitor-name</i> cache format table Example: Device# show flow monitor flow-monitor-1 cache format table	Displays flow cache contents in a tabular format.
Step 13	show flow monitor <i>flow-monitor-name</i> cache format record Example: Device# show flow monitor flow-monitor-1 cache format record	Displays flow cache contents in similar format as the flow record.
Step 14	show flow monitor <i>flow-monitor-name</i> cache format csv Example: Device# show flow monitor flow-monitor-1 cache format csv	Displays flow cache contents in CSV format.

Associating Flow Monitor to an interface

You can attach two different wired AVC monitors with different predefined records to an interface at the same time.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip flow monitor** *monitor-name* { **input** | **output** }
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface GigabitEthernet 1/0/1	Enters the interface configuration mode.

	Command or Action	Purpose
Step 3	ip flow monitor <i>monitor-name</i> { input output } Example: <pre>Device (config-if) # ip flow monitor flow-monitor-1 input</pre>	Associates a flow monitor to the interface for input and/or output packets.
Step 4	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

NBAR2 Custom Applications

NBAR2 supports the use of custom protocols to identify custom applications. Custom protocols support protocols and applications that NBAR2 does not currently support.

In every deployment, there are local and specific applications which are not covered by the NBAR2 protocol pack provided by Cisco. Local applications are mainly categorized as:

- Specific applications to an organization
- Applications specific to a geography

NBAR2 provides a way to manually customize such local applications. You can manually customize applications using the command **ip nbar custom** *myappname* in global configuration mode. Custom applications take precedence over built-in protocols. For each custom protocol, user can define a selector ID that can be used for reporting purposes.

There are various types of application customization:

Generic protocol customization

- HTTP
- SSL
- DNS

Composite : Customization based on multiple underlying protocols – **server-name**

Layer3/Layer4 customization

- IPv4 address
- DSCP values
- TCP/UDP ports
- Flow source or destination direction

Byte Offset : Customization based on specific byte values in the payload

HTTP Customization

HTTP customization could be based on a combination of HTTP fields from:

- **cookie** - HTTP Cookie
- **host** - Host name of Origin Server containing resource
- **method** - HTTP method
- **referrer** - Address the resource request was obtained from
- **url** - Uniform Resource Locator path
- **user-agent** - Software used by agent sending the request
- **version** - HTTP version
- **via** - HTTP via field

HTTP Customization

Custom application called MYHTTP using the HTTP host “*mydomain.com” with Selector ID 10.

```
Device# configure terminal
Device(config)# ip nbar custom MYHTTP http host *mydomain.com id 10
```

SSL Customization

Customization can be done for SSL encrypted traffic using information extracted from the SSL Server Name Indication (SNI) or Common Name (CN).

SSL Customization

Custom application called MYSSL using SSL unique-name “mydomain.com” with selector ID 11.

```
Device# configure terminal
Device(config)# ip nbar custom MYSSL ssl unique-name *mydomain.com id 11
```

DNS Customization

NBAR2 examines DNS request and response traffic, and can correlate the DNS response to an application. The IP address returned from the DNS response is cached and used for later packet flows associated with that specific application.

The command **ip nbar custom application-name dns domain-name id application-id** is used for DNS customization. To extend an existing application, use the command **ip nbar custom application-name dns domain-name domain-name extends existing-application**.

For more information on DNS based customization, see http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/qos_nbar/configuration/xs-3s/asr1000/qos-nbar-xe-3s-asr-1000-book/nbar-custapp-dns-xe.html.

DNS Customization

Custom application called MYDNS using the DNS domain name “mydomain.com” with selector ID 12.

```
Device# configure terminal
Device(config)# ip nbar custom MYDNS dns domain-name *mydomain.com id 12
```

Composite Customization

NBAR2 provides a way to customize applications based on domain names appearing in HTTP, SSL or DNS.

Composite Customization

Custom application called MYDOMAIN using HTTP, SSL or DNS domain name “mydomain.com” with selector ID 13.

```
Device# configure terminal
Device(config)# ip nbar custom MYDOMAIN composite server-name *mydomain.com id 13
```

L3/L4 Customization

Layer3/Layer4 customization is based on the packet tuple and is always matched on the first packet of a flow.

L3/L4 Customization

Custom application called LAYER4CUSTOM matching IP addresses 10.56.1.10 and 10.56.1.11, TCP and DSCP ef with selector ID 14.

```
Device# configure terminal
Device(config)# ip nbar custom LAYER4CUSTOM transport tcp id 14
Device(config-custom)# ip address 10.56.1.10 10.56.1.11
Device(config-custom)# dscp ef
```

Examples: Monitoring Custom Applications

Show Commands for Monitoring Custom Applications

show ip nbar protocol-id | inc Custom

```
Device# show ip nbar protocol-id | inc Custom
LAYER4CUSTOM          14          Custom
MYDNS                 12          Custom
MYDOMAIN              13          Custom
MYHTTP                10          Custom
MYSSL                 11          Custom
```

show ip nbar protocol-discovery protocol CUSTOM_APP

```
WSW-157# show ip nbar protocol-id MYSSL
Protocol Name          id          type
-----
MYSSL                  11          Custom
```

NBAR2 Dynamic Hitless Protocol Pack Upgrade

Protocol packs are software packages that update the NBAR2 protocol support on a device without replacing the Cisco software on the device. A protocol pack contains information on applications officially supported by NBAR2 which are compiled and packed together. For each application, the protocol-pack includes information on application signatures and application attributes. Each software release has a built-in protocol-pack bundled with it.

Protocol packs provide the following features:

- They are easy and fast to load.
- They are easy to upgrade to a higher version protocol pack or revert to a lower version protocol pack.
- They do not require the switch to be reloaded.

NBAR2 protocol packs are available for download on Cisco Software Center from this URL:
<https://software.cisco.com/download/navigator.html> .

Prerequisites for the NBAR2 Protocol Pack

Before loading a new protocol pack, you must copy the protocol pack to the flash on all the switch members.

To load a protocol pack, see [Loading the NBAR2 Protocol Pack, on page 152](#) .

Loading the NBAR2 Protocol Pack

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip nbar protocol-pack** *protocol-pack* [**force**]
4. **exit**
5. **show ip nbar protocol-pack** {protocol-pack | **active**} [**detail**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip nbar protocol-pack <i>protocol-pack</i> [force] Example: Device(config)# ip nbar protocol-pack flash:defProtoPack Example: Device(config)# default ip nbar protocol-pack	Loads the protocol pack. <ul style="list-style-type: none"> • Use the force keyword to specify and load a protocol pack of a lower version, which is different from the base protocol pack version. This also removes the configuration that is not supported by the current protocol pack on the switch. For reverting to the built-in protocol pack, use the following command:
Step 4	exit Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config)# exit	
Step 5	show ip nbar protocol-pack {protocol-pack active } [detail] Example: Device# show ip nbar protocol-pack active	Displays the protocol pack information. <ul style="list-style-type: none"> • Verify the loaded protocol pack version, publisher, and other details using this command. • Use the <i>protocol-pack</i> argument to display information about the specified protocol pack. • Use the active keyword to display active protocol pack information. • Use the detail keyword to display detailed protocol pack information.

Examples: Loading the NBAR2 Protocol Pack

The following example shows how to load a new protocol pack:

```
Device> enable
Device# configure terminal
Device(config)# ip nbar protocol-pack flash:newDefProtoPack
Device(config)# exit
```

The following example shows how to use the **force** keyword to load a protocol pack of a lower version:

```
Device> enable
Device# configure terminal
Device(config)# ip nbar protocol-pack flash:OldDefProtoPack force
Device(config)# exit
```

The following example shows how to revert to the built-in protocol pack:

```
Device> enable
Device# configure terminal
Device(config)# default ip nbar protocol-pack
Device(config)# exit
```

Monitoring Application Visibility and Control

This section describes the new commands for application visibility.

The following commands can be used to monitor application visibility on the and access ports.

Table 8: Monitoring Application Visibility Commands on the

Command	Purpose
---------	---------

<pre>show ip nbar protocol-discovery [interface <i>interface-type interface-number</i>] [stats{byte-count bit-rate packet-count max-bit-rate}] [protocol <i>protocol-name</i> top-n <i>number</i>]</pre>	<p>Displays the statistics gathered by the NBAR Protocol Discovery feature.</p> <ul style="list-style-type: none"> (Optional) Enter keywords and arguments to fine-tune the statistics displayed. For more information on each of the keywords, refer to the show ip nbar protocol-discovery command in Cisco IOS Quality of Service Solutions Command Reference.
<pre>show policy-map interface <i>interface-type</i> <i>interface-number</i></pre>	<p>Displays information about policy map applied to the interface.</p>
<pre>show platform software fed switch <i>switch id</i> wdavc flows</pre>	<p>Displays statistics about all flows on the specified switch.</p>

Examples: Application Visibility and Control Configuration

This example shows how to create class maps with apply match protocol filters for application name:

```
Device# configure terminal
Device(config)# class-map match-any NBAR-VOICE
Device(config-cmap)# match protocol ms-lync-audio
Device(config-cmap)#end
```

This example shows how to create policy maps and define existing class maps for egress QoS:

```
Device# configure terminal
Device(config)# policy-map test-avc-up
Device(config-pmap)# class cat-browsing
Device(config-pmap-c)# police 150000
Device(config-pmap-c)# set dscp 12
Device(config-pmap-c)#end
```

This example shows how to create policy maps and define existing class maps for ingress QoS:

```
Device# configure terminal
Device(config)# policy-map test-avc-down
Device(config-pmap)# class cat-browsing
Device(config-pmap-c)# police 200000
Device(config-pmap-c)# set dscp 10
Device(config-pmap-c)#end
```

This example shows how to apply policy maps to a switch port:

```
Device# configure terminal
Device(config)# interface GigabitEthernet 1/0/1
Device(config-if)# switchport mode access
Device(config-if)# switchport access vlan 20
Device(config-if)# service-policy input POLICING_IN
Device(config-if)#end
```

Show Commands for Viewing the Configuration

show ip nbar protocol-discovery

Displays a report of the Protocol Discovery statistics per interface.

The following is a sample output for the statistics per interface:


```
Deviceqos-cat9k-reg2-r1# show ip nbar protocol-discovery int GigabitEthernet1/0/1
```

```
GigabitEthernet1/0/1
```

```
Last clearing of "show ip nbar protocol-discovery" counters 00:03:16
```

Output	Input
-----	-----
Protocol	Packet Count
Packet Count	Byte Count
Byte Count	30sec Bit Rate (bps)
30sec Bit Rate (bps)	30sec Max Bit Rate (bps)
30sec Max Bit Rate (bps)	
-----	-----
-----	-----
ms-lync	60580
55911	31174777
28774864	3613000
93000	3613000
3437000	
Total	60580
55911	31174777
28774864	3613000
93000	3613000
3437000	

show policy-map interface

Displays the QoS statistics and the configured policy maps on all interfaces.

The following is a sample output for the policy-maps configured on all the interfaces:

```
Deviceqos-cat9k-reg2-r1# show policy-map int
```

```
GigabitEthernet1/0/1
```

```
Service-policy input: MARKING-IN
```

```
Class-map: NBAR-VOICE (match-any)
  718 packets
  Match: protocol ms-lync-audio
    0 packets, 0 bytes
    30 second rate 0 bps
  QoS Set
```

```

dscp ef

Class-map: NBAR-MM_CONFERENCING (match-any)
  6451 packets
  Match: protocol ms-lync
    0 packets, 0 bytes
    30 second rate 0 bps
  Match: protocol ms-lync-video
    0 packets, 0 bytes
    30 second rate 0 bps
  QoS Set
    dscp af41

Class-map: class-default (match-any)
  34 packets
  Match: any

```

Show Commands for Viewing Flow Monitor Configuration

show flow monitor wdavc

Displays information about the specified wired AVC flow monitor.

```
Device # show flow monitor wdavc
```

```
Flow Monitor wdavc:
  Description:      User defined
  Flow Record:     wdavc
  Flow Exporter:   wdavc-exp (inactive)
  Cache:
    Type:          normal (Platform cache)
    Status:        not allocated
    Size:          12000 entries
    Inactive Timeout: 15 secs
    Active Timeout: 1800 secs

```

show flow monitor wdavc statistics

Displays statistics for wired AVC flow monitor.

```
Device# show flow monitor wdavc statistics
Cache type:          Normal (Platform cache)
Cache size:         12000
Current entries:    13

Flows added:        26
Flows aged:         13
- Active timeout    ( 1800 secs) 1
- Inactive timeout  (   15 secs) 12

```

clear flow monitor wdavc statistics

Clears the statistics of the specified flow monitor. Use the **show flow monitor wdavc statistics** command after using the **clear flow monitor wdavc statistics** to verify that all the statistics have been reset. The following is a sample output of the **show flow monitor wdavc statistics** command after clearing flow monitor statistics.

```
Device# show flow monitor wдавc statistics
Cache type: Normal (Platform cache)
Cache size: 12000
Current entries: 0

Flows added: 0
Flows aged: 0
```

Show Commands for Viewing Cache Contents

show flow monitor wдавc cache format table

Displays flow cache contents in a tabular format.

```
Device# show flow monitor wдавc cache format table
Cache type: Normal (Platform cache)
Cache size: 12000
Current entries: 13

Flows added: 26
Flows aged: 13
- Active timeout ( 1800 secs) 1
- Inactive timeout ( 15 secs) 12
```

CONN	IPV4 INITIATOR ADDR	CONN	IPV4 RESPONDER ADDR	CONN	RESPONDER	PORT
FLOW	OBSPOINT ID	IP	VERSION	IP	PROT	APP NAME
dirn	flow
64.103.125.147	4294967305	144.254.71.184	4	17	port dns	53
.....	Input
64.103.121.103	4294967305	10.1.1.2	4	17	layer7 dhcp	67
....contd.....	Input
64.103.125.3	4294967305	64.103.125.97	4	17	layer7 dhcp	68
.....	Input
10.0.2.6	4294967305	157.55.40.149	4	6	layer7 ms-lync	443
.....	Input
64.103.126.28	4294967305	66.163.36.139	4	6	layer7 cisco-jabber-im	443
....contd.....	Input
64.103.125.2	4294967305	64.103.125.29	4	17	layer7 dhcp	68
.....	Input
64.103.125.97	4294967305	64.103.101.181	4	17	layer7 dhcp	67
.....	Input
192.168.100.6	4294967305	10.10.20.1	4	17	layer7 cisco-jabber-control	5060
.....	Input

```

.....contd.....
64.103.125.3          64.103.125.29          68
      4294967305      4      17 layer7 dhcp      Input
.....
10.80.101.18         10.80.101.6           5060
      4294967305      4      6 layer7 cisco-collab-control Input
.....
10.1.11.4            66.102.11.99         80
      4294967305      4      6 layer7 google-services Input
.....contd.....
64.103.125.2         64.103.125.97        68
      4294967305      4      17 layer7 dhcp      Input
.....
64.103.125.29        64.103.101.181       67
      4294967305      4      17 layer7 dhcp      Input
.....

```

show flow monitor wdavc cache format record

Displays flow cache contents in similar format as the flow record.

```

Device# show flow monitor wdavc cache format record
Cache type:                Normal (Platform cache)
Cache size:                 12000
Current entries:           13

Flows added:                26
Flows aged:                 13
  - Active timeout         ( 1800 secs)  1
  - Inactive timeout       (   15 secs)  12

CONNECTION IPV4 INITIATOR ADDRESS: 64.103.125.147
CONNECTION IPV4 RESPONDER ADDRESS: 144.254.71.184
CONNECTION RESPONDER PORT:        53
FLOW OBSPOINT ID:                4294967305
IP VERSION:                       4
IP PROTOCOL:                       17
APPLICATION NAME:                 port dns
flow direction:                   Input
timestamp abs first:              08:55:46.917
timestamp abs last:               08:55:46.917
connection initiator:              Initiator
connection count new:              2
connection server packets counter: 1
connection client packets counter: 1
connection server network bytes counter: 190
connection client network bytes counter: 106

CONNECTION IPV4 INITIATOR ADDRESS: 64.103.121.103
CONNECTION IPV4 RESPONDER ADDRESS: 10.1.1.2
CONNECTION RESPONDER PORT:        67
FLOW OBSPOINT ID:                4294967305
IP VERSION:                       4

```

```
IP PROTOCOL: 17
APPLICATION NAME: layer7 dhcp
flow direction: Input
timestamp abs first: 08:55:47.917
timestamp abs last: 08:55:47.917
connection initiator: Initiator
connection count new: 1
connection server packets counter: 0
connection client packets counter: 1
connection server network bytes counter: 0
connection client network bytes counter: 350

CONNECTION IPV4 INITIATOR ADDRESS: 64.103.125.3
CONNECTION IPV4 RESPONDER ADDRESS: 64.103.125.97
CONNECTION RESPONDER PORT: 68
FLOW OBSPOINT ID: 4294967305
IP VERSION: 4
IP PROTOCOL: 17
APPLICATION NAME: layer7 dhcp
flow direction: Input
timestamp abs first: 08:55:47.917
timestamp abs last: 08:55:53.917
connection initiator: Initiator
connection count new: 1
connection server packets counter: 0
connection client packets counter: 4
connection server network bytes counter: 0
connection client network bytes counter: 1412

CONNECTION IPV4 INITIATOR ADDRESS: 10.0.2.6
CONNECTION IPV4 RESPONDER ADDRESS: 157.55.40.149
CONNECTION RESPONDER PORT: 443
FLOW OBSPOINT ID: 4294967305
IP VERSION: 4
IP PROTOCOL: 6
APPLICATION NAME: layer7 ms-lync
flow direction: Input
timestamp abs first: 08:55:46.917
timestamp abs last: 08:55:46.917
connection initiator: Initiator
connection count new: 2
connection server packets counter: 10
connection client packets counter: 14
connection server network bytes counter: 6490
connection client network bytes counter: 1639

CONNECTION IPV4 INITIATOR ADDRESS: 64.103.126.28
CONNECTION IPV4 RESPONDER ADDRESS: 66.163.36.139
CONNECTION RESPONDER PORT: 443
FLOW OBSPOINT ID: 4294967305
IP VERSION: 4
```

```

IP PROTOCOL: 6
APPLICATION NAME: layer7 cisco-jabber-im
flow direction: Input
timestamp abs first: 08:55:46.917
timestamp abs last: 08:55:46.917
connection initiator: Initiator
connection count new: 2
connection server packets counter: 12
connection client packets counter: 10
connection server network bytes counter: 5871
connection client network bytes counter: 2088

```

```

CONNECTION IPV4 INITIATOR ADDRESS: 64.103.125.2
CONNECTION IPV4 RESPONDER ADDRESS: 64.103.125.29
CONNECTION RESPONDER PORT: 68
FLOW OBSPOINT ID: 4294967305
IP VERSION: 4
IP PROTOCOL: 17
APPLICATION NAME: layer7 dhcp
flow direction: Input
timestamp abs first: 08:55:47.917
timestamp abs last: 08:55:47.917
connection initiator: Initiator
connection count new: 1
connection server packets counter: 0
connection client packets counter: 2
connection server network bytes counter: 0
connection client network bytes counter: 712

```

```

CONNECTION IPV4 INITIATOR ADDRESS: 64.103.125.97
CONNECTION IPV4 RESPONDER ADDRESS: 64.103.101.181
CONNECTION RESPONDER PORT: 67
FLOW OBSPOINT ID: 4294967305
IP VERSION: 4
IP PROTOCOL: 17
APPLICATION NAME: layer7 dhcp
flow direction: Input
timestamp abs first: 08:55:47.917
timestamp abs last: 08:55:47.917
connection initiator: Initiator
connection count new: 1
connection server packets counter: 0
connection client packets counter: 1
connection server network bytes counter: 0
connection client network bytes counter: 350

```

```

CONNECTION IPV4 INITIATOR ADDRESS: 192.168.100.6
CONNECTION IPV4 RESPONDER ADDRESS: 10.10.20.1
CONNECTION RESPONDER PORT: 5060
FLOW OBSPOINT ID: 4294967305
IP VERSION: 4

```

```

IP PROTOCOL: 17
APPLICATION NAME: layer7 cisco-jabber-control
flow direction: Input
timestamp abs first: 08:55:46.917
timestamp abs last: 08:55:46.917
connection initiator: Initiator
connection count new: 1
connection server packets counter: 0
connection client packets counter: 2
connection server network bytes counter: 0
connection client network bytes counter: 2046

CONNECTION IPV4 INITIATOR ADDRESS: 64.103.125.3
CONNECTION IPV4 RESPONDER ADDRESS: 64.103.125.29
CONNECTION RESPONDER PORT: 68
FLOW OBSPOINT ID: 4294967305
IP VERSION: 4
IP PROTOCOL: 17
APPLICATION NAME: layer7 dhcp
flow direction: Input
timestamp abs first: 08:55:47.917
timestamp abs last: 08:55:47.917
connection initiator: Initiator
connection count new: 1
connection server packets counter: 0
connection client packets counter: 2
connection server network bytes counter: 0
connection client network bytes counter: 712

CONNECTION IPV4 INITIATOR ADDRESS: 10.80.101.18
CONNECTION IPV4 RESPONDER ADDRESS: 10.80.101.6
CONNECTION RESPONDER PORT: 5060
FLOW OBSPOINT ID: 4294967305
IP VERSION: 4
IP PROTOCOL: 6
APPLICATION NAME: layer7 cisco-collab-control
flow direction: Input
timestamp abs first: 08:55:46.917
timestamp abs last: 08:55:47.917
connection initiator: Initiator
connection count new: 2
connection server packets counter: 23
connection client packets counter: 27
connection server network bytes counter: 12752
connection client network bytes counter: 8773

CONNECTION IPV4 INITIATOR ADDRESS: 10.1.11.4
CONNECTION IPV4 RESPONDER ADDRESS: 66.102.11.99
CONNECTION RESPONDER PORT: 80
FLOW OBSPOINT ID: 4294967305
IP VERSION: 4

```

```

IP PROTOCOL: 6
APPLICATION NAME: layer7 google-services
flow direction: Input
timestamp abs first: 08:55:46.917
timestamp abs last: 08:55:46.917
connection initiator: Initiator
connection count new: 2
connection server packets counter: 3
connection client packets counter: 5
connection server network bytes counter: 1733
connection client network bytes counter: 663

```

```

CONNECTION IPV4 INITIATOR ADDRESS: 64.103.125.2
CONNECTION IPV4 RESPONDER ADDRESS: 64.103.125.97
CONNECTION RESPONDER PORT: 68
FLOW OBSPOINT ID: 4294967305
IP VERSION: 4
IP PROTOCOL: 17
APPLICATION NAME: layer7 dhcp
flow direction: Input
timestamp abs first: 08:55:47.917
timestamp abs last: 08:55:53.917
connection initiator: Initiator
connection count new: 1
connection server packets counter: 0
connection client packets counter: 4
connection server network bytes counter: 0
connection client network bytes counter: 1412

```

```

CONNECTION IPV4 INITIATOR ADDRESS: 64.103.125.29
CONNECTION IPV4 RESPONDER ADDRESS: 64.103.101.181
CONNECTION RESPONDER PORT: 67
FLOW OBSPOINT ID: 4294967305
IP VERSION: 4
IP PROTOCOL: 17
APPLICATION NAME: layer7 dhcp
flow direction: Input
timestamp abs first: 08:55:47.917
timestamp abs last: 08:55:47.917
connection initiator: Initiator
connection count new: 1
connection server packets counter: 0
connection client packets counter: 1
connection server network bytes counter: 0
connection client network bytes counter: 350

```

show flow monitor wdavc cache format csv

Displays flow cache contents in CSV format.

```

Device# show flow monitor wdavc cache format csv
Cache type: Normal (Platform cache)
Cache size: 12000

```



```

Current entries:                               13

Flows added:                                  26
Flows aged:                                   13
  - Active timeout      ( 1800 secs)         1
  - Inactive timeout    (   15 secs)         12

CONN IPV4 INITIATOR ADDR,CONN IPV4 RESPONDER ADDR,CONN RESPONDER PORT,FLOW
OBSPOINT ID,IP VERSION,IP
PROT,APP NAME,flow dirn,time abs first,time abs last,conn initiator,conn
count new,conn server packets
cnt,conn client packets cnt,conn server network bytes cnt,conn client
network bytes cnt
64.103.125.147,144.254.71.184,53,4294967305,4,17,port
dns,Input,08:55:46.917,08:55:46.917,Initiator,2,1,1,190,106
64.103.121.103,10.1.1.2,67,4294967305,4,17,layer7
dhcp,Input,08:55:47.917,08:55:47.917,Initiator,1,0,1,0,350
64.103.125.3,64.103.125.97,68,4294967305,4,17,layer7
dhcp,Input,08:55:47.917,08:55:53.917,Initiator,1,0,4,0,1412
10.0.2.6,157.55.40.149,443,4294967305,4,6,layer7 ms-
lync,Input,08:55:46.917,08:55:46.917,Initiator,2,10,14,6490,1639
64.103.126.28,66.163.36.139,443,4294967305,4,6,layer7 cisco-jabber-
im,Input,08:55:46.917,08:55:46.917,Initiator,2,12,10,5871,2088
64.103.125.2,64.103.125.29,68,4294967305,4,17,layer7
dhcp,Input,08:55:47.917,08:55:47.917,Initiator,1,0,2,0,712
64.103.125.97,64.103.101.181,67,4294967305,4,17,layer7
dhcp,Input,08:55:47.917,08:55:47.917,Initiator,1,0,1,0,350
192.168.100.6,10.10.20.1,5060,4294967305,4,17,layer7 cisco-jabber-
control,Input,08:55:46.917,08:55:46.917,Initiator,1,0,2,0,2046
64.103.125.3,64.103.125.29,68,4294967305,4,17,layer7
dhcp,Input,08:55:47.917,08:55:47.917,Initiator,1,0,2,0,712
10.80.101.18,10.80.101.6,5060,4294967305,4,6,layer7 cisco-collab-
control,Input,08:55:46.917,08:55:47.917,Initiator,2,23,27,12752,8773
10.1.11.4,66.102.11.99,80,4294967305,4,6,layer7 google-
services,Input,08:55:46.917,08:55:46.917,Initiator,2,3,5,1733,663
64.103.125.2,64.103.125.97,68,4294967305,4,17,layer7
dhcp,Input,08:55:47.917,08:55:53.917,Initiator,1,0,4,0,1412
64.103.125.29,64.103.101.181,67,4294967305,4,17,layer7
dhcp,Input,08:55:47.917,08:55:47.917,Initiator,1,0,1,0,350

```

Basic Troubleshooting - Questions and Answers

Following are the basic questions and answers for troubleshooting wired Application Visibility and Control:

- Question:** My IPv6 traffic is not being classified.

Answer: Currently only IPv4 traffic is supported.
- Question:** My multicast traffic is not being classified

Answer: Currently only unicast traffic is supported

3. **Question:** I send ping but I don't see them being classified
Answer: Only TCP/UDP protocols are supported
4. **Question:** Why can't I attach NBAR to an SVI?
Answer: NBAR is only supported on physical interfaces.
5. **Question:** I see that most of my traffic is CAPWAP traffic, why?
Answer: Make sure that you have enabled NBAR on an access port that is not connected to a wireless access port. All traffic coming from AP's will be classified as capwap. Actual classification in this case happens either on the AP or WLC.
6. **Question:** In protocol-discovery, I see traffic only on one side. Along with that, there are a lot of unknown traffic.
Answer: This usually indicates that NBAR sees asymmetric traffic: one side of the traffic is classified in one switch member and the other on a different member. The recommendation is to attach NBAR only on access ports where we see both sides of the traffic. If you have multiple uplinks, you can't attach NBAR on them due to this issue. Similar issue happens if you configure NBAR on an interface that is part of a port channel.
7. **Question:** With protocol-discovery, I see an aggregate view of all application. How can I see traffic distribution over time?
Answer: WebUI will give you view of traffic over time for the last 48 hours.
8. **Question:** I can't configure queue-based egress policy with **match protocol protocol-name** command.
Answer: Only **shape** and **set DSCP** are supported in a policy with NBAR2 based classifiers. Common practice is to set DSCP on ingress and perform shaping on egress based on DSCP.
9. **Question:** I don't have NBAR2 attached to any interface but I still see that NBAR2 is activated.
Answer: If you have any class-map with **match protocol protocol-name**, NBAR will be globally activated on the stack but no traffic will be subjected to NBAR classification. This is an expected behavior and it does not consume any resources.
10. **Question:** I see some traffic under the default QoS queue. Why?
Answer: For each new flow, it takes a few packets to classify it and install the result in the hardware. During this time, the classification would be 'un-known' and traffic will fall under the default queue.

Additional References for Application Visibility and Control

Related Documents

Related Topic	Document Title
QoS	<i>NBAR Configuration Guide, Cisco IOS XE Release 16.x</i>
NBAR2 Protocol Pack Hitless Upgrade	<i>NBAR Configuration Guide, Cisco IOS XE Release 16.x</i>

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>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.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	<p>http://www.cisco.com/support</p>

Feature History for Application Visibility and Control in a Wired Network

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Everest 16.5.1a	Application Visibility and Control in a Wired Network	AVC is a critical part of Cisco's efforts to evolve its Branch and Campus solutions from being strictly packet and connection based to being application-aware and application-intelligent.
Cisco IOS XE Fuji 16.8.1a	Wired Application Visibility and Control (Wired AVC) Attribute-based QoS (EasyQoS)	Support for defining QoS classes and policies based on Network-Based Application Recognition (NBAR) attributes instead of specific protocols, was made available, with a few limitations. Only business-relevance and traffic-class are the supported NBAR attributes.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 6

Configuring SDM Templates

- [SDM Templates, on page 167](#)
- [SDM Templates and Switch Stacks, on page 167](#)
- [How to Configure SDM Templates, on page 168](#)
- [Monitoring and Maintaining SDM Templates, on page 169](#)
- [Configuration Examples for SDM Templates, on page 169](#)
- [Additional References for SDM Templates, on page 170](#)
- [Feature History for SDM Templates, on page 170](#)

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.

Cisco Catalyst 9300 Series Switches support Access template.

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.

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.

You can use the **show switch** privileged EXEC command to see if any stack members are in SDM mismatch mode.

How to Configure SDM Templates

Setting the SDM Template

Follow these steps to use the SDM template to maximize feature usage:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `sdm prefer access`
4. `end`
5. `reload`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	sdm prefer access Example: Device(config)# <code>sdm prefer access</code>	Sets the switch to the access template.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	reload Example: Device# <code>reload</code>	Reloads the operating system. After the system reboots, you can use the <code>show sdm prefer</code> privileged EXEC command to verify the new template configuration. If you enter the <code>show sdm prefer</code> command before you enter the reload privileged EXEC command, the <code>show sdm prefer</code> command shows the template currently

Command or Action	Purpose
	in use and the template that will become active after a reload.

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.



Note The SDM templates contain only those commands that are defined as part of the templates. If a template enables another related command that is not defined in the template, then this other command will be visible when the **show running config** command is entered. For example, if the SDM template enables the **switchport voice vlan** command, then the **spanning-tree portfast edge** command may also be enabled (although it is not defined on the SDM template).

If the SDM template is removed, then other such related commands are also removed and have to be reconfigured explicitly.

Configuration Examples for SDM Templates

Examples: Displaying SDM Templates

The following example output shows the Access template information on Cisco Catalyst 9300 Series Switches:

```
Device# show sdm prefer access
This is the Access template.
Number of VLANs: 4094
Unicast MAC addresses: 32768
Overflow Unicast MAC addresses: 1024
L2 Multicast entries: 8192
Overflow L2 Multicast entries: 512
L3 Multicast entries: 8192
Overflow L3 Multicast entries: 512
Directly connected routes: 24576
Indirect routes: 8192
STP Instances: 1024
Security Access Control Entries: 5120
QoS Access Control Entries: 5120
Policy Based Routing ACEs: 1024
Netflow Input ACEs: 256
Netflow Output ACEs: 768
Ingress Netflow ACEs: 256
Egress Netflow ACEs: 768
Flow SPAN ACEs: 1024
Tunnels: 512
```

```

LISP Instance Mapping Entries:          512
Control Plane Entries:                  512
Input Netflow flows:                   32768
Output Netflow flows:                  32768
SGT/DGT (or) MPLS VPN entries:         8192
SGT/DGT (or) MPLS VPN Overflow entries: 512
Wired clients:                          2048
MACSec SPD Entries:                    256
MPLS L3 VPN VRF:                       255
MPLS Labels:                           2048
MPLS L3 VPN Routes VRF Mode:           7168
MPLS L3 VPN Routes Prefix Mode:        3072
MVPN MDT Tunnels:                      256
L2 VPN EOMPLS Attachment Circuit:      256
MAX VPLS Bridge Domains :              128
MAX VPLS Peers Per Bridge Domain:      32
MAX VPLS/VPWS Pseudowires :           1024

```

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.
* values can be modified by sdm cli.

Examples: Configuring SDM Templates

```

Device(config)# sdm prefer access
Device(config)# exit
Device# reload
Proceed with reload? [confirm]

```

Additional References for SDM Templates

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	<i>Command Reference (Catalyst 9300 Series Switches)</i>

Feature History for SDM Templates

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Everest 16.5.1a	SDM Template	Standard SDM templates can be used to configure system resources to optimize support for specific features.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 7

Configuring System Message Logs

- [Finding Feature Information, on page 173](#)
- [Information About Configuring System Message Logs, on page 173](#)
- [How to Configure System Message Logs, on page 176](#)
- [Monitoring and Maintaining System Message Logs, on page 184](#)
- [Configuration Examples for System Message Logs, on page 184](#)
- [Feature History and Information For System Message Logs, on page 185](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and 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 feature information table at the end of this module.

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 System Message Logs

System Message Logging

By default, a switch sends the output from system messages and **debug** privileged EXEC commands to a logging process. . 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. If a standalone switch, 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.



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 9: System Log Message Elements

Element	Description
<i>seq no:</i>	Stamps log messages with a sequence number only if the service sequence-numbers global configuration command is configured.
<i>timestamp</i> formats: <i>mm/dd h h:mm:ss</i> or <i>hh:mm:ss</i> (short uptime) or <i>d h</i> (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.
<i>facility</i>	The facility to which the message refers (for example, SNMP, SYS, and so forth).
<i>severity</i>	Single-digit code from 0 to 7 that is the severity of the message.
<i>MNEMONIC</i>	Text string that uniquely describes the message.

Element	Description
<i>description</i>	Text string containing detailed information about the event being reported.
<i>hostname-n</i>	Hostname of a stack member and its switch number in the stack. Though the active switch is a stack member, it does <i>not</i> append its hostname to system messages.

Default System Message Logging Settings

Table 10: 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.
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: Device# configure terminal	Enters global configuration mode.
Step 2	logging buffered <i>[size]</i> Example: Device(config)# logging buffered 8192	<p>Logs messages to an internal buffer on the switch or on a standalone switch or, in the case of a switch stack, on the active switch. The range is 4096 to 2147483647 bytes. The default buffer size is 4096 bytes.</p> <p>If a standalone switch or the active switch fails, the log file is lost unless you previously saved it to flash memory. See Step 4.</p> <p>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 <i>not</i> be set to this amount.</p>
Step 3	logging <i>host</i> Example: Device(config)# logging 125.1.1.100	<p>Logs messages to a UNIX syslog server host.</p> <p><i>host</i> specifies the name or IP address of the host to be used as the syslog server.</p> <p>To build a list of syslog servers that receive logging messages, enter this command more than once.</p>

	Command or Action	Purpose
Step 4	<p>logging file flash: <i>filename</i> [<i>max-file-size</i> [<i>min-file-size</i>]] [<i>severity-level-number</i> <i>type</i>]</p> <p>Example:</p> <pre>Device(config)# logging file flash:log_msg.txt 40960 4096 3</pre>	<p>Stores log messages in a file in flash memory on a standalone switch or, in the case of a switch stack, on the active switch.</p> <ul style="list-style-type: none"> • <i>filename</i>—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) <i>min-file-size</i>—Specifies the minimum logging file size. The range is 1024 to 2147483647. The default is 2048 bytes. • (Optional) <i>severity-level-number</i> <i>type</i>—Specifies either the logging severity level or the logging type. The severity range is 0 to 7.
Step 5	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 6	<p>terminal monitor</p> <p>Example:</p> <pre>Device# terminal monitor</pre>	<p>Logs messages to a nonconsole terminal during the current session.</p> <p>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.</p>

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	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	<p>line [console vty] <i>line-number</i> [<i>ending-line-number</i>]</p> <p>Example:</p> <pre>Device(config)# line console</pre>	<p>Specifies the line to be configured for synchronous logging of messages.</p> <ul style="list-style-type: none"> • console—Specifies configurations that occur through the switch console port or the Ethernet management port. • line vty <i>line-number</i>—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. <p>You can change the setting of all 16 vty lines at once by entering:</p> <pre>line vty 0 15</pre> <p>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:</p> <pre>line vty 2</pre> <p>When you enter this command, the mode changes to line configuration.</p>
Step 3	<p>logging synchronous [level [<i>severity-level</i> all] limit <i>number-of-buffers</i>]</p> <p>Example:</p> <pre>Device(config)# logging synchronous level 3 limit 1000</pre>	<p>Enables synchronous logging of messages.</p> <ul style="list-style-type: none"> • (Optional) level <i>severity-level</i>—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 <i>number-of-buffers</i>—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.

	Command or Action	Purpose
Step 4	end Example: <pre>Device(config)# end</pre>	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: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	no logging console Example: <pre>Device(config)# no logging console</pre>	Disables message logging.
Step 3	end Example: <pre>Device(config)# end</pre>	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: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	Use one of these commands: <ul style="list-style-type: none"> • service timestamps log uptime • service timestamps log datetime[msec localtime show-timezone] Example: Device(config)# <code>service timestamps log uptime</code> or Device(config)# <code>service timestamps log datetime</code>	Enables log time stamps. <ul style="list-style-type: none"> • 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: Device(config)# <code>end</code>	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: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	service sequence-numbers Example: Device(config)# <code>service sequence-numbers</code>	Enables sequence numbers.
Step 3	end Example: Device(config)# <code>end</code>	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: Device# <code>configure terminal</code>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	logging console <i>level</i> Example: Device(config)# logging console 3	Limits messages logged to the console. By default, the console receives debugging messages and numerically lower levels.
Step 3	logging monitor <i>level</i> Example: Device(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 <i>level</i> Example: Device(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: Device(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: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 2	logging history level Example: Device(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: Device(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: Device(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 <code>/etc/syslog.conf</code> . Example:	<ul style="list-style-type: none"> • local7—Specifies the logging facility.

	Command or Action	Purpose
	<code>local7.debug /usr/adm/logs/cisco.log</code>	<ul style="list-style-type: none"> 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: <pre>\$ touch /var/log/cisco.log \$ chmod 666 /var/log/cisco.log</pre>	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: <pre>\$ kill -HUP `cat /etc/syslog.pid`</pre>	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
<pre>show archive log config {all number [<i>end-number</i>] user <i>username</i> [<i>session number</i>] number [<i>end-number</i>] statistics} [<i>provisioning</i>]</pre>	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 active stack and a stack member (hostname *Switch-2*):

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channell1, 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-channell1, 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-channell1, 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)
```

Feature History and Information For System Message Logs

Release	Modification
Cisco IOS XE Everest 16.5.1a	This feature was introduced.



CHAPTER 8

Configuring Online Diagnostics

- [Information About Configuring Online Diagnostics, on page 187](#)
- [How to Configure Online Diagnostics, on page 189](#)
- [Monitoring and Maintaining Online Diagnostics, on page 194](#)
- [Configuration Examples for Online Diagnostic Tests, on page 194](#)
- [Additional References for Online Diagnostics, on page 196](#)
- [Feature Information for Configuring Online Diagnostics, on page 196](#)

Information About Configuring Online Diagnostics

With online diagnostics, you can test and verify the hardware functionality of the device while the device 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 device is connected to a live network; and health-monitoring runs in the background with user-defined intervals. The health-monitoring test runs for every 90, 100, or 150 seconds based on the test.

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 device or switch stack and the diagnostic tests that have already run.

Generic Online Diagnostics (GOLD)



Note

- Before you enable any online diagnostics tests, enable console logging to see all warning messages.
- While tests are running, all ports are shut down as a stress test is being performed with looping ports internally and external traffic might affect the test results. The switch must be rebooted to bring the switch to normal operation. When you issue the command to reload the switch, the system will ask you if the configuration should be saved.
- Do not save the configuration.
- If you are running the tests on other modules, after the test is initiated and complete, you must reset the module.

TestPortTxMonitoring

This test periodically monitors data-path traffic in transmitted direction of each network port that are physically connected to a device with the status as UP, and completes within a millisecond per port. This test also monitors the transmit counters at ASIC level to verify that the ports are not stuck. The test displays syslog messages and users can take corrective actions using the IOS Embedded Event Manager (EEM). Configure the time interval and threshold by entering the **diagnostic monitor interval** and **diagnostic monitor threshold** commands, respectively. The test leverages the Cisco Discovery Protocol (CDP) protocol that transmits packets. The test runs every 75 seconds, and the failure threshold is set to five by default.

Attribute	Description
Disruptive or Nondisruptive	Nondisruptive.
Recommendation	Do not disable.
Default	On.
Initial Release	IOS XE Everest 16.9.1.
Corrective action	Display a syslog message indicating the port(s) that failed.
Hardware support	All modules including supervisor engines.

TestUnusedPortLoopback

This test periodically verifies the data path between the supervisor module and network ports of a module during runtime to determine if any incoming network interface ports are locked. In this test, a Layer 2 packet is flooded onto the VLAN associated with the test port and the inband port of the supervisor engine. The packet loops back into the test port and returns to the supervisor engine on the same VLAN. This test runs only on unused (admin down, that is, the ports are shut down) network ports irrespective of whether a cable is connected or not, and completes within a millisecond per port. This test substitutes the lack of a nondisruptive loopback test in current ASICs. This test runs every 60 seconds.

Attribute	Description
Disruptive or Nondisruptive	Nondisruptive.
Recommendation	Do not disable. This test is automatically disabled during CPU-usage spikes to maintain accuracy.
Default	On.
Initial Release	IOS XE Everest 16.9.1.
Corrective action	Displays a syslog message indicating the port(s) that failed. For modules other than supervisor engines, if all port groups fail (for example, at least one port per port ASIC fails more than the failure threshold for all port ASICs), the default action is to reset the module and power down the module after two resets.
Hardware support	All modules including supervisor engines.

How to Configure Online Diagnostics

Starting Online Diagnostic Tests

After you configure diagnostic tests to run on the Device, 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 <i>number</i> test { <i>name</i> <i>test-id</i> <i>test-id-range</i> all basic complete minimal non-disruptive per-port } Example: <pre>Device# diagnostic start switch 2 test basic</pre>	Starts the diagnostic tests. The switch <i>number</i> keyword is supported only on stacking Device. You can specify the tests by using one of these options: <ul style="list-style-type: none"> • <i>name</i>—Enters the name of the test. • <i>test-id</i>—Enters the ID number of the test. • <i>test-id-range</i>—Enters the range of test IDs by using integers separated by a comma and a hyphen.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • all—Starts all of the tests. • basic— Starts the basic test suite. • 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.

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 Device. 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: Device# configure terminal	Enters global configuration mode.
Step 2	diagnostic schedule switch <i>number test</i> { <i>name</i> <i>test-id</i> <i>test-id-range</i> all basic complete minimal non-disruptive per-port } { daily on <i>mm dd yyyy hh:mm</i> port <i>inter-port-number port-number-list</i> weekly <i>day-of-week hh:mm</i> }	Schedules on-demand diagnostic tests for a specific day and time. The switch <i>number</i> keyword is supported only on stacking switches. The range is from 1 to 8 When specifying the tests to be scheduled, use these options: <ul style="list-style-type: none"> • <i>name</i>—Name of the test that appears in the show diagnostic content command output. • <i>test-id</i>—ID number of the test that appears in the show diagnostic content command output.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • <i>test-id-range</i>—ID numbers of the tests that appear in the show diagnostic content command output. • 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. <p>You can schedule the tests as follows:</p> <ul style="list-style-type: none"> • Daily—Use the daily <i>hh:mm</i> parameter. • Specific day and time—Use the on <i>mm dd yyyy hh:mm</i> parameter. • Weekly—Use the weekly <i>day-of-week hh:mm</i> parameter.

Configuring Health-Monitoring Diagnostics

You can configure health-monitoring diagnostic testing on a Device while it is connected to a live network. You can configure the execution interval for each health-monitoring test, enable the Device to generate a syslog message because of a test failure, and enable a specific test.

Use the **no** form of this command to disable testing.

By default, health monitoring is disabled, but the Device generates a syslog message when a test fails.

Follow these steps to configure and enable the health-monitoring diagnostic tests:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **diagnostic monitor interval** *switchnumber* **test** {*name* | *test-id* | *test-id-range* | **all**} *hh:mm:ss milliseconds day*
4. **diagnostic monitor syslog**
5. **diagnostic monitor threshold** *switch number number* **test** {*name* | *test-id* | *test-id-range* | **all**} **failure count** *count*
6. **diagnostic monitor** *switchnumber* **test** {*name* | *test-id* | *test-id-range* | **all**}
7. **end**
8. **show diagnostic** { **content** | **post** | **result** | **schedule** | **status** | **switch** }
9. **show running-config**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	diagnostic monitor interval switch <i>number test {name test-id test-id-range all} hh:mm:ss milliseconds day</i> Example: Device (config)# diagnostic monitor interval switch 2 test 1 12:30:00 750 5	Configures the health-monitoring interval of the specified tests. The switch <i>number</i> keyword is supported only on stacking switches. The range is from 1 to 9. When specifying the tests, use one of these parameters: <ul style="list-style-type: none"> • <i>name</i>—Name of the test that appears in the show diagnostic content command output. • <i>test-id</i>—ID number of the test that appears in the show diagnostic content command output. • <i>test-id-range</i>—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: <ul style="list-style-type: none"> • <i>hh:mm:ss</i>—Monitoring interval in hours, minutes, and seconds. The range for <i>hh</i> is 0 to 24, and the range for <i>mm</i> and <i>ss</i> is 0 to 60. • <i>milliseconds</i>—Monitoring interval in milliseconds (ms). The range is from 0 to 999. • <i>day</i>—Monitoring interval in the number of days. The range is from 0 to 20.
Step 4	diagnostic monitor syslog Example: Device (config)# diagnostic monitor syslog	(Optional) Configures the switch to generate a syslog message when a health-monitoring test fails.
Step 5	diagnostic monitor threshold switch <i>number number test {name test-id test-id-range all} failure count count</i> Example:	(Optional) Sets the failure threshold for the health-monitoring tests. When specifying the tests, use one of these parameters:

	Command or Action	Purpose
	<pre>Device(config)# diagnostic monitor threshold switch 2 test 1 failure count 20</pre>	<ul style="list-style-type: none"> • <i>name</i>—Name of the test that appears in the show diagnostic content command output. • <i>test-id</i>—ID number of the test that appears in the show diagnostic content command output. • <i>test-id-range</i>—ID numbers of the tests that appear in the show diagnostic content command output. • all—All of the diagnostic tests. <p>The range for the failure threshold <i>count</i> is 0 to 99.</p>
Step 6	<p>diagnostic monitor switch<i>number</i> test {<i>name</i> <i>test-id</i> <i>test-id-range</i> all}</p> <p>Example:</p> <pre>Device(config)# diagnostic monitor switch 2 test 1</pre>	<p>Enables the specified health-monitoring tests.</p> <p>The switch <i>number</i> keyword is supported only on stacking switches.</p> <p>When specifying the tests, use one of these parameters:</p> <ul style="list-style-type: none"> • <i>name</i>—Name of the test that appears in the show diagnostic content command output. • <i>test-id</i>—ID number of the test that appears in the show diagnostic content command output. • <i>test-id-range</i>—ID numbers of the tests that appear in the show diagnostic content command output. • all—All of the diagnostic tests.
Step 7	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 8	<p>show diagnostic { content post result schedule status switch }</p>	Display the online diagnostic test results and the supported test suites.
Step 9	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 10	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Monitoring and Maintaining Online Diagnostics

Configuration Examples for Online Diagnostic Tests

Examples: Start Diagnostic Tests

This example shows how to start a diagnostic test by using the test name:

```
Device# diagnostic start switch 2 test DiagPOETest
```

This example shows how to start all of the basic diagnostic tests:

```
Device# diagnostic start switch 1 test all
```

Example: Configure a Health Monitoring Test

This example shows how to configure a health-monitoring test:

```
Device(config)# diagnostic monitor threshold switch 1 test 1 failure count 50  
Device(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:

```
Device(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:

```
Device(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:

```
Device# show diagnostic ondemand settings
```

```
Test iterations = 1  
Action on test failure = continue
```

This example shows how to display diagnostic events for errors:

```
Device# 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:

```
Device# 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.
```

```
Device#
```

Additional References for Online Diagnostics

Related Documents

Related Topic	Document Title
System management commands	<i>Command Reference (Catalyst 9300 Series Switches)</i> <i>Command Reference (Catalyst 9400 Series Switches)</i>

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
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>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.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for Configuring Online Diagnostics

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Everest 16.5.1a	Online Diagnostics	With online diagnostics, you can test and verify the hardware functionality of the device while the device is connected to a live network.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 9

Managing Configuration Files

- [Prerequisites for Managing Configuration Files, on page 199](#)
- [Restrictions for Managing Configuration Files, on page 199](#)
- [Information About Managing Configuration Files, on page 199](#)
- [How to Manage Configuration File Information, on page 206](#)
- [Feature History for Managing Configuration Files, on page 235](#)

Prerequisites for Managing Configuration Files

- You should have at least a basic familiarity with the Cisco IOS environment and the command-line interface.
- You should have at least a minimal configuration running on your system. You can create a basic configuration file using the **setup** command.

Restrictions for Managing Configuration Files

- Many of the Cisco IOS commands described in this document are available and function only in certain configuration modes on the device.
- Some of the Cisco IOS configuration commands are only available on certain device platforms, and the command syntax may vary on different platforms.

Information About Managing Configuration Files

Types of Configuration Files

Configuration files contain the Cisco IOS software commands used to customize the functionality of your Cisco device. Commands are parsed (translated and executed) by the Cisco IOS software when the system is booted (from the startup-config file) or when you enter commands at the CLI in a configuration mode.

Startup configuration files (startup-config) are used during system startup to configure the software. Running configuration files (running-config) contain the current configuration of the software. The two configuration

files can be different. For example, you may want to change the configuration for a short time period rather than permanently. In this case, you would change the running configuration using the **configure terminal EXEC** command but not save the configuration using the **copy running-config startup-config EXEC** command.

To change the running configuration, use the **configure terminal** command, as described in the [Modifying the Configuration File, on page 207](#) section. As you use the Cisco IOS configuration modes, commands generally are executed immediately and are saved to the running configuration file either immediately after you enter them or when you exit a configuration mode.

To change the startup configuration file, you can either save the running configuration file to the startup configuration using the **copy running-config startup-config EXEC** command or copy a configuration file from a file server to the startup configuration (see the [Copying a Configuration File from a TFTP Server to the Device](#) section for more information).

Configuration Mode and Selecting a Configuration Source

To enter configuration mode on the device, enter the **configure** command at the privileged EXEC prompt. The Cisco IOS software responds with the following prompt asking you to specify the terminal, memory, or a file stored on a network server (network) as the source of configuration commands:

```
Configuring from terminal, memory, or network [terminal]?
```

Configuring from the terminal allows you to enter configuration commands at the command line, as described in the following section. See the [Re-executing the Configuration Commands in the Startup Configuration File](#) section for more information.

Configuring from the network allows you to load and execute configuration commands over the network. See the [Copying a Configuration File from a TFTP Server to the Device](#) section for more information.

Configuration File Changes Using the CLI

The Cisco IOS software accepts one configuration command per line. You can enter as many configuration commands as you want. You can add comments to a configuration file describing the commands you have entered. Precede a comment with an exclamation point (!). Because comments are *not* stored in NVRAM or in the active copy of the configuration file, comments do not appear when you list the active configuration with the **show running-config** or **more system:running-config EXEC** command. Comments are not displayed when you list the startup configuration with the **show startup-config** or **more nvram:startup-config EXEC** mode command. Comments are stripped out of the configuration file when it is loaded onto the device. However, you can list the comments in configuration files stored on a File Transfer Protocol (FTP), Remote Copy Protocol (RCP), or Trivial File Transfer Protocol (TFTP) server. When you configure the software using the CLI, the software executes the commands as you enter them.

Location of Configuration Files

Configuration files are stored in the following locations:

- The running configuration is stored in RAM.
- On all platforms except the Class A Flash file system platforms, the startup configuration is stored in nonvolatile random-access memory (NVRAM).

- On Class A Flash file system platforms, the startup configuration is stored in the location specified by the CONFIG_FILE environment variable (see the [Specifying the CONFIG_FILE Environment Variable on Class A Flash File Systems](#), on page 230 section). The CONFIG_FILE variable defaults to NVRAM and can be a file in the following file systems:
 - **nvr**am: (NVRAM)
 - **flash**: (internal flash memory)
 - **usbflash0**: (external usbflash file system)

Copy Configuration Files from a Network Server to the Device

You can copy configuration files from a TFTP, rcp, or FTP server to the running configuration or startup configuration of the device. You may want to perform this function for one of the following reasons:

- To restore a backed-up configuration file.
- To use the configuration file for another device. For example, you may add another device to your network and want it to have a similar configuration to the original device. By copying the file to the new device, you can change the relevant parts rather than recreating the whole file.
- To load the same configuration commands on to all of the devices in your network so that all of the devices have similar configurations.

The **copy {ftp: | rcp: | tftp:system:running-config}** EXEC command loads the configuration files into the device as if you were typing the commands on the command line. The device does not erase the existing running configuration before adding the commands. If a command in the copied configuration file replaces a command in the existing configuration file, the existing command is erased. For example, if the copied configuration file contains a different IP address in a particular command than the existing configuration, the IP address in the copied configuration is used. However, some commands in the existing configuration may not be replaced or negated. In this case, the resulting configuration file is a mixture of the existing configuration file and the copied configuration file, with the copied configuration file having precedence.

To restore a configuration file to an exact copy of a file stored on a server, you need to copy the configuration file directly to the startup configuration (using the **copy ftp: | rcp: | tftp:} nvram:startup-config** command) and reload the device.

To copy configuration files from a server to a device, perform the tasks described in the following sections.

The protocol that you use depends on which type of server you are using. The FTP and rcp transport mechanisms provide faster performance and more reliable delivery of data than TFTP. These improvements are possible because the FTP and rcp transport mechanisms are built on and use the TCP/IP stack, which is connection-oriented.

Copying a Configuration File from the Device to a TFTP Server

In some implementations of TFTP, you must create a dummy file on the TFTP server and give it read, write, and execute permissions before copying a file over it. Refer to your TFTP documentation for more information.

Copying a Configuration File from the Device to an RCP Server

You can copy a configuration file from the device to an RCP server.

One of the first attempts to use the network as a resource in the UNIX community resulted in the design and implementation of the remote shell protocol, which included the remote shell (rsh) and remote copy (rcp) functions. Rsh and rcp give users the ability to execute commands remotely and copy files to and from a file system residing on a remote host or server on the network. The Cisco implementation of rsh and rcp interoperates with standard implementations.

The rcp **copy** commands rely on the rsh server (or daemon) on the remote system. To copy files using rcp, you need not create a server for file distribution, as you do with TFTP. You need only have access to a server that supports the remote shell (rsh). (Most UNIX systems support rsh.) Because you are copying a file from one place to another, you must have read permission on the source file and write permission on the destination file. If the destination file does not exist, rcp creates it for you.

Although the Cisco rcp implementation emulates the functions of the UNIX rcp implementation—copying files among systems on the network—the Cisco command syntax differs from the UNIX rcp command syntax. The Cisco rcp support offers a set of **copy** commands that use rcp as the transport mechanism. These rcp **copy** commands are similar in style to the Cisco TFTP **copy** commands, but they offer an alternative that provides faster performance and reliable delivery of data. These improvements are possible because the rcp transport mechanism is built on and uses the TCP/IP stack, which is connection-oriented. You can use rcp commands to copy system images and configuration files from the device to a network server and vice versa.

You also can enable rcp support to allow users on remote systems to copy files to and from the device.

To configure the Cisco IOS software to allow remote users to copy files to and from the device, use the **ip rcmd rcp-enable** global configuration command.

Restrictions

The RCP protocol requires a client to send a remote username on each RCP request to a server. When you copy a configuration file from the device to a server using RCP, the Cisco IOS software sends the first valid username it encounters in the following sequence:

1. The username specified in the **copy EXEC** command, if a username is specified.
2. The username set by the **ip rcmd remote-username** global configuration command, if the command is configured.
3. The remote username associated with the current tty (terminal) process. For example, if the user is connected to the device through Telnet and was authenticated through the **username** command, the device software sends the Telnet username as the remote username.
4. The device host name.

For the RCP copy request to execute successfully, an account must be defined on the network server for the remote username. If the server has a directory structure, the configuration file or image is written to or copied from the directory associated with the remote username on the server. For example, if the system image resides in the home directory of a user on the server, you can specify that user name as the remote username.

Use the **ip rcmd remote-username** command to specify a username for all copies. (Rcmd is a UNIX routine used at the super-user level to execute commands on a remote machine using an authentication scheme based on reserved port numbers. Rcmd stands for “remote command”). Include the username in the **copy** command if you want to specify a username for that copy operation only.

If you are writing to the server, the RCP server must be properly configured to accept the RCP write request from the user on the device. For UNIX systems, you must add an entry to the .rhosts file for the remote user on the RCP server. For example, suppose the device contains the following configuration lines:


```
hostname Device1
ip rcmd remote-username User0
```

If the device IP address translates to device1.example.com, then the .rhosts file for User0 on the RCP server should contain the following line:

```
Device1.example.com Device1
```

Requirements for the RCP Username

The RCP protocol requires a client to send a remote username on each RCP request to a server. When you copy a configuration file from the device to a server using RCP, the Cisco IOS software sends the first valid username it encounters in the following sequence:

1. The username specified in the **copy EXEC** command, if a username is specified.
2. The username set by the **ip rcmd remote-username** global configuration command, if the command is configured.
3. The remote username associated with the current tty (terminal) process. For example, if the user is connected to the device through Telnet and is authenticated through the **username** command, the device software sends the Telnet username as the remote username.
4. The device host name.

For the RCP copy request to execute, an account must be defined on the network server for the remote username. If the server has a directory structure, the configuration file or image is written to or copied from the directory associated with the remote username on the server. For example, if the system image resides in the home directory of a user on the server, specify that user name as the remote username.

Refer to the documentation for your RCP server for more information.

Copying a Configuration File from the Device to an FTP Server

You can copy a configuration file from the device to an FTP server.

Understanding the FTP Username and Password



Note The password must not contain the special character '@'. If the character '@' is used, the copy fails to parse the IP address of the server.

The FTP protocol requires a client to send a remote username and password on each FTP request to a server. When you copy a configuration file from the device to a server using FTP, the Cisco IOS software sends the first valid username it encounters in the following sequence:

1. The username specified in the **copy EXEC** command, if a username is specified.
2. The username set by the **ip ftp username** global configuration command, if the command is configured.
3. Anonymous.

The device sends the first valid password it encounters in the following sequence:

1. The password specified in the **copy** command, if a password is specified.
2. The password set by the **ip ftp password** command, if the command is configured.
3. The device forms a password *username @devicename.domain* . The variable *username* is the username associated with the current session, *devicename* is the configured host name, and *domain* is the domain of the device.

The username and password must be associated with an account on the FTP server. If you are writing to the server, the FTP server must be properly configured to accept the FTP write request from the user on the device.

If the server has a directory structure, the configuration file or image is written to or copied from the directory associated with the username on the server. For example, if the system image resides in the home directory of a user on the server, specify that user name as the remote username.

Refer to the documentation for your FTP server for more information.

Use the **ip ftp username** and **ip ftp password** global configuration commands to specify a username and password for all copies. Include the username in the **copy EXEC** command if you want to specify a username for that copy operation only.

Copying files through a VRF

You can copy files through a VRF interface specified in the **copy** command. Specifying the VRF in the **copy** command is easier and more efficient as you can directly change the source interface without using a change request for the configuration.

Example

The following example shows how to copy files through a VRF, using the **copy** command:

```
Device# copy scp: flash-1: vrf test-vrf
Address or name of remote host [10.1.2.3]?
Source username [ScpUser]?
Source filename [/auto/tftp-server/ScpUser/vrf_test.txt]?
Destination filename [vrf_test.txt]?
Getting the vrf name as test-vrf
Password:
Sending file modes: C0644 10 vrf_test.txt
!
223 bytes copied in 22.740 secs (10 bytes/sec)
```

Copy Configuration Files from a Switch to Another Switch

You can copy the configurations from one switch to another. This is a 2-step process - Copy the configurations from the switch to the TFTP server, and then from TFTP to another switch.

To copy your current configurations from the switch, run the command **copy startup-config tftp:** and follow the instructions. The configurations are copied onto the TFTP server.

Then, login to another switch and run the command **copy tftp: startup-config** and follow the instructions. The configurations are now copied onto the other switch.

After the configurations are copied, to save your configurations, use **write memory** command and then either reload the switch or run the **copy startup-config running-config** command

Configuration Files Larger than NVRAM

To maintain a configuration file that exceeds the size of NVRAM, you should be aware of the information in the following sections.

Compressing the Configuration File

The **service compress-config** global configuration command specifies that the configuration file be stored compressed in NVRAM. Once the configuration file has been compressed, the device functions normally. When the system is booted, it recognizes that the configuration file is compressed, expands it, and proceeds normally. The **more nvram:startup-config EXEC** command expands the configuration before displaying it.

Before you compress configuration files, refer to the appropriate hardware installation and maintenance publication. Verify that your system's ROMs support file compression. If not, you can install new ROMs that support file compression.

The size of the configuration must not exceed three times the NVRAM size. For a 128-KB size NVRAM, the largest expanded configuration file size is 384 KB.

The **service compress-config** global configuration command works only if you have Cisco IOS software Release 10.0 or later release boot ROMs. Installing new ROMs is a one-time operation and is necessary only if you do not already have Cisco IOS Release 10.0 in ROM. If the boot ROMs do not recognize a compressed configuration, the following message is displayed:

```
Boot ROMs do not support NVRAM compression Config NOT written to NVRAM
```

Storing the Configuration in Flash Memory on Class A Flash File Systems

On class A Flash file system devices, you can store the startup configuration in flash memory by setting the **CONFIG_FILE** environment variable to a file in internal flash memory or flash memory in a PCMCIA slot.

See the [Specifying the CONFIG_FILE Environment Variable on Class A Flash File Systems](#), on page 230 section for more information.

Care must be taken when editing or changing a large configuration. Flash memory space is used every time a **copy system:running-config nvram:startup-config EXEC** command is issued. Because file management for flash memory (such as optimizing free space) is not done automatically, you must pay close attention to available flash memory. Use the **squeeze** command to reclaim used space. We recommend that you use a large-capacity Flash card of at least 20 MB.

Loading the Configuration Commands from the Network

You can also store large configurations on FTP, RCP, or TFTP servers and download them at system startup. To use a network server to store large configurations, see the [Copying a Configuration File from the Device to a TFTP Server](#), on page 209 and [Configuring the Device to Download Configuration Files](#), on page 205 sections for more information on these commands.

Configuring the Device to Download Configuration Files

You can configure the device to load one or two configuration files at system startup. The configuration files are loaded into memory and read in as if you were typing the commands at the command line. Thus, the configuration for the device is a mixture of the original startup configuration and the one or two downloaded configuration files.

Network Versus Host Configuration Files

For historical reasons, the first file the device downloads is called the network configuration file. The second file the device downloads is called the host configuration file. Two configuration files can be used when all of the devices on a network use many of the same commands. The network configuration file contains the standard commands used to configure all of the devices. The host configuration files contain the commands specific to one particular host. If you are loading two configuration files, the host configuration file should be the configuration file you want to have precedence over the other file. Both the network and host configuration files must reside on a network server reachable via TFTP, RCP, or FTP, and must be readable.

How to Manage Configuration File Information

Displaying Configuration File Information

To display information about configuration files, complete the tasks in this section:

SUMMARY STEPS

1. **enable**
2. **show boot**
3. **more *file-url***
4. **show running-config**
5. **show startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show boot Example: Device# show boot	Lists the contents of the BOOT environment variable (if set), the name of the configuration file pointed to by the CONFIG_FILE environment variable, and the contents of the BOOTLDR environment variable.
Step 3	more <i>file-url</i> Example: Device# more 10.1.1.1	Displays the contents of a specified file.
Step 4	show running-config Example: Device# show running-config	Displays the contents of the running configuration file. (Command alias for the more system:running-config command.)

	Command or Action	Purpose
Step 5	show startup-config Example: <pre>Device# show startup-config</pre>	<p>Displays the contents of the startup configuration file. (Command alias for the more nvram:startup-config command.)</p> <p>On all platforms except the Class A Flash file system platforms, the default startup-config file usually is stored in NVRAM.</p> <p>On the Class A Flash file system platforms, the CONFIG_FILE environment variable points to the default startup-config file.</p> <p>The CONFIG_FILE variable defaults to NVRAM.</p>

Modifying the Configuration File

The Cisco IOS software accepts one configuration command per line. You can enter as many configuration commands as you want. You can add comments to a configuration file describing the commands you have entered. Precede a comment with an exclamation point (!). Because comments are *not* stored in NVRAM or in the active copy of the configuration file, comments do not appear when you list the active configuration with the **show running-config** or **more system:running-config** EXEC commands. Comments do not display when you list the startup configuration with the **show startup-config** or **more nvram:startup-config** EXEC mode commands. Comments are stripped out of the configuration file when it is loaded onto the device. However, you can list the comments in configuration files stored on a File Transfer Protocol (FTP), Remote Copy Protocol (RCP), or Trivial File Transfer Protocol (TFTP) server. When you configure the software using the CLI, the software executes the commands as you enter them. To configure the software using the CLI, use the following commands in privileged EXEC mode:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **configuration command**
4. Do one of the following:
 - **end**
 - **^Z**
5. **copy system:running-config nvram:startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	configuration command Example: <pre>Device(config)# configuration command</pre>	Enter the necessary configuration commands. The Cisco IOS documentation set describes configuration commands organized by technology.
Step 4	Do one of the following: <ul style="list-style-type: none"> • end • ^Z Example: <pre>Device(config)# end</pre>	Ends the configuration session and exits to EXEC mode. Note When you press the Ctrl and Z keys simultaneously, ^Z is displayed to the screen.
Step 5	copy system:running-config nvram:startup-config Example: <pre>Device# copy system:running-config nvram:startup-config</pre>	Saves the running configuration file as the startup configuration file. You may also use the copy running-config startup-config command alias, but you should be aware that this command is less precise. On most platforms, this command saves the configuration to NVRAM. On the Class A Flash file system platforms, this step saves the configuration to the location specified by the CONFIG_FILE environment variable (the default CONFIG_FILE variable specifies that the file should be saved to NVRAM).

Examples

In the following example, the device prompt name of the device is configured. The comment line, indicated by the exclamation mark (!), does not execute any command. The **hostname** command is used to change the device name from device to new_name. By pressing Ctrl-Z (^Z) or entering the **end** command, the user quits configuration mode. The **copy system:running-config nvram:startup-config** command saves the current configuration to the startup configuration.

```
Device# configure terminal
Device(config)# !The following command provides the switch host name.
Device(config)# hostname new_name
new_name(config)# end
new_name# copy system:running-config nvram:startup-config
```

When the startup configuration is NVRAM, it stores the current configuration information in text format as configuration commands, recording only non-default settings. The memory is checksummed to guard against corrupted data.



Note Some specific commands might not get saved to NVRAM. You need to enter these commands again if you reboot the machine. These commands are noted in the documentation. We recommend that you keep a list of these settings so that you can quickly reconfigure your device after rebooting.

Copying a Configuration File from the Device to a TFTP Server

To copy configuration information on a TFTP network server, complete the tasks in this section:

SUMMARY STEPS

1. **enable**
2. **copy system:running-config tftp: [[[//location]/directory]/filename]**
3. **copy nvram:startup-config tftp: [[[//location]/directory]/filename]**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	copy system:running-config tftp: [[[//location]/directory]/filename] Example: Device# copy system:running-config tftp: //server1/topdir/file10	Copies the running configuration file to a TFTP server.
Step 3	copy nvram:startup-config tftp: [[[//location]/directory]/filename] Example: Device# copy nvram:startup-config tftp: //server1/1stidir/file10	Copies the startup configuration file to a TFTP server.

Examples

The following example copies a configuration file from a device to a TFTP server:

```
Device# copy system:running-config tftp://172.16.2.155/tokyo-config
Write file tokyo-config on host 172.16.2.155? [confirm] Y
Writing tokyo-config!!! [OK]
```

What to Do Next

After you have issued the **copy** command, you may be prompted for additional information or for confirmation of the action. The prompt displayed depends on how much information you provide in the **copy** command and the current setting of the **file prompt** global configuration command.

Copying a Configuration File from the Device to an RCP Server

To copy a startup configuration file or a running configuration file from the device to an RCP server, use the following commands beginning in privileged EXEC mode:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip rcmd remote-username** *username*
4. **end**
5. Do one of the following:
 - **copy system:running-config rcp:** [[[//[*username@*]*location*]/*directory*]/*filename*]
 - **copy nvram:startup-config rcp:** [[[//[*username@*]*location*]/*directory*]/*filename*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip rcmd remote-username <i>username</i> Example: Device(config)# ip rcmd remote-username NetAdmin1	(Optional) Changes the default remote username.
Step 4	end Example: Device(config)# end	(Optional) Exits global configuration mode.
Step 5	Do one of the following: <ul style="list-style-type: none"> • copy system:running-config rcp: [[[//[<i>username@</i>]<i>location</i>]/<i>directory</i>]/<i>filename</i>] 	<ul style="list-style-type: none"> • Specifies that the device running configuration file is to be stored on an RCP server or

	Command or Action	Purpose
	<ul style="list-style-type: none"> • copy nvram:startup-config rcp: [[[/[username@]location]/directory]/filename] <p>Example:</p> <pre>Device# copy system:running-config rcp: //NetAdmin1@example.com/dir-files/file1</pre>	<ul style="list-style-type: none"> • Specifies that the device startup configuration file is to be stored on an RCP server

Examples

Storing a Running Configuration File on an RCP Server

The following example copies the running configuration file named runfile2-config to the netadmin1 directory on the remote host with an IP address of 172.16.101.101:

```
Device# copy system:running-config rcp://netadmin1@172.16.101.101/runfile2-config
Write file runfile2-config on host 172.16.101.101?[confirm]
Building configuration...[OK]
Connected to 172.16.101.101
Device#
```

Storing a Startup Configuration File on an RCP Server

The following example shows how to store a startup configuration file on a server by using RCP to copy the file:

```
Device# configure terminal
Device(config)# ip rcmd remote-username netadmin2
Device(config)# end
Device# copy nvram:startup-config rcp:
Remote host[]? 172.16.101.101
Name of configuration file to write [start-config]?
Write file start-config on host 172.16.101.101?[confirm]
![OK]
```

What to Do Next

After you have issued the **copy EXEC** command, you may be prompted for additional information or for confirmation of the action. The prompt displayed depends on how much information you provide in the **copy** command and the current setting of the **file prompt** global configuration command.

Copying a Configuration File from the Device to the FTP Server

To copy a startup configuration file or a running configuration file from the device to an FTP server, complete the following tasks:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip ftp username** *username*
4. **ip ftp password** *password*
5. **end**
6. Do one of the following:
 - **copy system:running-config ftp:** [[[/[*username* [:*password*]@]*location*]/*directory*]/*filename*]
 - or
 - **copy nvram:startup-config ftp:** [[[/[*username* [:*password*]@]*location*]/*directory*]/*filename*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode on the device.
Step 3	ip ftp username <i>username</i> Example: Device(config)# ip ftp username NetAdmin1	(Optional) Specifies the default remote username.
Step 4	ip ftp password <i>password</i> Example: Device(config)# ip ftp password adminpassword	(Optional) Specifies the default password.
Step 5	end Example: Device(config)# end	(Optional) Exits global configuration mode. This step is required only if you override the default remote username or password (see Steps 2 and 3).
Step 6	Do one of the following: <ul style="list-style-type: none"> • copy system:running-config ftp: [[[/[<i>username</i> [:<i>password</i>]@]<i>location</i>]/<i>directory</i>]/<i>filename</i>] • copy nvram:startup-config ftp: [[[/[<i>username</i> [:<i>password</i>]@]<i>location</i>]/<i>directory</i>]/<i>filename</i>] Example:	Copies the running configuration or startup configuration file to the specified location on the FTP server.

	Command or Action	Purpose
	Device# <code>copy system:running-config ftp:</code>	

Examples

Storing a Running Configuration File on an FTP Server

The following example copies the running configuration file named `runfile-config` to the `netadmin1` directory on the remote host with an IP address of `172.16.101.101`:

```
Device# copy system:running-config ftp://netadmin1:mypass@172.16.101.101/runfile-config
Write file runfile-config on host 172.16.101.101?[confirm]
Building configuration...[OK]
Connected to 172.16.101.101
Device#
```

Storing a Startup Configuration File on an FTP Server

The following example shows how to store a startup configuration file on a server by using FTP to copy the file:

```
Device# configure terminal
Device(config)# ip ftp username netadmin2
Device(config)# ip ftp password mypass
Device(config)# end
Device# copy nvram:startup-config ftp:
Remote host[]? 172.16.101.101
Name of configuration file to write [start-config]?
Write file start-config on host 172.16.101.101?[confirm]
![OK]
```

What to Do Next

After you have issued the `copy EXEC` command, you may be prompted for additional information or for confirmation of the action. The prompt displayed depends on how much information you provide in the `copy` command and the current setting of the `file prompt` global configuration command.

Copying a Configuration File from a TFTP Server to the Device

To copy a configuration file from a TFTP server to the device, complete the tasks in this section:

SUMMARY STEPS

1. `enable`
2. `copy tftp: [[[/location]/directory]/filename] system:running-config`
3. `copy tftp: [[[/location]/directory]/filename] nvram:startup-config`
4. `copy tftp: [[[/location]/directory]/filename] flash-[n]:/directory/startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	copy tftp: [[[//location]/directory]/filename] system:running-config Example: Device# copy tftp://server1/dir10/datasource system:running-config	Copies a configuration file from a TFTP server to the running configuration.
Step 3	copy tftp: [[[//location]/directory]/filename] nvrnram:startup-config Example: Device# copy tftp://server1/dir10/datasource nvrnram:startup-config	Copies a configuration file from a TFTP server to the startup configuration.
Step 4	copy tftp: [[[//location]directory]/filename]flash-[n]:/directory/startup-config Example: Device# copy tftp://server1/dir10/datasource flash:startup-config	Copies a configuration file from a TFTP server to the startup configuration.

Examples

In the following example, the software is configured from the file named **tokyo-config** at IP address 172.16.2.155:

```
Device# copy tftp://172.16.2.155/tokyo-config system:running-config
Configure using tokyo-config from 172.16.2.155? [confirm] Y
Booting tokyo-config from 172.16.2.155:!!! [OK - 874/16000 bytes]
```

What to Do Next

After you have issued the **copy EXEC** command, you may be prompted for additional information or for confirmation of the action. The prompt displayed depends on how much information you provide in the **copy** command and the current setting of the **file prompt** global configuration command.

Copying a Configuration File from the rcp Server to the Device

To copy a configuration file from an rcp server to the running configuration or startup configuration, complete the following tasks:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip rcmd remote-username *username***
4. **end**
5. Do one of the following:
 - **copy rcp:[[[//[*username@*]*location*]/*directory*]/*filename*]**system:running-config****
 - **copy rcp:[[[//[*username@*]*location*]/*directory*]/*filename*]**nvrnram:startup-config****

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	(Optional) Enters configuration mode from the terminal. This step is required only if you override the default remote username (see Step 3).
Step 3	ip rcmd remote-username <i>username</i> Example: Device(config)# ip rcmd remote-username NetAdmin1	(Optional) Specifies the remote username.
Step 4	end Example: Device(config)# end	(Optional) Exits global configuration mode. This step is required only if you override the default remote username (see Step 2).
Step 5	Do one of the following: <ul style="list-style-type: none"> • copy rcp:[[[//[<i>username@</i>]<i>location</i>]/<i>directory</i>]/<i>filename</i>]system:running-config • copy rcp:[[[//[<i>username@</i>]<i>location</i>]/<i>directory</i>]/<i>filename</i>]nvrnram:startup-config Example: Device# copy	Copies the configuration file from an rcp server to the running configuration or startup configuration.

	Command or Action	Purpose
	rcp://[user1@example.com/dir10/fileone] nvram:startup-config	

Examples

Copy RCP Running-Config

The following example copies a configuration file named `host1-config` from the `netadmin1` directory on the remote server with an IP address of `172.16.101.101`, and loads and runs the commands on the device:

```
Device# copy rcp://netadmin1@172.16.101.101/host1-config system:running-config
Configure using host1-config from 172.16.101.101? [confirm]
Connected to 172.16.101.101
Loading 1112 byte file host1-config:[OK]
Device#
%SYS-5-CONFIG: Configured from host1-config by rcp from 172.16.101.101
```

Copy RCP Startup-Config

The following example specifies a remote username of `netadmin1`. Then it copies the configuration file named `host2-config` from the `netadmin1` directory on the remote server with an IP address of `172.16.101.101` to the startup configuration.

```
Device# configure terminal
Device(config)# ip rcmd remote-username netadmin1
Device(config)# end
Device# copy rcp: nvram:startup-config
Address of remote host [255.255.255.255]? 172.16.101.101
Name of configuration file[rtr2-config]? host2-config
Configure using host2-config from 172.16.101.101?[confirm]
Connected to 172.16.101.101
Loading 1112 byte file host2-config:[OK]
[OK]
Device#
%SYS-5-CONFIG_NV:Non-volatile store configured from host2-config by rcp from 172.16.101.101
```

What to Do Next

After you have issued the `copy EXEC` command, you may be prompted for additional information or for confirmation of the action. The prompt displayed depends on how much information you provide in the `copy` command and the current setting of the `file prompt` global configuration command.

Copying a Configuration File from an FTP Server to the Device

To copy a configuration file from an FTP server to the running configuration or startup configuration, complete the tasks in this section:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `ip ftp username username`

4. **ip ftp password** *password*

5. **end**

6. Do one of the following:

- **copy ftp:** [[[//[username[:password]@]location] /directory] /filename]system:running-config

- **copy ftp:** [[[//[username[:password]@]location] /directory] /filename]nvram:startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	(Optional) Allows you to enter global configuration mode. This step is required only if you want to override the default remote username or password (see Steps 3 and 4).
Step 3	ip ftp username <i>username</i> Example: Device(config)# ip ftp username NetAdmin1	(Optional) Specifies the default remote username.
Step 4	ip ftp password <i>password</i> Example: Device(config)# ip ftp password adminpassword	(Optional) Specifies the default password.
Step 5	end Example: Device(config)# end	(Optional) Exits global configuration mode. This step is required only if you override the default remote username or password (see Steps 3 and 4).
Step 6	Do one of the following: <ul style="list-style-type: none"> • copy ftp: [[[//[username[:password]@]location] /directory] /filename]system:running-config • copy ftp: [[[//[username[:password]@]location] /directory] /filename]nvram:startup-config Example: Device# copy ftp:nvram:startup-config	Using FTP copies the configuration file from a network server to running memory or the startup configuration.

Examples

Copy FTP Running-Config

The following example copies a host configuration file named `host1-config` from the `netadmin1` directory on the remote server with an IP address of `172.16.101.101`, and loads and runs the commands on the device:

```
Device# copy ftp://netadmin1:mypass@172.16.101.101/host1-config system:running-config
Configure using host1-config from 172.16.101.101? [confirm]
Connected to 172.16.101.101
Loading 1112 byte file host1-config:[OK]
Device#
%SYS-5-CONFIG: Configured from host1-config by ftp from 172.16.101.101
```

Copy FTP Startup-Config

The following example specifies a remote username of `netadmin1`. Then it copies the configuration file named `host2-config` from the `netadmin1` directory on the remote server with an IP address of `172.16.101.101` to the startup configuration:

```
Device# configure terminal
Device(config)# ip ftp username netadmin1
Device(config)# ip ftp password mypass
Device(config)# end
Device# copy ftp: nvram:startup-config
Address of remote host [255.255.255.255]? 172.16.101.101
Name of configuration file[host1-config]? host2-config
Configure using host2-config from 172.16.101.101?[confirm]
Connected to 172.16.101.101
Loading 1112 byte file host2-config:[OK]
[OK]
Device#
%SYS-5-CONFIG_NV:Non-volatile store configured from host2-config by ftp from 172.16.101.101
```

What to Do Next

After you have issued the `copy EXEC` command, you may be prompted for additional information or for confirmation of the action. The prompt displayed depends on how much information you provide in the `copy` command and the current setting of the `file prompt` global configuration command.

Maintaining Configuration Files Larger than NVRAM

To maintain a configuration file that exceeds the size of NVRAM, perform the tasks described in the following sections:

Compressing the Configuration File

To compress configuration files, complete the tasks in this section:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `service compress-config`

4. **end**
5. Do one of the following:
 - Use FTP, RCP, or TFTP to copy the new configuration.
 - **configure terminal**
6. **copy system:running-config nvram:startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	service compress-config Example: Device(config)# service compress-config	Specifies that the configuration file be compressed.
Step 4	end Example: Device(config)# end	Exits global configuration mode.
Step 5	Do one of the following: <ul style="list-style-type: none"> • Use FTP, RCP, or TFTP to copy the new configuration. • configure terminal Example: Device# configure terminal	Enters the new configuration: <ul style="list-style-type: none"> • If you try to load a configuration that is more than three times larger than the NVRAM size, the following error message is displayed: “[buffer overflow - <i>file-size</i> /<i>buffer-size</i> bytes].”
Step 6	copy system:running-config nvram:startup-config Example: Device(config)# copy system:running-config nvram:startup-config	When you have finished changing the running-configuration, save the new configuration.

Examples

The following example compresses a 129-KB configuration file to 11 KB:

```

Device# configure terminal

Device(config)# service compress-config

Device(config)# end

Device# copy tftp://172.16.2.15/tokyo-config system:running-config

Configure using tokyo-config from 172.16.2.155? [confirm] y

Booting tokyo-config from 172.16.2.155:!!! [OK - 874/16000 bytes]
Device# copy system:running-config nvram:startup-config

Building configuration...
Compressing configuration from 129648 bytes to 11077 bytes
[OK]

```

Storing the Configuration in Flash Memory on Class A Flash File Systems

To store the startup configuration in flash memory, complete the tasks in this section:

SUMMARY STEPS

1. **enable**
2. **copy nvram:startup-config** *flash-filesystem:filename*
3. **configure terminal**
4. **boot config flash-filesystem:** *filename*
5. **end**
6. Do one of the following:
 - Use FTP, RCP, or TFTP to copy the new configuration. If you try to load a configuration that is more than three times larger than the NVRAM size, the following error message is displayed: “[buffer overflow - file-size /buffer-size bytes].”
 - **configure terminal**
7. **copy system:running-config nvram:startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	copy nvram:startup-config <i>flash-filesystem:filename</i> Example: Device# copy nvram:startup-config usbflash0:switch-config	Copies the current startup configuration to the new location to create the configuration file.

	Command or Action	Purpose
Step 3	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 4	boot config flash-filesystem: <i>filename</i> Example: <pre>Device(config)# boot config usbflash0:switch-config</pre>	Specifies that the startup configuration file be stored in flash memory by setting the CONFIG_FILE variable.
Step 5	end Example: <pre>Device(config)# end</pre>	Exits global configuration mode.
Step 6	Do one of the following: <ul style="list-style-type: none"> • Use FTP, RCP, or TFTP to copy the new configuration. If you try to load a configuration that is more than three times larger than the NVRAM size, the following error message is displayed: “[buffer overflow - <i>file-size</i> /<i>buffer-size</i> bytes].” • configure terminal Example: <pre>Device# configure terminal</pre>	Enters the new configuration.
Step 7	copy system:running-config nvram:startup-config Example: <pre>Device(config)# copy system:running-config nvram:startup-config</pre>	When you have finished changing the running-configuration, save the new configuration.

Examples

The following example stores the configuration file in usbflash0:

```
Device# copy nvram:startup-config usbflash0:switch-config

Device# configure terminal

Device(config)# boot config usbflash0:switch-config

Device(config)# end

Device# copy system:running-config nvram:startup-config
```

Loading the Configuration Commands from the Network

To use a network server to store large configurations, complete the tasks in this section:

SUMMARY STEPS

1. **enable**
2. **copy system:running-config {ftp: | rcp: | tftp:}**
3. **configure terminal**
4. **boot network {ftp:[[[//[username [:password]@]location]/directory]/filename] | rcp:[[[//[username@]location]/directory]/filename] | tftp:[[[//[location]/directory]/filename]}}**
5. **service config**
6. **end**
7. **copy system:running-config nvram:startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	copy system:running-config {ftp: rcp: tftp:} Example: Device# copy system:running-config ftp:	Saves the running configuration to an FTP, RCP, or TFTP server.
Step 3	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 4	boot network {ftp:[[[//[username [:password]@]location]/directory]/filename] rcp:[[[//[username@]location]/directory]/filename] tftp:[[[//[location]/directory]/filename]}} Example: Device(config)# boot network ftp://user1:guessme@example.com/dir10/file1	Specifies that the startup configuration file be loaded from the network server at startup.
Step 5	service config Example: Device(config)# service config	Enables the switch to download configuration files at system startup.
Step 6	end Example:	Exits global configuration mode.

	Command or Action	Purpose
	Device(config)# end	
Step 7	copy system:running-config nvram:startup-config Example: Device# copy system:running-config nvram:startup-config	Saves the configuration.

Copying Configuration Files from Flash Memory to the Startup or Running Configuration

To copy a configuration file from flash memory directly to your startup configuration in NVRAM or your running configuration, enter one of the commands in Step 2:

SUMMARY STEPS

1. **enable**
2. Do one of the following:
 - **copy filesystem: [partition-number:][filename] nvram:startup-config**
 - **copy filesystem: [partition-number:][filename] system:running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	Do one of the following: <ul style="list-style-type: none"> • copy filesystem: [partition-number:][filename] nvram:startup-config • copy filesystem: [partition-number:][filename] system:running-config Example: Device# copy usbflash0:4:ios-upgrade-1 nvram:startup-config	<ul style="list-style-type: none"> • Loads a configuration file directly into NVRAM or • Copies a configuration file to your running configuration

Examples

The following example copies the file named ios-upgrade-1 from partition 4 of the flash memory PC Card in usbflash0 to the device startup configurations:

```
Device# copy usbflash0:4:ios-upgrade-1 nvram:startup-config

Copy 'ios-upgrade-1' from flash device as 'startup-config' ? [yes/no] yes

[OK]
```

Copying Configuration Files Between Flash Memory File Systems

On platforms with multiple flash memory file systems, you can copy files from one flash memory file system, such as internal flash memory to another flash memory file system. Copying files to different flash memory file systems lets you create backup copies of working configurations and duplicate configurations for other devices. To copy a configuration file between flash memory file systems, use the following commands in EXEC mode:

SUMMARY STEPS

1. **enable**
2. **show** *source-filesystem:*
3. **copy** *source-filesystem: [partition-number:][filename] dest-filesystem:[partition-number:][filename]*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show <i>source-filesystem:</i> Example: Device# show flash:	Displays the layout and contents of flash memory to verify the filename.
Step 3	copy <i>source-filesystem: [partition-number:][filename] dest-filesystem:[partition-number:][filename]</i> Example: Device# copy flash: usbflash0:	Copies a configuration file between flash memory devices. <ul style="list-style-type: none"> • The source device and the destination device cannot be the same. For example, the copy usbflash0: usbflash0: command is invalid.

Example

The following example copies the file named running-config from partition 1 on internal flash memory to partition 1 of usbflash0 on a device. In this example, the source partition is not specified, so the device prompts for the partition number:

```
Device# copy flash: usbflash0:

System flash
```


	Command or Action	Purpose
Step 2	configure terminal Example: Device# configure terminal	(Optional) Enters global configuration mode. This step is required only if you override the default remote username or password (see Steps 3 and 4).
Step 3	ip ftp username <i>username</i> Example: Device(config)# ip ftp username Admin01	(Optional) Specifies the remote username.
Step 4	ip ftp password <i>password</i> Example: Device(config)# ip ftp password adminpassword	(Optional) Specifies the remote password.
Step 5	end Example: Device(config)# end	(Optional) Exits configuration mode. This step is required only if you override the default remote username (see Steps 3 and 4).
Step 6	copy ftp: [[//location]/directory] /bundle_name flash: Example: Device>copy ftp:/cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin flash:	Copies the configuration file from a network server to the flash memory device using FTP.

What to Do Next

After you have issued the **copy** EXEC command, you may be prompted for additional information or for confirmation of the action. The prompt displayed depends on how much information you provide in the **copy** command and the current setting of the **file prompt** global configuration command.

Copying a Configuration File from an RCP Server to Flash Memory Devices

To copy a configuration file from an RCP server to a flash memory device, complete the tasks in this section:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip rcmd remote-username *username***
4. **end**
5. **copy rcp: [[[[/username@]location] /directory] /bundle_name] flash:**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	(Optional) Enters global configuration mode. This step is required only if you override the default remote username or password (see Step 3).
Step 3	ip rcmd remote-username <i>username</i> Example: Device(config)# ip rcmd remote-username Admin01	(Optional) Specifies the remote username.
Step 4	end Example: Device(config)# end	(Optional) Exits configuration mode. This step is required only if you override the default remote username or password (see Step 3).
Step 5	copy rcp: [[[/[<i>username</i>@]<i>location</i>]/<i>directory</i>]/<i>bundle_name</i>] flash: Example: Device# copy rcp://netadmin@172.16.101.101/bundle1 flash:	Copies the configuration file from a network server to the flash memory device using RCP. Respond to any device prompts for additional information or confirmation. Prompting depends on how much information you provide in the copy command and the current setting of the file prompt command.

Copying a Configuration File from a TFTP Server to Flash Memory Devices

To copy a configuration file from a TFTP server to a flash memory device, complete the tasks in this section:

SUMMARY STEPS

1. **enable**
2. **copy tftp: [[[/*location*]/*directory*]/*bundle_name* flash:**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	copy tftp: [[[<i>location</i>]/ <i>directory</i>]/ <i>bundle_name</i> flash: Example: <pre>Device# copy tftp:/cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin flash:</pre>	Copies the file from a TFTP server to the flash memory device. Reply to any device prompts for additional information or confirmation. Prompting depends on how much information you provide in the copy command and the current setting of the file prompt command.

Examples

The following example shows the copying of the configuration file named switch-config from a TFTP server to the flash memory card inserted in usbflash0. The copied file is renamed new-config.

```
Device#
copy tftp:switch-config usbflash0:new-config
```

Re-executing the Configuration Commands in the Startup Configuration File

To re-execute the commands located in the startup configuration file, complete the task in this section:

SUMMARY STEPS

1. **enable**
2. **configure memory**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure memory Example: <pre>Device# configure memory</pre>	Re-executes the configuration commands located in the startup configuration file.

Clearing the Startup Configuration

You can clear the configuration information from the startup configuration. If you reboot the device with no startup configuration, the device enters the Setup command facility so that you can configure the device from scratch. To clear the contents of your startup configuration, complete the task in this section:

SUMMARY STEPS

1. **enable**
2. **erase nvram**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	erase nvram Example: Device# erase nvram	Clears the contents of your startup configuration. <p>Note For all platforms except the Class A Flash file system platforms, this command erases NVRAM. The startup configuration file cannot be restored once it has been deleted. On Class A Flash file system platforms, when you use the erase startup-config EXEC command, the device erases or deletes the configuration pointed to by the CONFIG_FILE environment variable. If this variable points to NVRAM, the device erases NVRAM. If the CONFIG_FILE environment variable specifies a flash memory device and configuration filename, the device deletes the configuration file. That is, the device marks the file as “deleted,” rather than erasing it. This feature allows you to recover a deleted file.</p>

Deleting a Specified Configuration File

To delete a specified configuration on a specific flash device, complete the task in this section:

SUMMARY STEPS

1. **enable**
2. **delete** *flash-filesystem:filename*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	delete <i>flash-filesystem:filename</i> Example: <pre>Device# delete usbflash0:myconfig</pre>	<p>Deletes the specified configuration file on the specified flash device.</p> <p>Note On Class A and B Flash file systems, when you delete a specific file in flash memory, the system marks the file as deleted, allowing you to later recover a deleted file using the undelete EXEC command. Erased files cannot be recovered. To permanently erase the configuration file, use the squeeze EXEC command. On Class C Flash file systems, you cannot recover a file that has been deleted. If you attempt to erase or delete the configuration file specified by the CONFIG_FILE environment variable, the system prompts you to confirm the deletion.</p>

Specifying the CONFIG_FILE Environment Variable on Class A Flash File Systems

On Class A flash file systems, you can configure the Cisco IOS software to load the startup configuration file specified by the CONFIG_FILE environment variable. The CONFIG_FILE variable defaults to NVRAM. To change the CONFIG_FILE environment variable, complete the tasks in this section:

SUMMARY STEPS

1. **enable**
2. **copy** [*flash-url* | *ftp-url* | *rcp-url* | *tftp-url* | **system:running-config** | **nvram:startup-config**] *dest-flash-url*
3. **configure terminal**
4. **boot config** *dest-flash-url*
5. **end**
6. **copy** **system:running-config** **nvram:startup-config**
7. **show boot**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	copy [<i>flash-url</i> <i>ftp-url</i> <i>rcp-url</i> <i>tftp-url</i> system:running-config nvram:startup-config] <i>dest-flash-url</i> Example:	<p>Copies the configuration file to the flash file system from which the device loads the file on restart.</p>

	Command or Action	Purpose
	Device# copy system:running-config nvram:startup-config	
Step 3	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 4	boot config <i>dest-flash-url</i> Example: Device(config)# boot config 172.16.1.1	Sets the CONFIG_FILE environment variable. This step modifies the runtime CONFIG_FILE environment variable.
Step 5	end Example: Device(config)# end	Exits global configuration mode.
Step 6	copy system:running-config nvram:startup-config Example: Device# copy system:running-config nvram:startup-config	Saves the configuration performed in Step 3 to the startup configuration.
Step 7	show boot Example: Device# show boot	(Optional) Allows you to verify the contents of the CONFIG_FILE environment variable.

Examples

The following example copies the running configuration file to the device. This configuration is then used as the startup configuration when the system is restarted:

```
Device# copy system:running-config usbflash0:config2
Device# configure terminal
Device(config)# boot config usbflash0:config2
Device(config)# end
Device# copy system:running-config nvram:startup-config
[ok]
Device# show boot
BOOT variable = usbflash0:rsp-boot-m
CONFIG_FILE variable = nvram:
Current CONFIG_FILE variable = usbflash0:config2
Configuration register is 0x010F
```

What to Do Next

After you specify a location for the startup configuration file, the **nvr_{am}:startup-config** command is aliased to the new location of the startup configuration file. The **more nvr_{am}:startup-config EXEC** command displays the startup configuration, regardless of its location. The **erase nvr_{am}:startup-config EXEC** command erases the contents of NVRAM and deletes the file pointed to by the CONFIG_FILE environment variable.

When you save the configuration using the **copy system:running-config nvr_{am}:startup-config** command, the device saves a complete version of the configuration file to the location specified by the CONFIG_FILE environment variable and a distilled version to NVRAM. A distilled version is one that does not contain access list information. If NVRAM contains a complete configuration file, the device prompts you to confirm your overwrite of the complete version with the distilled version. If NVRAM contains a distilled configuration, the device does not prompt you for confirmation and proceeds with overwriting the existing distilled configuration file in NVRAM.



Note If you specify a file in a flash device as the CONFIG_FILE environment variable, every time you save your configuration file with the **copy system:running-config nvr_{am}:startup-config** command, the old configuration file is marked as “deleted,” and the new configuration file is saved to that device. Eventually, Flash memory fills up as the old configuration files still take up memory. Use the **squeeze EXEC** command to permanently delete the old configuration files and reclaim the space.

Configuring the Device to Download Configuration Files

You can specify an ordered list of network configuration and host configuration filenames. The Cisco IOS XE software scans this list until it loads the appropriate network or host configuration file.

To configure the device to download configuration files at system startup, perform at least one of the tasks described in the following sections:

- [Configuring the Device to Download the Network Configuration File](#)
- [Configuring the Device to Download the Host Configuration File](#)

If the device fails to load a configuration file during startup, it tries again every 10 minutes (the default setting) until a host provides the requested files. With each failed attempt, the device displays the following message on the console terminal:

```
Booting host-config... [timed out]
```

If there are any problems with the startup configuration file, or if the configuration register is set to ignore NVRAM, the device enters the Setup command facility.

Configuring the Device to Download the Network Configuration File

To configure the Cisco IOS software to download a network configuration file from a server at startup, complete the tasks in this section:

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **boot network** {**ftp**:[[[//[username [:password]@]location]/directory]/filename] | **rcp**:[[[//[username@]location]/directory]/filename] | **tftp**:[[[//location]/directory]/filename]}
4. **service config**
5. **end**
6. **copy system:running-config nvram:startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	boot network { ftp :[[[//[username [:password]@]location]/directory]/filename] rcp :[[[//[username@]location]/directory]/filename] tftp :[[[//location]/directory]/filename]} Example: Device(config)# boot network tftp:hostfile1	Specifies the network configuration file to download at startup, and the protocol to be used (TFTP, RCP, or FTP). <ul style="list-style-type: none"> • If you do not specify a network configuration filename, the Cisco IOS software uses the default filename network-config. If you omit the address, the device uses the broadcast address. • You can specify more than one network configuration file. The software tries them in order entered until it loads one. This procedure can be useful for keeping files with different configuration information loaded on a network server.
Step 4	service config Example: Device(config)# service config	Enables the system to automatically load the network file on restart.
Step 5	end Example: Device(config)# end	Exits global configuration mode.
Step 6	copy system:running-config nvram:startup-config Example: Device# copy system:running-config nvram:startup-config	Saves the running configuration to the startup configuration file.

Configuring the Device to Download the Host Configuration File

To configure the Cisco IOS software to download a host configuration file from a server at startup, complete the tasks in this section:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **boot host** {**ftp**:[[[//*username* [*:password*]@]*location*]/*directory*]/*filename*] | **rcp**:[[[//*username*@]*location*]/*directory*]/*filename*] | **tftp**:[[[//*location*]/*directory*]/*filename*] }
4. **service config**
5. **end**
6. **copy system:running-config nvram:startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	boot host { ftp :[[[// <i>username</i> [<i>:password</i>]@] <i>location</i>]/ <i>directory</i>]/ <i>filename</i>] rcp :[[[// <i>username</i> @] <i>location</i>]/ <i>directory</i>]/ <i>filename</i>] tftp :[[[// <i>location</i>]/ <i>directory</i>]/ <i>filename</i>] } Example: Device(config)# boot host tftp:hostfile1	Specifies the host configuration file to download at startup, and the protocol to be used (FTP, RCP, or TFTP): <ul style="list-style-type: none"> • If you do not specify a host configuration filename, the device uses its own name to form a host configuration filename by converting the name to all lowercase letters, removing all domain information, and appending “-config.” If no host name information is available, the software uses the default host configuration filename device-config. If you omit the address, the device uses the broadcast address. • You can specify more than one host configuration file. The Cisco IOS software tries them in order entered until it loads one. This procedure can be useful for keeping files with different configuration information loaded on a network server.
Step 4	service config Example: Device(config)# service config	Enables the system to automatically load the host file upon restart.

	Command or Action	Purpose
Step 5	end Example: <pre>Device(config)# end</pre>	Exits global configuration mode.
Step 6	copy system:running-config nvram:startup-config Example: <pre>Device# copy system:running-config nvram:startup-config</pre>	Saves the running configuration to the startup configuration file.

Example

In the following example, a device is configured to download the host configuration file named `hostfile1` and the network configuration file named `networkfile1`. The device uses TFTP and the broadcast address to obtain the file:

```
Device# configure terminal
Device(config)# boot host tftp:hostfile1
Device(config)# boot network tftp:networkfile1
Device(config)# service config
Device(config)# end
Device# copy system:running-config nvram:startup-config
```

Feature History for Managing Configuration Files

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Everest 16.5.1a	Managing Configuration Files	Configuration files contain the Cisco IOS software commands used to customize the functionality of your Cisco device. Commands are parsed (translated and executed) by the Cisco IOS software when the system is booted (from the startup-config file) or when you enter commands at the CLI in a configuration mode.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 10

Configuration Replace and Configuration Rollback

- [Prerequisites for Configuration Replace and Configuration Rollback, on page 237](#)
- [Restrictions for Configuration Replace and Configuration Rollback, on page 238](#)
- [Information About Configuration Replace and Configuration Rollback, on page 238](#)
- [How to Use Configuration Replace and Configuration Rollback, on page 241](#)
- [Configuration Examples for Configuration Replace and Configuration Rollback, on page 247](#)
- [Feature History for Configuration Replace and Configuration Rollback, on page 249](#)

Prerequisites for Configuration Replace and Configuration Rollback

The format of the configuration files used as input by the Configuration Replace and Configuration Rollback feature must comply with standard Cisco software configuration file indentation rules as follows:

- Start all commands on a new line with no indentation, unless the command is within a configuration submode.
- Indent commands within a first-level configuration submode one space.
- Indent commands within a second-level configuration submode two spaces.
- Indent commands within subsequent submodes accordingly.

These indentation rules describe how the software creates configuration files for such commands as **show running-config** or **copy running-config destination-url**. Any configuration file generated on a Cisco device complies with these rules.

Free memory larger than the combined size of the two configuration files (the current running configuration and the saved replacement configuration) is required.

Restrictions for Configuration Replace and Configuration Rollback

If the device does not have free memory larger than the combined size of the two configuration files (the current running configuration and the saved replacement configuration), the configuration replace operation is not performed.

Certain Cisco configuration commands such as those pertaining to physical components of a networking device (for example, physical interfaces) cannot be added or removed from the running configuration. For example, a configuration replace operation cannot remove the **interface ethernet 0** command line from the current running configuration if that interface is physically present on the device. Similarly, the **interface ethernet 1** command line cannot be added to the running configuration if no such interface is physically present on the device. A configuration replace operation that attempts to perform these types of changes results in error messages indicating that these specific command lines failed.

In very rare cases, certain Cisco configuration commands cannot be removed from the running configuration without reloading the device. A configuration replace operation that attempts to remove this type of command results in error messages indicating that these specific command lines failed.

Information About Configuration Replace and Configuration Rollback

Configuration Archive

The Cisco IOS configuration archive is intended to provide a mechanism to store, organize, and manage an archive of Cisco IOS configuration files to enhance the configuration rollback capability provided by the **configure replace** command. Before this feature was introduced, you could save copies of the running configuration using the **copy running-config destination-url** command, storing the replacement file either locally or remotely. However, this method lacked any automated file management. On the other hand, the Configuration Replace and Configuration Rollback feature provides the capability to automatically save copies of the running configuration to the Cisco IOS configuration archive. These archived files serve as checkpoint configuration references and can be used by the **configure replace** command to revert to previous configuration states.

The **archive config** command allows you to save Cisco IOS configurations in the configuration archive using a standard location and filename prefix that is automatically appended with an incremental version number (and optional timestamp) as each consecutive file is saved. This functionality provides a means for consistent identification of saved Cisco IOS configuration files. You can specify how many versions of the running configuration are kept in the archive. After the maximum number of files are saved in the archive, the oldest file is automatically deleted when the next, most recent file is saved. The **show archive** command displays information for all configuration files saved in the Cisco IOS configuration archive.

The Cisco IOS configuration archive, in which the configuration files are stored and available for use with the **configure replace** command, can be located on the following file systems: FTP, HTTP, RCP, TFTP.

Configuration Replace

The **configure replace** privileged EXEC command provides the capability to replace the current running configuration with any saved Cisco IOS configuration file. This functionality can be used to revert to a previous configuration state, effectively rolling back any configuration changes that were made since the previous configuration state was saved.

When using the **configure replace** command, you must specify a saved Cisco IOS configuration as the replacement configuration file for the current running configuration. The replacement file must be a complete configuration generated by a Cisco IOS device (for example, a configuration generated by the **copy running-config destination-url** command), or, if generated externally, the replacement file must comply with the format of files generated by Cisco IOS devices. When the **configure replace** command is entered, the current running configuration is compared with the specified replacement configuration and a set of diffs is generated. The algorithm used to compare the two files is the same as that employed by the **show archive config differences** command. The resulting diffs are then applied by the Cisco IOS parser to achieve the replacement configuration state. Only the diffs are applied, avoiding potential service disruption from reapplying configuration commands that already exist in the current running configuration. This algorithm effectively handles configuration changes to order-dependent commands (such as access lists) through a multiple pass process. Under normal circumstances, no more than three passes are needed to complete a configuration replace operation, and a limit of five passes is performed to preclude any looping behavior.

The Cisco IOS **copy source-url running-config** privileged EXEC command is often used to copy a stored Cisco IOS configuration file to the running configuration. When using the **copy source-url running-config** command as an alternative to the **configure replace target-url** privileged EXEC command, the following major differences should be noted:

- The **copy source-url running-config** command is a merge operation and preserves all of the commands from both the source file and the current running configuration. This command does not remove commands from the current running configuration that are not present in the source file. In contrast, the **configure replace target-url** command removes commands from the current running configuration that are not present in the replacement file and adds commands to the current running configuration that need to be added.
- The **copy source-url running-config** command applies every command in the source file, whether or not the command is already present in the current running configuration. This algorithm is inefficient and, in some cases, can result in service outages. In contrast, the **configure replace target-url** command only applies the commands that need to be applied—no existing commands in the current running configuration are reapplied.
- A partial configuration file may be used as the source file for the **copy source-url running-config** command, whereas a complete Cisco IOS configuration file must be used as the replacement file for the **configure replace target-url** command.

A locking feature for the configuration replace operation was introduced. When the **configure replace** command is used, the running configuration file is locked by default for the duration of the configuration replace operation. This locking mechanism prevents other users from changing the running configuration while the replacement operation is taking place, which might otherwise cause the replacement operation to terminate unsuccessfully. You can disable the locking of the running configuration by using the **no lock** keyword when issuing the **configure replace** command.

The running configuration lock is automatically cleared at the end of the configuration replace operation. You can display any locks that may be currently applied to the running configuration using the **show configuration lock** command.

Configuration Rollback

The concept of rollback comes from the transactional processing model common to database operations. In a database transaction, you might make a set of changes to a given database table. You then must choose whether to commit the changes (apply the changes permanently) or to roll back the changes (discard the changes and revert to the previous state of the table). In this context, rollback means that a journal file containing a log of the changes is discarded, and no changes are applied. The result of the rollback operation is to revert to the previous state, before any changes were applied.

The **configure replace** command allows you to revert to a previous configuration state, effectively rolling back changes that were made since the previous configuration state was saved. Instead of basing the rollback operation on a specific set of changes that were applied, the Cisco IOS configuration rollback capability uses the concept of reverting to a specific configuration state based on a saved Cisco IOS configuration file. This concept is similar to the database idea of saving a checkpoint (a saved version of the database) to preserve a specific state.

If the configuration rollback capability is desired, you must save the Cisco IOS running configuration before making any configuration changes. Then, after entering configuration changes, you can use that saved configuration file to roll back the changes (using the **configure replace target-url** command). Furthermore, because you can specify any saved Cisco IOS configuration file as the replacement configuration, you are not limited to a fixed number of rollbacks, as is the case in some rollback models.

Configuration Rollback Confirmed Change

The Configuration Rollback Confirmed Change feature allows configuration changes to be performed with an optional requirement that they be confirmed. If this confirmation is not received, the configuration is returned to the state prior to the changes being applied. The mechanism provides a safeguard against inadvertent loss of connectivity between a network device and the user or management application due to configuration changes.

Benefits of Configuration Replace and Configuration Rollback

- Allows you to revert to a previous configuration state, effectively rolling back configuration changes.
- Allows you to replace the current running configuration file with the startup configuration file without having to reload the device or manually undo CLI changes to the running configuration file, therefore reducing system downtime.
- Allows you to revert to any saved Cisco IOS configuration state.
- Simplifies configuration changes by allowing you to apply a complete configuration file to the device, where only the commands that need to be added or removed are affected.
- When using the **configure replace** command as an alternative to the **copy source-url running-config** command, increases efficiency and prevents risk of service outages by not reapplying existing commands in the current running configuration.

How to Use Configuration Replace and Configuration Rollback

Creating a Configuration Archive

No prerequisite configuration is needed to use the **configure replace** command. Using the **configure replace** command in conjunction with the Cisco IOS configuration archive and the **archive config** command is optional but offers significant benefit for configuration rollback scenarios. Before using the **archive config** command, the configuration archive must be configured. Perform this task to configure the characteristics of the configuration archive.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **archive**
4. **path** *url*
5. **maximum** *number*
6. **time-period** *minutes*
7. **end**
8. **archive config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	archive Example: Device(config)# archive	Enters archive configuration mode.
Step 4	path <i>url</i> Example: Device(config-archive)# path flash:myconfiguration	Specifies the location and filename prefix for the files in the Cisco IOS configuration archive. Note If a directory is specified in the path instead of file, the directory name must be followed by a forward slash as follows: path flash:/directory/. The forward slash is not necessary after a filename; it is only necessary when specifying a directory.

	Command or Action	Purpose
Step 5	maximum <i>number</i> Example: <pre>Device(config-archive)# maximum 14</pre>	(Optional) Sets the maximum number of archive files of the running configuration to be saved in the Cisco IOS configuration archive. <ul style="list-style-type: none"> The <i>number</i> argument is the maximum number of archive files of the running configuration to be saved in the Cisco IOS configuration archive. Valid values are from 1 to 14. The default is 10. Note Before using this command, you must configure the path command to specify the location and filename prefix for the files in the Cisco IOS configuration archive.
Step 6	time-period <i>minutes</i> Example: <pre>Device(config-archive)# time-period 1440</pre>	(Optional) Sets the time increment for automatically saving an archive file of the current running configuration in the Cisco IOS configuration archive. <ul style="list-style-type: none"> The <i>minutes</i> argument specifies how often, in minutes, to automatically save an archive file of the current running configuration in the Cisco IOS configuration archive. Note Before using this command, you must configure the path command to specify the location and filename prefix for the files in the Cisco IOS configuration archive.
Step 7	end Example: <pre>Device(config-archive)# end</pre>	Exits to privileged EXEC mode.
Step 8	archive config Example: <pre>Device# archive config</pre>	Saves the current running configuration file to the configuration archive. Note The path command must be configured before using this command.

Performing a Configuration Replace or Configuration Rollback Operation

Perform this task to replace the current running configuration file with a saved Cisco IOS configuration file.



Note You must create a configuration archive before performing this procedure. See [Creating a Configuration Archive](#) for detailed steps. The following procedure details how to return to that archived configuration in the event of a problem with the current running configuration.

SUMMARY STEPS

1. **enable**
2. **configure replace** *target-url* [**nolock**] [**list**] [**force**] [**ignore case**] [**revert trigger** [**error**]] [**timer** *minutes*] | **time** *minutes*]]
3. **configure revert** { **now** | **timer** { *minutes* | **idle** *minutes* } }
4. **configure confirm**
5. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure replace <i>target-url</i> [nolock] [list] [force] [ignore case] [revert trigger [error]] [timer <i>minutes</i>] time <i>minutes</i>]]</p> <p>Example:</p> <pre>Device# configure replace flash: startup-config time 120</pre>	<p>Replaces the current running configuration file with a saved Cisco IOS configuration file.</p> <ul style="list-style-type: none"> • The <i>target - url</i> argument is a URL (accessible by the Cisco IOS file system) of the saved Cisco IOS configuration file that is to replace the current running configuration, such as the configuration file created using the archive config command. • The list keyword displays a list of the command lines applied by the Cisco IOS software parser during each pass of the configuration replace operation. The total number of passes performed is also displayed. • The force keyword replaces the current running configuration file with the specified saved Cisco IOS configuration file without prompting you for confirmation. • The time <i>minutes</i> keyword and argument specify the time (in minutes) within which you must enter the configure confirm command to confirm replacement of the current running configuration file. If the configure confirm command is not entered within the specified time limit, the configuration replace operation is automatically reversed (in other words, the current running configuration file is restored to the configuration state that existed prior to entering the configure replace command). • The nolock keyword disables the locking of the running configuration file that prevents other users from changing the running configuration during a configuration replace operation.

	Command or Action	Purpose
		<ul style="list-style-type: none"> The revert trigger keywords set the following triggers for reverting to the original configuration: <ul style="list-style-type: none"> error —Reverts to the original configuration upon error. timer minutes —Reverts to the original configuration if specified time elapses. The ignore case keyword allows the configuration to ignore the case of the confirmation command.
Step 3	configure revert { now timer { <i>minutes</i> idle <i>minutes</i> } } Example: <pre>Device# configure revert now</pre>	(Optional) To cancel the timed rollback and trigger the rollback immediately, or to reset parameters for the timed rollback, use the configure revert command in privileged EXEC mode. <ul style="list-style-type: none"> now —Triggers the rollback immediately. timer —Resets the configuration revert timer. <ul style="list-style-type: none"> Use the <i>minutes</i> argument with the timer keyword to specify a new revert time in minutes. Use the idle keyword along with a time in minutes to set the maximum allowable time period of no activity before reverting to the saved configuration.
Step 4	configure confirm Example: <pre>Device# configure confirm</pre>	(Optional) Confirms replacement of the current running configuration file with a saved Cisco IOS configuration file. Note Use this command only if the time seconds keyword and argument of the configure replace command are specified.
Step 5	exit Example: <pre>Device# exit</pre>	Exits to user EXEC mode.

Monitoring and Troubleshooting the Feature

Perform this task to monitor and troubleshoot the Configuration Replace and Configuration Rollback feature.

SUMMARY STEPS

1. **enable**
2. **show archive**

3. **debug archive versioning**
4. **debug archive config timestamp**
5. **exit**

DETAILED STEPS

Step 1 **enable**

Use this command to enable privileged EXEC mode. Enter your password if prompted.

Example:

```
Device> enable
Device#
```

Step 2 **show archive**

Use this command to display information about the files saved in the Cisco IOS configuration archive.

Example:

```
Device# show archive
There are currently 1 archive configurations saved.
The next archive file will be named flash:myconfiguration-2
Archive #  Name
0
1      flash:myconfiguration-1 <- Most Recent
2
3
4
5
6
7
8
9
10
11
12
13
14
```

The following is sample output from the **show archive** command after several archive files of the running configuration have been saved. In this example, the maximum number of archive files to be saved is set to three.

Example:

```
Device# show archive
There are currently 3 archive configurations saved.
The next archive file will be named flash:myconfiguration-8
Archive #  Name
0
1      :Deleted
2      :Deleted
3      :Deleted
4      :Deleted
5      flash:myconfiguration-5
6      flash:myconfiguration-6
7      flash:myconfiguration-7 <- Most Recent
8
```

```

9
10
11
12
13
14

```

Step 3 debug archive versioning

Use this command to enable debugging of the Cisco IOS configuration archive activities to help monitor and troubleshoot configuration replace and rollback.

Example:

```

Device# debug archive versioning
Jan  9 06:46:28.419:backup_running_config
Jan  9 06:46:28.419:Current = 7
Jan  9 06:46:28.443:Writing backup file flash:myconfiguration-7
Jan  9 06:46:29.547: backup worked

```

Step 4 debug archive config timestamp

Use this command to enable debugging of the processing time for each integral step of a configuration replace operation and the size of the configuration files being handled.

Example:

```

Device# debug archive config timestamp
Device# configure replace flash:myconfiguration force
Timing Debug Statistics for IOS Config Replace operation:
  Time to read file usbflash0:sample_2.cfg = 0 msec (0 sec)
  Number of lines read:55
  Size of file          :1054
Starting Pass 1
  Time to read file system:running-config = 0 msec (0 sec)
  Number of lines read:93
  Size of file          :2539
  Time taken for positive rollback pass = 320 msec (0 sec)
  Time taken for negative rollback pass = 0 msec (0 sec)
  Time taken for negative incremental diffs pass = 59 msec (0 sec)
  Time taken by PI to apply changes = 0 msec (0 sec)
  Time taken for Pass 1 = 380 msec (0 sec)
Starting Pass 2
  Time to read file system:running-config = 0 msec (0 sec)
  Number of lines read:55
  Size of file          :1054
  Time taken for positive rollback pass = 0 msec (0 sec)
  Time taken for negative rollback pass = 0 msec (0 sec)
  Time taken for Pass 2 = 0 msec (0 sec)
Total number of passes:1
Rollback Done

```

Step 5 exit

Use this command to exit to user EXEC mode.

Example:

```

Device# exit
Device>

```

Configuration Examples for Configuration Replace and Configuration Rollback

Creating a Configuration Archive

The following example shows how to perform the initial configuration of the Cisco IOS configuration archive. In this example, flash:myconfiguration is specified as the location and filename prefix for the files in the configuration archive and a value of 10 is set as the maximum number of archive files to be saved.

```
configure terminal
!
archive
 path flash:myconfiguration
 maximum 10
end
```

Replacing the Current Running Configuration with a Saved Cisco IOS Configuration File

The following example shows how to replace the current running configuration with a saved Cisco IOS configuration file named flash:myconfiguration. The **configure replace** command interactively prompts you to confirm the operation.

```
Device# configure replace flash:myconfiguration
This will apply all necessary additions and deletions
to replace the current running configuration with the
contents of the specified configuration file, which is
assumed to be a complete configuration, not a partial
configuration. Enter Y if you are sure you want to proceed. ? [no]: Y
Total number of passes: 1
Rollback Done
```

In the following example, the **list** keyword is specified in order to display the command lines that were applied during the configuration replace operation:

```
Device# configure replace flash:myconfiguration list
This will apply all necessary additions and deletions
to replace the current running configuration with the
contents of the specified configuration file, which is
assumed to be a complete configuration, not a partial
configuration. Enter Y if you are sure you want to proceed. ? [no]: Y
!Pass 1
!List of Commands:
no snmp-server community public ro
snmp-server community mystring ro

end
Total number of passes: 1
Rollback Done
```

Reverting to the Startup Configuration File

The following example shows how to revert to the Cisco IOS startup configuration file using the **configure replace** command. This example also shows the use of the optional **force** keyword to override the interactive user prompt:

```
Device# configure replace flash:startup-config force
Total number of passes: 1
Rollback Done
```

Performing a Configuration Replace Operation with the **configure confirm** Command

The following example shows the use of the **configure replace** command with the **time minutes** keyword and argument. You must enter the **configure confirm** command within the specified time limit to confirm replacement of the current running configuration file. If the **configure confirm** command is not entered within the specified time limit, the configuration replace operation is automatically reversed (in other words, the current running configuration file is restored to the configuration state that existed prior to entering the **configure replace** command).

```
Device# configure replace flash:startup-config time 120
This will apply all necessary additions and deletions
to replace the current running configuration with the
contents of the specified configuration file, which is
assumed to be a complete configuration, not a partial
configuration. Enter Y if you are sure you want to proceed. ? [no]: Y
Total number of passes: 1
Rollback Done
Device# configure confirm
```

The following example shows the use of the **configure revert** command with the **timer** keyword. You must enter the **configure revert** command to cancel the timed rollback and trigger the rollback immediately, or to reset parameters for the timed rollback.

```
Device# configure revert timer 100
```

Performing a Configuration Rollback Operation

The following example shows how to make changes to the current running configuration and then roll back the changes. As part of the configuration rollback operation, you must save the current running configuration before making changes to the file. In this example, the **archive config** command is used to save the current running configuration. The generated output of the **configure replace** command indicates that only one pass was performed to complete the rollback operation.



Note Before using the **archive config** command, you must configure the **path** command to specify the location and filename prefix for the files in the Cisco IOS configuration archive.

You first save the current running configuration in the configuration archive as follows:

```
archive config
```

You then enter configuration changes as shown in the following example:

```
configure terminal
!
user netops2 password rain
user netops3 password snow
exit
```

After having made changes to the running configuration file, assume you now want to roll back these changes and revert to the configuration that existed before the changes were made. The **show archive** command is used to verify the version of the configuration to be used as a replacement file. The **configure replace** command is then used to revert to the replacement configuration file as shown in the following example:

```
Device# show archive
There are currently 1 archive configurations saved.
The next archive file will be named flash:myconfiguration-2
Archive # Name
  0
  1      flash:myconfiguration-1 <- Most Recent
  2
  3
  4
  5
  6
  7
  8
  9
 10
Device# configure replace flash:myconfiguration-1
Total number of passes: 1
Rollback Done
```

Feature History for Configuration Replace and Configuration Rollback

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Everest 16.5.1a	Configuration Replace and Configuration Rollback	The Cisco IOS configuration archive is intended to provide a mechanism to store, organize, and manage an archive of Cisco IOS configuration files to enhance the configuration rollback capability provided by the configure replace command.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 11

Software Maintenance Upgrade

The Software Maintenance Upgrade (SMU) is a package that can be installed on a system to provide a fix or a security resolution to a released image.

- [Restrictions for Software Maintenance Upgrade, on page 251](#)
- [Information About Software Maintenance Upgrade, on page 251](#)
- [How to Manage Software Maintenance Updates, on page 252](#)
- [Configuration Examples for Software Maintenance Upgrade, on page 254](#)
- [Additional References for Software Maintenance Upgrade, on page 258](#)
- [Feature History for Software Maintenance Upgrade, on page 258](#)

Restrictions for Software Maintenance Upgrade

- SMU supports patching using install mode only.

Information About Software Maintenance Upgrade

The following sections provide detailed information about Software Maintenance Upgrade (SMU).

SMU Overview

The SMU is a package that can be installed on a system to provide a fix or a security resolution to a released image. An SMU package is provided on a per release and per component basis.

An SMU provides a significant benefit over classic Cisco IOS software because it allows you to address network issues quickly while reducing the time and scope of the testing required. The Cisco IOS XE platform internally validates SMU compatibility and does not allow you to install noncompatible SMUs.

All the SMUs are integrated into the subsequent Cisco IOS XE software maintenance releases. An SMU is an independent and self-sufficient package and it does not have any prerequisites or dependencies. You can choose which SMUs to install or uninstall in any order.

SMUs are supported only on Extended Maintenance releases and for the full lifecycle of the underlying software release.

Perform these basic steps to install an SMU:

1. Add the SMU to the filesystem.
2. Activate the SMU on the system.
3. Commit the SMU changes so that it is persistent across reloads.

SMU Workflow

The SMU process is initiated with a request to the Cisco Customer Support. Contact your customer support to raise an SMU request.

At release time, the SMU package is posted to the [Cisco Software Download](#) page and can be downloaded and installed.

SMU Package

The SMU package contains a small set of files for patching the release along with metadata that describes the contents of the package, and fix for the reported issue that the SMU is requested for. The SMU package also supports patching of the public key infrastructure (PKI) component.

SMU Reload

The SMU type describes the effect the installed SMU has on the corresponding system. SMUs might not have an impact on traffic, or might result in device restart, reload, or switchover. Run the **show install package flash: filename** command to verify whether a reload is required or not.

Hot patching enables SMU to take effect after activation without the system having to be reloaded. After the SMU is committed, the changes are persistent across reloads. In certain cases, SMUs may require a cold (complete) reload of the operating system. This action affects the traffic flow for the duration of the reload. If a cold reload is required, users will be prompted to confirm the action.

How to Manage Software Maintenance Updates

The following sections provide information about managing SMUs.

You can install, activate, and commit an SMU package using a single command or using separate commands

Installing an SMU Package

This task shows how to use the **install add file activate commit** command for installing an SMU package.

SUMMARY STEPS

1. **enable**
2. **install add file flash: filename [activate commit]**
3. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	install add file flash: <i>filename</i> [activate commit] Example: Device# install add file flash:cat9k_iosxe.BLD_SMU_20180302_085005_ TWIG_LATEST_20180306_013805.3.SSA.smu.bin activate commit	Copies the maintenance update package from a remote location (through FTP, HTTP, HTTPS, or TFTP) to the device, performs a compatibility check for the platform and image versions, activates the SMU package, and makes the package persistent across reloads. This command extracts the individual components of the .bin file into the subpackages and packages.conf files. Note If the SMU file is copied using tftp, use bootflash to activate the SMU.
Step 3	exit Example: Device# exit	Exits privileged EXEC mode and returns to user EXEC mode.

Managing an SMU Package

SUMMARY STEPS

1. enable
2. install add file flash: *filename*
3. install activate file flash: *filename*
4. install commit
5. install rollback to {base | committed | id *commit-ID*}
6. install deactivate file flash: *filename*
7. install remove {file flash: *filename* | inactive}
8. show version
9. show install summary

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	install add file flash: <i>filename</i> Example:	Copies the SMU package from a source location to the device (in case source location is remote), and then performs a compatibility check for the platform and image versions,

	Command or Action	Purpose
	<pre>Device# install add file flash:cat9k_iosxe.BLD_SMU_20180302_085005_ TWIG_LATEST_20180306_013805.3.SSA.smu.bin</pre>	and adds the SMU package on all member nodes or FRUs, as applicable. This command also runs base compatibility checks on a file to ensure that the SMU package is supported on the platform. It also adds an entry in the <code>package/SMU.sta</code> file, so that its status can be monitored and maintained.
Step 3	<p>install activate file flash: <i>filename</i></p> <p>Example:</p> <pre>Device# install activate add file flash:cat9k_iosxe.BLD_SMU_20180302_085005_ TWIG_LATEST_20180306_013805.3.SSA.smu.bin</pre>	Runs compatibility checks, installs the package, and updates the package status details.
Step 4	<p>install commit</p> <p>Example:</p> <pre>Device# install commit</pre>	Commits the activation changes to be persistent across reloads. The commit can be done after activation when the system is up, or after the first reload. If a package is activated, but not committed, it remains active after the first reload, but not after the second reload.
Step 5	<p>install rollback to {base committed id commit-ID}</p> <p>Example:</p> <pre>Device# install rollback to committed</pre>	Returns the device to the previous installation state.
Step 6	<p>install deactivate file flash: <i>filename</i></p> <p>Example:</p> <pre>Device# install deactivate file flash:cat9k_iosxe.BLD_SMU_20180302_085005_ TWIG_LATEST_20180306_013805.3.SSA.smu.bin</pre>	Deactivates an active package and updates the package status.
Step 7	<p>install remove {file flash: filename inactive}</p> <p>Example:</p> <pre>Device# install remove file flash:cat9k_iosxe.BLD_SMU_20180302_085005_ TWIG_LATEST_20180306_013805.3.SSA.smu.bin</pre>	Verifies if the specified SMU is inactive and if it is, deletes it from the file system. The inactive option deletes all the inactive packages from the file system.
Step 8	<p>show version</p> <p>Example:</p> <pre>Device# show version</pre>	Displays the image version on the device.
Step 9	<p>show install summary</p> <p>Example:</p> <pre>Device# show install summary</pre>	Displays information about the installation status of packages. The output of this command varies according to the install commands that are configured.

Configuration Examples for Software Maintenance Upgrade

The following is a list of SMU configuration examples.

The following example shows how to copy an SMU file to TFTP:

```
Device# copy
tftp://172.19.1.250//auto/ftpboot/user/cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin

tftp:Destination filename [cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin]?

Accessing
tftp://172.19.1.250//auto/tftpboot/folder1/cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin...
Loading /auto/tftpboot/folder1/cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin from
172.19.1.250 (via GigabitEthernet0): !
[OK - 17668 bytes]
17668 bytes copied in 0.058 secs (304621 bytes/sec)
```

The following is sample output from the **show install summary** command:

```
Device# show install summary

[ Switch 1 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type St Filename/Version
-----
SMU C flash:cat9k_lite_iosxe.16.09.04.CSCvk70181.SPA.smu.bin
IMG C 16.6.3.0
```

The following example shows how to add a maintenance update package file:

```
Device# install add file flash:cat9k-
universalk9.2017-01-10_13.15.1.CSCxxxxxxx.SSA.dmp.bin

install_add: START Sat Feb 26 14:06:04 PST 2017
SUCCESS: install_add flash:cat9k_iosxe.2017-01-10_13.15.1.CSCxxxxxxx.SSA.dmp.bin
Sat Feb 26 14:06:12 PST 2017
Device#
```

The following is sample output from the **show install summary** command after adding an SMU package file to the device:

```
Device# show install summary

Active Packages:
No packages
Inactive Packages:
flash:cat9k-universalk9.2017-01-10_13.15.1.CSCxxxxxxx.SSA.dmp.bin
Committed Packages:
No packages
Uncommitted Packages:
No packages
Device#
```

The following example shows how to activate an added SMU package file:

```
Device# install activate file flash:cat9k_iosxe.2017-01-10_13.15.1.
CSCxxxxxxx.SSA.dmp.bin

install_activate: START Sat Feb 26 14:10:55 PST 2017
The activation step would require a reload. Do you want to proceed? [y/n]y
Regular SMU. Reloading the box to complete activation of the SMU...
```

```
Feb 26 14:11:23.873 R0/0: %PMAN-5-EXITACTION: Process manager is exiting:
reload action requested
Initializing Hardware ...
Checking for PCIe device presence...done
System integrity status: 0x610
Rom image verified correctly
<after reload>
Device#
```

The following sample output from the **show version** command:

```
Device# show version

Cisco IOS XE Software, Version BLD_POLARIS_DEV_SMU_LATEST_20170110_13.15.1 -
SMU-PATCHED
Cisco IOS Software [Everest], Catalyst L3 Switch Software (CAT9K_IOSXE), Experimental Version

16.6.20170110_13.15.1 [BLD_V166_SMU_LATEST_20170127_13.15.1 SMU-PATCHED]
Copyright (c) 1986-2017 by Cisco Systems, Inc.
Compiled Sat 26-Feb-17 16:07 by mcpre
...
```

The following sample output from the **show install summary** command displays the status of the model package as active and uncommitted:

```
Device# show install summary

Active Packages:
  flash:cat9k-universalk9.2017-01-10_13.15.1.CSCxxx.SSA.dmp.bin
Inactive Packages:
  No packages
Committed Packages:
  flash:cat9k-universalk9.2017-01-10_13.15.1.CSCxxx.SSA.dmp.bin
Uncommitted Packages:
  No packages
Device#
```

The following is sample output from the **show install active** command:

```
Device# show install active

Active Packages:
flash:cat9k-universalk9.2017-01-10_13.15.1.CSCxxx.SSA.dmp.bin
```

The following example shows how to execute the **install commit** command:

```
Device# install commit

install_commit: START Sat Feb 26 06:46:48 UTC 2017
SUCCESS: install_commit Sat Feb 26 06:46:52 UTC 2017
Device#
```

The following sample output from the **show install summary** command displays that the update package is now committed, and that it will be persistent across reloads:

```
Device# show install summary

Active Packages:
flash:cat9k-universalk9.2017-01-10_13.15.1.CSCxxx.SSA.dmp.bin
Inactive Packages:
No packages
```

```
Committed Packages:
flash:cat9k-universalk9.2017-01-10_13.15.1.CSCxxx.SSA.dmp.bin
Uncommitted Packages:
No packages
Device#
```

The following example shows how to rollback an update package to the committed package:

```
Device# install rollback to base

install_rollback: START Sat Feb 26 11:27:41 PST 2017
This rollback would require a reload. Do you want to proceed? [y/n]y
2 install_rollback: Reloading the box to take effect

Initializing Hardware ...
<after reload>
Device#
```

The following is sample output from the **show install summary** command:

```
Device# show install summary

Active Packages:
flash:cat9k-universalk9.2017-01-10_13.15.1.CSCxxx.SSA.dmp.bin
Inactive Packages:
No packages
Committed Packages:
flash:cat9k-universalk9.2017-01-10_13.15.1.CSCxxxxxxx.SSA.dmp.bin
Uncommitted Packages:
No packages
Device#
```

The following is sample output from the **show install log** command:

```
Device# show install log

[0|install_op_boot]: START Sat Feb 26 19:31:50 Universal 2017
[0|install_op_boot]: END SUCCESS Sat Feb 26 19:31:56 Universal 2017
```

The following example shows how to deactivate an SMU package file:

```
Device# install deactivate file flash:cat9k-
universalk9.2017-01-10_13.15.1.CSCxxxxxxx.SSA.dmp.bin

install_deactivate: START Sat Feb 26 10:49:07 PST 2017
The activation step would require a reload. Do you want to proceed? [y/n]y
Regular SMU. Reloading the box to complete activation of the SMU...

Initializing Hardware...
...
<after reload>
Device#
```

The following is sample output from the **show install summary** command:

```
Device# show install summary

Active Packages:
No packages
Inactive Packages: flash:cat9k-universalk9.2017-01-10_13.15.1.CSCxxxxxxx.SSA.dmp.bin
```

```
Committed Packages:
No packages
Uncommitted Packages:
No packages
Device#
```

The following example shows how to remove an SMU from the device:

```
Device# install remove file flash:cat9k-
universalk9.2017-01-10_13.15.1.CSCxxxxxxx.SSA.dmp.bin

install_remove: START Sat Feb 26 12:09:43 PST 2017
SUCCESS: install_remove /flash/cat9k-universalk9.2017-01-10_13.15.1.
CSCxxxxxxx.SSA.dmp.bin Sat Feb 26 12:09:49 PST 2017
Device#
```

The following is sample output from the **show install summary** command:

```
Device# show install summary

Active Packages:
No packages
Inactive Packages:
No packages
Committed Packages:
No packages
Uncommitted Packages:
No packages
```

Additional References for Software Maintenance Upgrade

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	<i>Command Reference (Catalyst 9300 Series Switches)</i>

Feature History for Software Maintenance Upgrade

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Everest 16.6.1	Software Maintenance Upgrade (SMU)	An SMU is a package that can be installed on a system to provide a fix or a security resolution to a released image.

Release	Feature	Feature Information
Cisco IOS XE Fuji 16.9.1	Hot patching	Hot patching enables SMU to take effect after activation without the system having to be reloaded.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 12

Working with the Flash File System

- Finding Feature Information, on page 261
- Information About the Flash File System, on page 261
- Displaying Available File Systems, on page 262
- Setting the Default File System, on page 264
- Displaying Information About Files on a File System, on page 264
- Changing Directories and Displaying the Working Directory , on page 266
- Creating Directories , on page 266
- Copying Files, on page 267
- Creating, Displaying and Extracting Files , on page 270
- Additional References for Flash File System, on page 272
- Feature History for Flash File System, on page 273

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and 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 feature information table at the end of this module.

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 the Flash File System

The flash file system is a single flash device on which you can store files. It also provides several commands to help you manage software bundles and configuration files. The default flash file system on the device is named flash:.

As viewed from the active device, or any stack member, flash: refers to the local flash device, which is the device attached to the same device on which the file system is being viewed. In a device stack, each of the flash devices from the various stack members can be viewed from the active device. The names of these flash file systems include the corresponding device member numbers. For example, flash-3:, as viewed from the active device, refers to the same file system as does flash: on stack member 3. Use the **show file systems** privileged EXEC command to list all file systems, including the flash file systems in the device stack.

Only one user at a time can manage the software bundles and configuration files for a device stack .

Displaying Available File Systems

To display the available file systems on your device, use the **show file systems** privileged EXEC command as shown in this example for a standalone device:

```
Device# show file systems
Size(b) Free(b) Type Flags Prefixes
- - opaque rw system:
- - opaque rw tmpsys:
1651314688 1559785472 disk rw crashinfo:
* 11353194496 9693396992 disk rw flash:
8049967104 7959392256 disk ro webui:
- - opaque rw null:
- - opaque ro tar:
- - network rw tftp:
2097152 2080848 nvram rw nvram:
- - opaque wo syslog:
- - network rw rcp:
- - network rw http:
- - network rw ftp:
- - network rw scp:
- - network rw https:
- - opaque ro cns:

Device# show file systems
File Systems:

          Size(b)          Free(b)          Type  Flags  Prefixes
          -          -          -          -          -
          -          -          opaque   rw      system:
          -          -          opaque   rw      tmpsys:
* 11250098176 9694093312          disk    rw      bootflash: flash:
    1651314688 1232220160          disk    rw      crashinfo:
118148280320 112084115456          disk    rw      disk0:
    189628416 145387520           disk    rw      usbflash0:
    7763918848 7696850944          disk    ro      webui:
          -          -          opaque   rw      null:
          -          -          opaque   ro      tar:
          -          -          network   rw      tftp:
    33554432    33532852          nvram   rw      nvram:
          -          -          opaque   wo      syslog:
          -          -          network   rw      rcp:
          -          -          network   rw      http:
          -          -          network   rw      ftp:
          -          -          network   rw      scp:
          -          -          network   rw      https:
          -          -          opaque   ro      cns:
```

This example shows a device stack. In this example, the active device is stack member 1; the file system on stack member 2 is displayed as flash-2:, the file system on stack member 3 is displayed as flash-3: and so on up to stack member 9, displayed as flash-9: for a 9-member stack. The example also shows the crashinfo directories and a USB flash drive plugged into the active device:

```
Device# show file systems
File Systems:
```

```

Size(b) Free(b) Type Flags Prefixes
- - opaque rw system:
- - opaque rw tmpsys:
1651314688 1520742400 disk rw crashinfo: crashinfo-1:
1651507200 1516240896 disk rw crashinfo-2: stby-crashinfo:
1651507200 1517289472 disk rw crashinfo-3:
1651507200 1519386624 disk rw crashinfo-4:
1651507200 1524629504 disk rw crashinfo-5:
1651507200 1523580928 disk rw crashinfo-6:
1651507200 1517289472 disk rw crashinfo-7:
1651507200 1526726656 disk rw crashinfo-8:
* 11353194496 7916576768 disk rw flash: flash-1:
11353980928 7944011776 disk rw flash-2: stby-flash:
11353980928 7876902912 disk rw flash-3:
11353980928 7944011776 disk rw flash-4:
11353980928 7939817472 disk rw flash-5:
11353980928 7944011776 disk rw flash-6:
11353980928 7944011776 disk rw flash-7:
11353980928 7944011776 disk rw flash-8:
3824013312 3756507136 disk ro webui:
- - opaque rw null:
- - opaque ro tar:
- - network rw tftp:
2097152 2052489 nvram rw nvram:
- - opaque wo syslog:
- - network rw rcp:
- - network rw http:
- - network rw ftp:
- - network rw scp:
- - network rw https:
- - opaque ro cns:
2097152 2052489 nvram rw stby-nvram:
- - nvram rw stby-rcsf:
- - opaque rw revrcsf:

```

Table 11: show file systems Field Descriptions

Field	Value
Size(b)	Amount of memory in the file system in bytes.
Free(b)	Amount of free memory in the file system in bytes.
Type	<p>Type of file system.</p> <p>disk—The file system is for a flash memory device, USB flash, and crashinfo file.</p> <p>network—The file system for network devices; for example, an FTP server or and HTTP server.</p> <p>nvram—The file system is for a NVRAM device.</p> <p>opaque—The file system is a locally generated pseudo file system (for example, the system) or a download interface, such as brimux.</p> <p>unknown—The file system is an unknown type.</p>

Field	Value
Flags	Permission for file system. ro —read-only. rw —read/write. wo —write-only.
Prefixes	Alias for file system. crashinfo: —Crashinfo file. flash: —Flash file system. ftp: —FTP server. http: —HTTP server. https: —Secure HTTP server. nvr: —NVRAM. null: —Null destination for copies. You can copy a remote file to null to find its size. rcp: —Remote Copy Protocol (RCP) server. scp: —Session Control Protocol (SCP) server. system: —Contains the system memory, including the running configuration. tftp: —TFTP network server. usbflash0: —USB flash memory. ymodem: —Obtain the file from a network machine by using the Ymodem protocol.

Setting the Default File System

You can specify the file system or directory that the system uses as the default file system by using the **cd** *filesystem:* privileged EXEC command. You can set the default file system to omit the *filesystem:* argument from related commands. For example, for all privileged EXEC commands that have the optional *filesystem:* argument, the system uses the file system specified by the **cd** command.

By default, the default file system is *flash:*.

You can display the current default file system as specified by the **cd** command by using the **pwd** privileged EXEC command.

Displaying Information About Files on a File System

You can view a list of the contents of a file system before manipulating its contents. For example, before copying a new configuration file to flash memory, you might want to verify that the file system does not

already contain a configuration file with the same name. Similarly, before copying a flash configuration file to another location, you might want to verify its filename for use in another command. To display information about files on a file system, use one of the privileged EXEC commands listed in the following table.

Table 12: Commands for Displaying Information About Files

Command	Description
dir [/all] [filesystem:filename]	Displays a list of files on a file system.
show file systems	Displays more information about each of the files on a file system.
show file information file-url	Displays information about a specific file.
show file descriptors	Displays a list of open file descriptors. File descriptors are the internal representations of open files. You can use this command to see if another user has a file open.

For example, to display a list of all files in a file system, use the **dir** privileged EXEC command:

```
device# dir flash:
Directory of bootflash:/

616513  drwx           4096  Jul 15 2015 07:11:35 +00:00  .installer
608402  -rw-          33818  Sep 25 2015 11:41:35 +00:00  bootloader_evt_handle.log
608403  drwx           4096  Feb 27 2017 13:56:47 +00:00  .ssh
608410  -rw-           0      Jun 5 2015 10:16:17 +00:00  dc_stats.txt
608411  drwx          20480  Sep 23 2015 11:50:13 +00:00  core
624625  drwx           4096  Sep 23 2015 12:29:27 +00:00  .prst_sync
640849  drwx           4096  Feb 27 2017 13:57:30 +00:00  .rollback_timer
608412  drwx           4096  Jun 17 2015 18:12:47 +00:00  orch_test_logs
608413  -rw-        33554432  Sep 25 2015 11:43:15 +00:00  nvram_config
608417  -rw-           35     Sep 25 2015 20:17:42 +00:00  pnp-tech-time
608439  -rw-        214054  Sep 25 2015 20:17:48 +00:00  pnp-tech-discovery-summary
608419  drwx           4096  Jul 23 2015 07:50:25 +00:00  util
616514  drwx           4096  Mar 18 2015 11:09:04 +00:00  onep
608442  -rw-           556   Mar 18 2015 11:19:34 +00:00  vlan.dat
608448  -rw-       1131779  Mar 28 2015 13:13:48 +00:00  log.txt
616516  drwx           4096   Apr 1 2015 09:34:56 +00:00  gs_script
616517  drwx           4096   Apr 6 2015 09:42:38 +00:00  tools
608440  -rw-           252   Sep 25 2015 11:41:52 +00:00  boothelper.log
624626  drwx           4096  Apr 17 2015 06:10:55 +00:00  SD_AVC_AUTO_CONFIG
608488  -rw-         98869  Sep 25 2015 11:42:15 +00:00  memleak.tcl
608437  -rwx          17866  Jul 16 2015 04:01:10 +00:00  ardbeg_x86
632745  drwx           4096  Aug 20 2015 11:35:09 +00:00  CRDU
632746  drwx           4096  Sep 16 2015 08:57:44 +00:00  ardmore
608418  -rw-       1595361  Jul 8 2015 11:18:33 +00:00  system-report_RP_0_20150708-111832-UTC.tar.gz
608491  -rw-       67587176  Aug 12 2015 05:30:35 +00:00  mcln_x86_kernel_20170628.SSA
608492  -rwx       74880100  Aug 12 2015 05:30:57 +00:00  stardust.x86.idprom.0718B

11250098176 bytes total (9128050688 bytes free)
device#
```

Changing Directories and Displaying the Working Directory

Follow these steps to change directories and to display the working directory:

SUMMARY STEPS

1. **enable**
2. **dir filesystem:**
3. **cd directory_name**
4. **pwd**
5. **cd**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	dir filesystem: Example: Device# dir flash:	Displays the directories on the specified file system. For <i>filesystem:</i> , use <i>flash:</i> for the system board flash device. To access flash partitions of device members in a stack, use <i>flash-n</i> where <i>n</i> is the stack member number. For example, <i>flash-4</i> .
Step 3	cd directory_name Example: Device# cd new_configs	Navigates to the specified directory. The command example shows how to navigate to the directory named <i>new_configs</i> .
Step 4	pwd Example: Device# pwd	Displays the working directory.
Step 5	cd Example: Device# cd	Navigates to the default directory.

Creating Directories

Beginning in privileged EXEC mode, follow these steps to create a directory:

SUMMARY STEPS

1. **dir** *filesystem*:
2. **mkdir** *directory_name*
3. **dir** *filesystem*:

DETAILED STEPS

	Command or Action	Purpose
Step 1	dir <i>filesystem</i> : Example: Device# dir flash:	Displays the directories on the specified file system. For <i>filesystem</i> :, use flash: for the system board flash device.
Step 2	mkdir <i>directory_name</i> Example: Device# mkdir new_configs	Creates a new directory. Directory names are case sensitive and are limited to 45 characters between the slashes (/); the name cannot contain control characters, spaces, slashes, quotes, semicolons, or colons.
Step 3	dir <i>filesystem</i> : Example: Device# dir flash:	Verifies your entry.

Removing Directories

To remove a directory with all its files and subdirectories, use the **delete /force /recursive** *filesystem:/file-url* privileged EXEC command.

Use the **/recursive** keyword to delete the named directory and all subdirectories and the files contained in it. Use the **/force** keyword to suppress the prompting that confirms a deletion of each file in the directory. You are prompted only once at the beginning of this deletion process.

For *filesystem*, use **flash:** for the system board flash device. For *file-url*, enter the name of the directory to be deleted. All of the files in the directory and the directory are removed.



Caution When directories are deleted, their contents cannot be recovered.

Copying Files

To copy a file from a source to a destination, use the **copy** *source-url destination-url* privileged EXEC command. For the source and destination URLs, you can use **running-config** and **startup-config** keyword shortcuts. For example, the **copy running-config startup-config** command saves the currently running configuration file to the NVRAM section of flash memory to be used as the configuration during system initialization.

You can also copy from special file systems (**xmodem:**, **ymodem:**) as the source for the file from a network machine that uses the Xmodem or Ymodem protocol.

Network file system URLs include ftp:, rep:, tftp:, scp:, http:, and https: and have these syntaxes:

- FTP—ftp:[[/username [:password]@location]/directory]/filename
- RCP—rcp:[[/username@location]/directory]/filename
- TFTP—tftp:[[/location]/directory]/filename
- SCP—scp:[[/username [:password]@location]/directory]/filename
- HTTP—http:[[/username [:password]@location]/directory]/filename
- HTTPS—https:[[/username [:password]@location]/directory]/filename



Note The password must not contain the special character '@'. If the character '@' is used, the copy fails to parse the IP address of the server.

Local writable file systems include flash:.

Some invalid combinations of source and destination exist. Specifically, you cannot copy these combinations:

- From a running configuration to a running configuration
- From a startup configuration to a startup configuration
- From a device to the same device (for example, the **copy flash: flash:** command is invalid)

Copying Files from One Device in a Stack to Another Device in the Same Stack

To copy a file from one device in a stack to another device in the same stack, use the **flash-X:** notation, where **X** is the device number.

To view all devices in a stack, use the **show switch** command in privileged EXEC mode, as in the following example of a 9-member device stack:

```
Device# show switch
Switch/Stack Mac Address : 0006.f6b9.b580 - Local Mac Address Mac persistency wait time:
Indefinite
```

Switch#	Role	Mac Address	Priority	H/W Version	Current State
*1	Active	0006.f6b9.b580	15	P3B	Ready
2	Standby	0006.f6ba.0c80	14	P3B	Ready
3	Member	0006.f6ba.3300	7	P3B	Ready
4	Member	0006.f6b9.df80	6	P3B	Ready
5	Member	0006.f6ba.3880	13	P1A	Ready
6	Member	1ce6.c7b6.ef00	4	PP	Ready
7	Member	2037.06ce.2580	3	P2A	Ready
8	Member	2037.0653.7e00	2	P5A	Ready
9	Member	2037.0653.9280	1	P5B	Ready

To view all devices in a stack, use the **show switch** command in privileged EXEC mode, as in the following example of a 8-member device stack:

```
Device# show switch
Switch/Stack Mac Address : 046c.9d01.3b80 - Local Mac Address
Mac persistency wait time: 4 mins
```

Switch#	Role	Mac Address	Priority	H/W Version	Current State
*1	Active	046c.9d01.3b80	15	P4B	Ready
2	Standby	046c.9d01.0f80	13	P3C	Ready
3	Member	046c.9d01.1180	11	P4B	Ready
4	Member	046c.9d01.0e80	9	P3C	Ready
5	Member	046c.9d01.4d00	7	P3C	Ready
6	Member	046c.9d01.2800	5	P3C	Ready
7	Member	046c.9d01.6e80	3	P4B	Ready
8	Member	046c.9d01.8180	1	P4B	Ready

To view all file systems available to copy on a specific device, use the **copy** command as in the following example of a 5-member stack:

```
Device# copy flash:?
flash:.installer
flash:.prst_sync
flash:.rollback_timer
flash:boothelper.log
flash:bootloader_evt_handle.log
flash:cat9k-cc_srdriver.16.05.01a.SPA.pkg
flash:cat9k-espbase.16.05.01a.SPA.pkg
flash:cat9k-guestshell.16.05.01a.SPA.pkg
flash:cat9k-rpbase.16.05.01a.SPA.pkg
flash:cat9k-rpboot.16.05.01a.SPA.pkg
flash:cat9k-sipbase.16.05.01a.SPA.pkg
flash:cat9k-sipspa.16.05.01a.SPA.pkg
flash:cat9k-srdriver.16.05.01a.SPA.pkg
flash:cat9k-webui.16.05.01a.SPA.pkg
flash:cat9k-wlc.16.05.01a.SPA.pkg
flash:core
flash:dc_profile_dir
flash:dc_stats.txt
flash:gs_script
flash:nvram_config
flash:packages.conf
```

This example shows how to copy a config file stored in the flash partition of device 2 to the flash partition of device 4. It assumes that device 2 and device 4 are in the same stack.

```
Device# copy flash-2:config.txt flash-4:config.txt
```

Deleting Files

When you no longer need a file on a flash memory device, you can permanently delete it. To delete a file or directory from a specified flash device, use the **delete** [**/force**] [**/recursive**] [*filesystem:*]/file-url privileged EXEC command.

Use the **/recursive** keyword for deleting a directory and all subdirectories and the files contained in it. Use the **/force** keyword to suppress the prompting that confirms a deletion of each file in the directory. You are prompted only once at the beginning of this deletion process. Use the **/force** and **/recursive** keywords for

deleting old software images that were installed by using the **archive download-sw** command but are no longer needed.

If you omit the *filesystem:* option, the device uses the default device specified by the **cd** command. For *file-url*, you specify the path (directory) and the name of the file to be deleted.

When you attempt to delete any files, the system prompts you to confirm the deletion.



Caution When files are deleted, their contents cannot be recovered.

This example shows how to delete the file *myconfig* from the default flash memory device:

```
Device# delete myconfig
```

Creating, Displaying and Extracting Files

You can create a file and write files into it, list the files in a file, and extract the files from a file as described in the next sections.

Beginning in privileged EXEC mode, follow these steps to create a file, display the contents, and extract it:

SUMMARY STEPS

1. **archive tar /create** *destination-url* **flash:** */file-url*
2. **archive tar /table** *source-url*
3. **archive tar /xtract** *source-url* **flash:***/file-url* [*dir/file...*]
4. **more** [*/ascii* | */binary* | */ebcdic*] */file-url*

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>archive tar /create <i>destination-url</i> flash: <i>/file-url</i></p> <p>Example:</p> <pre>device# archive tar /create tftp:172.20.10.30/saved. flash:/new-configs</pre>	<p>Creates a file and adds files to it.</p> <p>For <i>destination-url</i>, specify the destination URL alias for the local or network file system and the name of the file to create:</p> <ul style="list-style-type: none"> • Local flash file system syntax: <ul style="list-style-type: none"> flash: • FTP syntax: <ul style="list-style-type: none"> ftp:[[<i>/username[:password]</i>@<i>location</i>]/<i>directory</i>]/-<i>filename</i>. • RCP syntax: <ul style="list-style-type: none"> rcp:[[<i>/username</i>@<i>location</i>]/<i>directory</i>]/-<i>filename</i>. • TFTP syntax: <ul style="list-style-type: none"> tftp:[[<i>/location</i>]/<i>directory</i>]/-<i>filename</i>. <p>For flash:<i>/file-url</i>, specify the location on the local flash file system in which the new file is created. You can also</p>

	Command or Action	Purpose
		specify an optional list of files or directories within the source directory to add to the new file. If none are specified, all files and directories at this level are written to the newly created file.
Step 2	<p>archive tar /table <i>source-url</i></p> <p>Example:</p> <pre>device# archive tar /table flash: /new_configs</pre>	<p>Displays the contents of a file.</p> <p>For <i>source-url</i>, specify the source URL alias for the local or network file system. The <i>-filename.</i> is the file to display. These options are supported:</p> <ul style="list-style-type: none"> Local flash file system syntax: <ul style="list-style-type: none"> flash: FTP syntax: <ul style="list-style-type: none"> ftp:[[//username[:password]@location]/directory]/-filename. RCP syntax: <ul style="list-style-type: none"> rcp:[[//username@location]/directory]/-filename. TFTP syntax: <ul style="list-style-type: none"> tftp:[[//location]/directory]/-filename. <p>You can also limit the file displays by specifying a list of files or directories after the file. Only those files appear. If none are specified, all files and directories appear.</p>
Step 3	<p>archive tar /xtract <i>source-url flash:/file-url [dir/file...]</i></p> <p>Example:</p> <pre>device# archive tar /xtract tftp:/172.20.10.30/saved. flash:/new_configs</pre>	<p>Extracts a file into a directory on the flash file system.</p> <p>For <i>source-url</i>, specify the source URL alias for the local file system. The <i>-filename.</i> is the file from which to extract files. These options are supported:</p> <ul style="list-style-type: none"> Local flash file system syntax: <ul style="list-style-type: none"> flash: FTP syntax: <ul style="list-style-type: none"> ftp:[[//username[:password]@location]/directory]/-filename. RCP syntax: <ul style="list-style-type: none"> rcp:[[//username@location]/directory]/-filename. TFTP syntax: <ul style="list-style-type: none"> tftp:[[//location]/directory]/-filename. <p>For flash:/file-url [dir/file...], specify the location on the local flash file system from which the file is extracted. Use the <i>dir/file...</i> option to specify a list of files or directories within the file to be extracted. If none are specified, all files and directories are extracted.</p>
Step 4	<p>more [/ascii /binary /ebcdic] <i>/file-url</i></p> <p>Example:</p>	Displays the contents of any readable file, including a file on a remote file system.

	Command or Action	Purpose
	device# more flash:/new-configs	

Additional References for Flash File System

Related Documents

Related Topic	Document Title
Commands for managing flash: file systems	<i>Cisco IOS Configuration Fundamentals Command Reference</i>

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	--

MIBs

MIBs	MIBs Link
No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	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

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	--

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>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.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	<p>http://www.cisco.com/cisco/web/support/index.html</p>

Feature History for Flash File System

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Everest 16.5.1a	Flash File System	The flash file system is a single flash device on which you can store files. It also provides several commands to help you manage software bundles and configuration files.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 13

Factory Reset

- [Prerequisites for Performing Factory Reset, on page 275](#)
- [Limitations for Performing Factory Reset, on page 275](#)
- [Information About Factory Reset, on page 275](#)
- [How to Perform Factory Reset, on page 276](#)
- [Feature History and Information for Factory Reset, on page 277](#)

Prerequisites for Performing Factory Reset

- Ensure that all the software images, configurations, and personal data are backed up before performing the Factory Reset operation.
- Ensure that the device is not in the stacking mode as Factory Reset is supported only in the standalone mode. For Modular-chassis in high availability mode, Factory Reset is applied per supervisor.
- Ensure that there is uninterrupted power supply when the process is in progress.
- Ensure that you take a backup of the current image before you begin the Factory Reset process.
- Ensure that neither In-Service Software Upgrade (ISSU) nor In-Service Software Downgrade (ISSD) is in progress before starting the Factory Reset process.

Limitations for Performing Factory Reset

- Software patches, if any, that are installed on the switch will not be restored after the Factory Reset operation.
- If the Factory Reset command is issued through a vty session, the session is not restored after completion of the Factory Reset process.

Information About Factory Reset

Factory Reset removes all the customer specific data that has been added to the device since the time of its shipping. Data erased includes configurations, log files, boot variables, core files, and credentials like FIPS-related keys.

The following table provides details about the data that is erased and retained during the Factory Reset process:

Table 13: Data Erased and Retained During Factory Reset

Data Erased	Data Retained
All Cisco IOS images, including the current boot image	Data from Remote field-replaceable units (FRUs)
Crash information and logs	Value of the configuration register
User data, and startup and running configuration	Contents of USB
Credentials like FIPS-related keys	Credentials like Secure Unique Device Identifier (SUDI) certificates, Public key infrastructure (PKI) keys
Onboard Failure Logging (OBFL) logs	Licenses
ROMMON variables added by the user	

The Factory Reset process can be used in the following two scenarios:

- Return Material Authorization (RMA) for a device—If you have to return a device to Cisco for RMA, remove all the customer-specific data before obtaining an RMA certificate for the device.
- Recovering the compromised device— If the key material or credentials stored on a device is compromised, reset the device to factory configuration, and then reconfigure the device.

The device reloads to perform the Factory Reset task. Note that this reload results in a ROMMON mode.

After the Factory Reset operation is complete, you can load the Cisco IOS image either through a USB or TFTP.

How to Perform Factory Reset

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	factory-reset { all config boot-vars } Example: Device# factory-reset all	Use the command with all options enabled. No system configuration is required to use the factory reset command. Erases all the content from the NVRAM, all the Cisco IOS images, including the current boot image, boot variables, startup and running configuration data, and user data.

	Command or Action	Purpose
		Use the option config to reset the start-up configurations. Use the option boot-vars to reset the user added boot variables. After the Factory Reset process is successfully completed, the device reboots and stops at ROMMON mode.

Feature History and Information for Factory Reset

Release	Feature Information
Cisco IOS XE Fuji 16.8.1a	This feature was introduced.



CHAPTER 14

Configuring Secure Storage

- [Information About Secure Storage, on page 279](#)
- [Enabling Secure Storage , on page 279](#)
- [Disabling Secure Storage , on page 280](#)
- [Verifying the Status of Encryption, on page 281](#)
- [Feature Information for Secure Storage, on page 281](#)

Information About Secure Storage

Secure Storage feature allows you to secure critical configuration information by encrypting it. It encrypts asymmetric key-pairs, pre-shared secrets, the type 6 password encryption key and certain credentials. An instance-unique encryption key is stored in the hardware trust anchor to prevent it from being compromised.

Enabling Secure Storage

Before you begin

By default, this feature is disabled.

SUMMARY STEPS

1. `configure terminal`
2. `service private-config-encryption`
3. `end`
4. `write memory`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<code>configure terminal</code> Example: Device# <code>configure terminal</code>	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	service private-config-encryption Example: Device(config)# service private-config-encryption	Enables the Secure Storage feature on your device.
Step 3	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 4	write memory Example: Device# write memory	Encrypts the private-config file and saves the file in an encrypted format.

Disabling Secure Storage

Before you begin

To disable Secure Storage feature on a device, perform this task:

SUMMARY STEPS

1. **configure terminal**
2. **no service private-config-encryption**
3. **end**
4. **write memory**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	no service private-config-encryption Example: Device(config)# no service private-config-encryption	Disables the Secure Storage feature on your device. When secure storage is disabled, all the user data is stored in plain text in the NVRAM.
Step 3	end Example: Device(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 4	write memory Example: Device# write memory	Decrypts the private-config file and saves the file in plane format.

Verifying the Status of Encryption

Use the **show parser encrypt file status** command to verify the status of encryption. The following command output indicates that the feature is available but the file is not encrypted. The file is in 'plain text' format.

```
Device#show parser encrypt file status
Feature: Enabled
File Format: Plain Text
Encryption Version: Ver1
```

Feature Information for Secure Storage

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Everest 16.5.1a	Secure Storage	Secure Storage feature allows you to secure critical configuration information by encrypting it. It encrypts asymmetric key-pairs, pre-shared secrets, the type 6 password encryption key and certain credentials. An instance-unique encryption key is stored in the hardware trust anchor to prevent it from being compromised.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 15

Conditional Debug and Radioactive Tracing

- [Finding Feature Information, on page 283](#)
- [Introduction to Conditional Debugging, on page 283](#)
- [Introduction to Radioactive Tracing, on page 284](#)
- [How to Configure Conditional Debug and Radioactive Tracing, on page 284](#)
- [Monitoring Conditional Debugging, on page 288](#)
- [Configuration Examples for Conditional Debugging, on page 288](#)
- [Additional References for Conditional Debugging and Radioactive Tracing, on page 289](#)
- [Feature History for Conditional Debugging and Radioactive Tracing, on page 290](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and 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 feature information table at the end of this module.

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.

Introduction to Conditional Debugging

The Conditional Debugging feature allows you to selectively enable debugging and logging for specific features based on the set of conditions you define. This feature is useful in systems where a large number of features are supported.



Note Only Control Plane Tracing is supported.

The Conditional debug allows granular debugging in a network that is operating at a large scale with a large number of features. It allows you to observe detailed debugs for granular instances within the system. This is very useful when we need to debug only a particular session among thousands of sessions. It is also possible to specify multiple conditions.

A condition refers to a feature or identity, where identity could be an interface, IP Address, or a MAC address and so on.



Note MAC address is the only supported condition.

This is in contrast to the general debug command, that produces its output without discriminating on the feature objects that are being processed. General debug command consumes a lot of system resources and impacts the system performance.

Introduction to Radioactive Tracing

Radioactive tracing provides the ability to stitch together a chain of execution for operations of interest across the system, at an increased verbosity level. This provides a way to conditionally print debug information (up to DEBUG Level or a specified level) across threads, processes and function calls.



Note The default level is **DEBUG**. The users cannot change this to another level.

The following features are enabled for Radioactive Tracing:

- IGMP Snooping
- Layer 2 Multicast

How to Configure Conditional Debug and Radioactive Tracing

Conditional Debugging and Radioactive Tracing

Radioactive Tracing when coupled with Conditional Debugging, enable us to have a single debug CLI to debug all execution contexts related to the condition. This can be done without being aware of the various control flow processes of the feature within the box and without having to issue debugs at these processes individually.

Location of Tracefiles

By default the tracefile logs will be generated for each process and saved into either the **/tmp/rp/trace** or **/tmp/fp/trace** directory. In this temp directory, the trace logs are written to files, which are of 1 MB size each. The directory can hold up to a maximum of 25 such files for a given process. When a tracefile in the **/tmp** directory reaches its 1MB limit or whatever size was configured for it during the boot time, it is rotated out to an archive location in the **/crashinfo** partition under **tracelogs** directory.

The **/tmp** directory holds only a single tracefile for a given process. Once the file reaches its file size limit it is rotated out to **/crashinfo/tracelogs**. In the archive directory, up to 25 files are accumulated, after which the oldest one is replaced by the newly rotated file from **/tmp**.

The tracefiles in the crashinfo directory are located in the following formats:

1. Process-name_Process-ID_running-counter.timestamp.gz
Example: IOSRP_R0-0.bin_0.14239.20151101234827.gz
2. Process-name_pmanlog_Process-ID_running-counter.timestamp.bin.gz
Example: wcm_pmanlog_R0-0.30360_0.20151028233007.bin.gz

Configuring Conditional Debugging

To configure conditional debugging, follow the steps given below:

SUMMARY STEPS

1. **enable**
2. **debug platform condition mac** {mac-address}
3. **debug platform condition start**
4. **show platform condition** OR **show debug**
5. **debug platform condition stop**
6. **request platform software trace archive** [last {number} days] [target {crashinfo: |flashinfo:}]
7. **show platform software trace** [filter-binary | level | message]
8. **clear platform condition all**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	debug platform condition mac {mac-address} Example: Device# debug platform condition mac bc16.6509.3314	Configures conditional debugging for the MAC Address specified.
Step 3	debug platform condition start Example: Device# debug platform condition start	Starts conditional debugging (this will start radioactive tracing if there is a match on one of the conditions above).
Step 4	show platform condition OR show debug Example: Device# show platform condition Device# show debug	Displays the current conditions set.
Step 5	debug platform condition stop Example:	Stops conditional debugging (this will stop radioactive tracing).

	Command or Action	Purpose
	Device# <code>debug platform condition stop</code>	
Step 6	<p>request platform software trace archive [last {number} days] [target {crashinfo: flashinfo:}]</p> <p>Example:</p> <pre>Device# request platform software trace archive last 2 days</pre>	(Optional) Displays historical logs of merged tracefiles on the system. Filter on any combination of number of days or location.
Step 7	<p>show platform software trace [filter-binary level message]</p> <p>Example:</p> <pre>Device# show platform software trace message</pre>	<p>(Optional) Displays logs merged from the latest tracefile. Filter on any combination of application condition, trace module name, and trace level.</p> <ul style="list-style-type: none"> • filter-binary - Filter the modules to be collated • level - Show trace levels • message - Show trace message ring contents <p>Note On the device:</p> <ul style="list-style-type: none"> • Available from IOS console in addition to linux shell. • Generates a file with merged logs. • Displays merged logs only from staging area
Step 8	<p>clear platform condition all</p> <p>Example:</p> <pre>Device# clear platform condition all</pre>	Clears all conditions.

What to do next



Note The commands **request platform software trace filter-binary** and **show platform software trace filter-binary** work in a similar way. The only difference is:

- **request platform software trace filter-binary** - Sources the data from historical logs.
- **show platform software trace filter-binary** – Sources the data from the flash Temp directory.

Of these, `mac_log <..date..>` is the most important file, as it gives the messages for the MAC we are debugging. The command **show platform software trace filter-binary** also generates the same flash files, and also prints the `mac_log` on the screen.

Radioactive Tracing for L2 Multicast

To identify a specific multicast receiver, specify the MAC address of the joiner or the receiver client, Group Multicast IP address and Snooping VLAN. Additionally, enable the trace level for the debug. The debug level will provide detailed traces and better visibility into the system.

debug platform condition feature multicast controlplane mac *client MAC address ip Group IP address vlan id level debug level*

Recommended Workflow for Trace files

The Recommended Workflow for Trace files is listed below:

1. To request the tracelogs for a specific time period.
EXAMPLE 1 day.
Use the command:
Device#request platform software trace archive last 1 day
2. The system generates a tar ball (.gz file) of the tracelogs in the location /flash:
3. Copy the file off the switch. By copying the file, the tracelogs can be used to work offline. For more details on copying files, see section below.
4. Delete the tracelog file (.gz) file from /flash: location. This will ensure enough space on the switch for other operations.

Copying tracefiles off the box

An example of the tracefile is shown below:

```
Device# dir crashinfo:/tracelogs
Directory of crashinfo:/tracelogs/

50664 -rwx 760 Sep 22 2015 11:12:21 +00:00 plogd_F0-0.bin_0.gz
50603 -rwx 991 Sep 22 2015 11:12:08 +00:00 fed_pmanlog_F0-0.bin_0.9558.20150922111208.gz
50610 -rw- 11 Nov 2 2015 00:15:59 +00:00 timestamp
50611 -rwx 1443 Sep 22 2015 11:11:31 +00:00
auto_upgrade_client_sh_pmanlog_R0-.bin_0.3817.20150922111130.gz
50669 -rwx 589 Sep 30 2015 03:59:04 +00:00 cfgwr-8021_R0-0.bin_0.gz
50612 -rwx 1136 Sep 22 2015 11:11:46 +00:00 reflector_803_R0-0.bin_0.1312.20150922111116.gz
50794 -rwx 4239 Nov 2 2015 00:04:32 +00:00 IOSRP_R0-0.bin_0.14239.20151101234827.gz
50615 -rwx 131072 Nov 2 2015 00:19:59 +00:00 linux_iosd_image_pmanlog_R0-0.bin_0
--More--
```

The trace files can be copied using one of the various options shown below:

```
Device# copy crashinfo:/tracelogs ?
crashinfo: Copy to crashinfo: file system
flash: Copy to flash: file system
ftp: Copy to ftp: file system
http: Copy to http: file system
https: Copy to https: file system
null: Copy to null: file system
```

```

nvram: Copy to nvram: file system
rcp: Copy to rcp: file system
running-config Update (merge with) current system configuration
scp: Copy to scp: file system
startup-config Copy to startup configuration
syslog: Copy to syslog: file system
system: Copy to system: file system
tftp: Copy to tftp: file system
tmpsys: Copy to tmpsys: file system

```

The general syntax for copying onto a TFTP server is as follows:

```

Device# copy source: tftp:
Device# copy crashinfo:/tracelogs/IOSRP_R0-0.bin_0.14239.20151101234827.gz tftp:
Address or name of remote host []? 2.2.2.2
Destination filename [IOSRP_R0-0.bin_0.14239.20151101234827.gz]?

```



Note It is important to clear the generated report or archive files off the switch in order to have flash space available for tracelog and other purposes.

Monitoring Conditional Debugging

The table shown below lists the various commands that can be used to monitor conditional debugging.

Command	Purpose
show platform condition	Displays the current conditions set.
show debug	Displays the current debug conditions set.
show platform software trace filter-binary	Displays logs merged from the latest tracefile.
request platform software trace filter-binary	Displays historical logs of merged tracefiles on the system.

Configuration Examples for Conditional Debugging

The following is an output example of the *show platform condition* command.

```

Device# show platform condition
Conditional Debug Global State: Stop
Conditions Direction
-----|-----
MAC Address 0024.D7C7.0054 N/A
Feature Condition Type Value
-----|-----
Device#

```

The following is an output example of the *show debug* command.

```

Device# show debug
IOSXE Conditional Debug Configs:
Conditional Debug Global State: Start

```

```

Conditions Direction
-----|-----
MAC Address 0024.D7C7.0054 N/A
Feature Condition Type Value
-----|-----
Packet Infra debugs:
Ip Address Port
-----|-----
Device#

```

The following is a sample of the `debug platform condition stop` command.

```

Device# debug platform condition stop
Conditional Debug Global State: Stop

```

Additional References for Conditional Debugging and Radioactive Tracing

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	<i>Command Reference (Catalyst 9300 Series Switches)</i>

MIBs

MIB	MIBs Link
All the 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
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>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.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History for Conditional Debugging and Radioactive Tracing

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Everest 16.5.1a	Conditional Debugging and Radioactive Tracing	The Conditional Debugging feature allows you to selectively enable debugging and logging for specific features based on the set of conditions you define.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 16

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.

- [Information About Troubleshooting the Software Configuration, on page 291](#)
- [How to Troubleshoot the Software Configuration, on page 298](#)
- [Verifying Troubleshooting of the Software Configuration, on page 312](#)
- [Scenarios for Troubleshooting the Software Configuration, on page 314](#)
- [Configuration Examples for Troubleshooting Software, on page 316](#)
- [Feature History for Troubleshooting Software Configuration, on page 318](#)

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. Follow the steps described in the [Recovering from a Software Failure, on page 298](#) section to recover from a software failure.

Lost or Forgotten Password on a Device

The default configuration for the device allows an end user with physical access to the device 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 device.



Note On these devices, 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.



Note You cannot recover encryption password key, when Cisco WLC configuration is copied from one Cisco WLC to another (in case of an RMA).

Follow the steps described in the section [Recovering from a Lost or Forgotten Password, on page 302](#) to recover from a lost or forgotten password.

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 9300 Switches)*.

Refer to the section [Scenarios to Troubleshoot Power over Ethernet \(PoE\), on page 314](#) for various PoE troubleshooting scenarios.

Disabled Port Caused by Power Loss

If a powered device (such as a Cisco IP Phone 7910) that is connected to a PoE Device 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 Device to recover from the error-disabled state.

On a Device, 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 Device 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.

Refer to the section [Executing Ping, on page 309](#) to understand how **ping** works.

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 Device in the path. When the Device detects a device in the path that does not support Layer 2 traceroute, the Device continues to send Layer 2 trace queries and lets them time out.

The Device 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 device is reachable from another device when you can test connectivity by using the **ping** privileged EXEC command. All devices 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 device that is not in the physical path from the source device to the destination device. All devices 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 device 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 device uses the associated MAC address and identifies the physical path.
 - If an ARP entry does not exist, the device 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.
- Layer 2 traceroute opens a listening socket on the User Datagram Protocol (UDP) port 2228 that can be accessed remotely with any IPv4 address, and does not require any authentication. This UDP socket allows to read VLAN information, links, presence of particular MAC addresses, and CDP neighbor information, from the device. This information can be used to eventually build a complete picture of the Layer 2 network topology.
- Layer 2 traceroute is enabled by default and can be disabled by running the **no l2 traceroute** command in global configuration mode. To re-enable Layer 2 traceroute, use the **l2 traceroute** command in global configuration mode.

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 Device 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 Device is the destination of the traceroute, it is displayed as the final destination in the traceroute output. Intermediate Device 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 Device is a multilayer Device that is routing a particular packet, this Device 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.

Go to [Example: Performing a Traceroute to an IP Host, on page 317](#) to see an example of IP traceroute process.

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.

System Report

System reports or crashinfo files save information that helps Cisco technical support representatives to debug problems that caused the Cisco IOS image to fail (crash). It is necessary to quickly and reliably collect critical crash information with high fidelity and integrity. Further, it is necessary to collect this information and bundle it in a way that it can be associated or identified with a specific crash occurrence.

System reports are generated in these situations:

- In case of a switch failure—A system report is generated on the member that failed; reports are not generated on other members in the stack.
- In case of a switchover—System reports are generated only on high availability (HA) member switches. Reports are not generated for non-HA members.

The system does not generate reports in case of a reload.

During a process crash, the following is collected locally from the switch:

1. Full process core
2. Tracelogs
3. IOS syslogs (not guaranteed in case of non-active crashes)
4. System process information
5. Bootup logs
6. Reload logs
7. Certain types of /proc information

This information is stored in separate files which are then archived and compressed into one bundle. This makes it convenient to get a crash snapshot in one place, and can be then moved off the box for analysis. This report is generated before the switch goes down to rommon/bootloader.

Except for the full core and tracelogs, everything else is a text file.

Use the **request platform software process core fed switch active** command to generate the core dump.

```
Device# request platform software process core fed switch active
SUCCESS: Core file generated.
```

```
Device# dir bootflash:/core
Directory of bootflash:/core/
16430  -rw-          10941657  Apr  6 2022 00:15:20 +00:00
Switch_1_RP_0_fed_18469_20220406-001511-UTC.core.gz
16812  -rw-           1          Apr  6 2022 00:01:48 +00:00  .callhome
16810  drwx           4096   Jan 18 2022 21:10:35 +00:00  modules
```

Crashinfo Files

By default the system report file will be generated and saved into the /crashinfo directory. If it cannot be saved to the crashinfo partition for lack of space, then it will be saved to the /flash directory.

To display the files, enter the **dir crashinfo:** command. The following is sample output of a crashinfo directory:

```
Device# dir crashinfo:
Directory of crashinfo:/

23665 drwx 86016 Jun  9 2017 07:47:51 -07:00 tracelogs
11  -rw-  0 May 26 2017 15:32:44 -07:00 koops.dat
12  -rw- 4782675 May 29 2017 15:47:16 -07:00 system-report_1_20170529-154715-PDT.tar.gz
1651507200 bytes total (1519386624 bytes free)
```

System reports are located in the crashinfo directory in the following format:

```
system-report_[switch number]_[date]-[timestamp]-UTC.gz
```

After a switch crashes, check for a system report file. 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 while troubleshooting the issue.

The system report generated can be further copied using TFTP, HTTP and few other options.

```
Device# copy crashinfo: ?
crashinfo: Copy to crashinfo: file system
flash: Copy to flash: file system
ftp: Copy to ftp: file system
http: Copy to http: file system
https: Copy to https: file system
null: Copy to null: file system
nvram: Copy to nvram: file system
rcp: Copy to rcp: file system
running-config Update (merge with) current system configuration
scp: Copy to scp: file system
startup-config Copy to startup configuration
syslog: Copy to syslog: file system
system: Copy to system: file system
tftp: Copy to tftp: file system
tmpsys: Copy to tmpsys: file system
```

The general syntax for copying onto TFTP server is as follows:

```
Device# copy crashinfo: tftp:
Source filename [system-report_1_20150909-092728-UTC.gz]?
```

```
Address or name of remote host []? 1.1.1.1
Destination filename [system-report_1_20150909-092728-UTC.gz]?
```

The tracelogs from all members in the stack can be collected by issuing a trace archive command. This command provides time period options. The command syntax is as follows:

```
Device# request platform software trace archive ?
last      Archive trace files of last x days
target    Location and name for the archive file
```

The tracelogs stored in crashinfo: or flash: directory from within the last 3650 days can be collected.

```
Device# request platform software trace archive last ?
<1-3650> Number of days (1-3650)
Switch#request platform software trace archive last 3650 days target ?
crashinfo: Archive file name and location
flash:     Archive file name and location
```



Note It is important to clear the system reports or trace archives from flash or crashinfo directory once they are copied out, in order to have space available for tracelogs and other purposes.

Onboard Failure Logging on the Switch

You can use the onboard failure logging (OBFL) feature to collect information about the Device. The information includes uptime, temperature, and voltage information and helps Cisco technical support representatives to troubleshoot Device 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 Device and small form-factor pluggable (SFP) modules. The Device stores this information in the flash memory:

- CLI commands—Record of the OBFL CLI commands that are entered on a standalone Device or a switch stack member.
- Environment data—Unique device identifier (UDI) information for a standalone Device or a switch 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 Device or a switch stack member.
- Power over Ethernet (PoE)—Record of the power consumption of PoE ports on a standalone Device or a switch stack member.
- Temperature—Temperature of a standalone Device or a switch stack member.
- Uptime data—Time when a standalone Device or a switch stack member starts, the reason the Device restarts, and the length of time the Device has been running since it last restarted.
- Voltage—System voltages of a standalone Device or a switch stack member.

You should manually set the system clock or configure it by using Network Time Protocol (NTP).

When the Device is running, you can retrieve the OBFL data by using the **show logging onboard** privileged EXEC commands. If the Device fails, contact your Cisco technical support representative to find out how to retrieve the data.

When an OBFL-enabled Device is restarted, there is a 10-minute delay before logging of new data begins.

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 device does not shut down, and this error message appears:

```
Multiple fan(FRU/PS) failure detected. System may get overheated. Change fan quickly.
```

The Device 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 Device fails, the Device automatically shuts down, and this error message appears:

```
Faulty (FRU/PS) fans detected, shutting down system!
```

After the first fan shuts down, if the Device detects a second fan failure, the Device waits for 20 seconds before it shuts down.

To restart the Device, 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, some of which are the following:

- 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.

Set the baud rate of the terminal to match the the default rate of 9600 bits per second [bps] of the switch console port. If the baud rate is set to a value other than 9600 bps, access to the console will be lost until the speed is set back to the default.

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 prompt, ensure that you can ping your TFTP server.

a) Set switch IP address: **set IP_ADDRESS** *ip_address*

Example:

```
switch: set IP_ADDRESS 192.0.2.123
```

b) Set switch subnet mask: **set IP_SUBNET_MASK** *subnet_mask*

Example:

```
switch: set IP_SUBNET_MASK 255.255.255.0
```

c) Set default gateway: **set DEFAULT_GATEWAY** *ip_address*

Example:

```
switch: set DEFAULT_ROUTER 192.0.2.1
```

d) 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 Choose one of the following:

- From the bootloader prompt, initiate the **boot tftp** command that assists you in recovering the software image on your switch.

```
switch: boot tftp://10.168.0.1/cat9k/cat9k_iosxe.2017-08-25_09.41.bin
attempting to boot from [tftp://10.168.0.1/cat9k/cat9k_iosxe.2017-08-25_09.41.SSA.bin]

interface : eth0
macaddr   : E4:AA:5D:59:7B:44
ip        : 10.168.247.10
netmask   : 10.255.0.0
gateway   : 10.168.0.1
server    : 10.168.0.1
file      : cat9k/cat9k_iosxe.2017-08-25_09.41.bin
```

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 [Everest], Catalyst L3 Switch Software (CAT9K_IOSXE), Version 16.6.1 RELEASE SOFTWARE (fc2)
Copyright (c) 1986-2017 by Cisco Systems, Inc.
Compiled Thu 24-Aug-17 13:23 by mcpre

Cisco IOS-XE software, Copyright (c) 2005-2017 by cisco Systems, Inc. All rights reserved. Certain components of Cisco IOS-XE software are licensed under the GNU General Public License ("GPL") Version 2.0. The software code licensed under GPL Version 2.0 is free software that comes with ABSOLUTELY NO WARRANTY. You can redistribute and/or modify such GPL code under the terms of GPL Version 2.0. For more details, see the documentation or "License Notice" file accompanying the IOS-XE software, or the applicable URL provided on the flyer accompanying the IOS-XE software.

FIPS: Flash Key Check : Begin
FIPS: Flash Key Check : End, Not Found, FIPS Mode Not Enabled

This product contains cryptographic features and is subject to United States and local country laws governing import, export, transfer and use. Delivery of Cisco cryptographic products does not imply third-party authority to import, export, distribute or use encryption. Importers, exporters, distributors and users are responsible for compliance with U.S. and local country laws. By using this product you agree to comply with applicable laws and regulations. If you are unable to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
<http://www.cisco.com/wwl/export/crypto/tool/stqrg.html>

If you require further assistance please contact us by sending email to export@cisco.com.

cisco C9XXX (X86) processor (revision V00) with 869398K/6147K bytes of memory.
Processor board ID FXS1939Q3LZ
144 Gigabit Ethernet interfaces
16 Ten Gigabit Ethernet interfaces
4 Forty Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
15958516K bytes of physical memory.
11161600K bytes of Bootflash at bootflash:.
1638400K bytes of Crash Files at crashinfo:.
0K bytes of WebUI ODM Files at webui:.

```
%INIT: waited 0 seconds for NVRAM to be available
```

```
Press RETURN to get started!
```

- Install the software from the recovery partition. This recovery image is required for recovery using the emergency-install feature.

- Verify that you have a recovery image in your recovery partition (sda9):

Example:

```
switch: dir sda9:

Size              Attributes      Name
-----
21680202         -rw-           cat9k-recovery.SSA.bin
-----
```

- From the bootloader 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://10.255.254.254/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin
WARNING: The system partition (bootflash:) will be erased during the system recovery install
process.
Are you sure you want to proceed? [y] y/n [n]: y
Starting system recovery (tftp://10.255.254.254/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin)
...
Attempting to boot from [sda9:cat9k-recovery.SSA.bin]
Located cat9k-recovery.SSA.bin
#####

Warning: ignoring ROMMON var "BOOT_PARAM"

PLATFORM_TYPE C9X00 speed 9600

Booting Recovery Image 16.5.1a

Initiating Emergency Installation of bundle
tftp://10.255.254.254/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin

Downloading bundle tftp://10.255.254.254/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin...
curl_vrf=2
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
 100 485M  100 485M    0     0  5143k      0  0:01:36  0:01:36  ---:--:-- 5256k
 100 485M  100 485M    0     0  5143k      0  0:01:36  0:01:36  ---:--:-- 5143k

Validating bundle tftp://10.255.254.254/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin...
Installing bundle tftp://10.255.254.254/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin...
Verifying bundle tftp://10.255.254.254/auto/tftpboot/X86/cat9k_iosxe.16.05.01a.SPA.bin...
Package cat9k-cc_srdriver.16.05.01a.SPA.pkg /temp//stage/cat9k-cc_srdriver.16.05.01a.SPA.pkg is
Digitally Signed
Package cat9k-espbase.16.05.01a.SPA.pkg /temp//stage/cat9k-espbase.16.05.01a.SPA.pkg is Digitally
Signed
Package cat9k-guestshell.16.05.01a.SPA.pkg /temp//stage/cat9k-guestshell.16.05.01a.SPA.pkg is
```

```

Digitally Signed
Package cat9k-rpbase.16.05.01a.SPA.pkg /temp//stage/cat9k-rpbase.16.05.01a.SPA.pkg is Digitally
Signed
Package cat9k-sipbase.16.05.01a.SPA.pkg /temp//stage/cat9k-sipbase.16.05.01a.SPA.pkg is Digitally
Signed
Package cat9k-sipspa.16.05.01a.SPA.pkg /temp//stage/cat9k-sipspa.16.05.01a.SPA.pkg is Digitally
Signed
Package cat9k-srdriver.16.05.01a.SPA.pkg /temp//stage/cat9k-srdriver.16.05.01a.SPA.pkg is Digitally
Signed
Package cat9k-webui.16.05.01a.SPA.pkg /temp//stage/cat9k-webui.16.05.01a.SPA.pkg is Digitally
Signed
Package cat9k-wlc.16.05.01a.SPA.pkg /temp//stage/cat9k-wlc.16.05.01a.SPA.pkg is Digitally Signed
Package /cat9k-rpboot.16.05.01a.SPA.pkg /temp//rpboot/cat9k-rpboot.16.05.01a.SPA.pkg is Digitally
Signed
Preparing flash...
Flash filesystem unmounted successfully /dev/sdb3
Syncing device...
Emergency Install successful... Rebooting
Will reboot now

Initializing Hardware...

System Bootstrap, Version 16.5.2r, RELEASE SOFTWARE (P)
Compiled Wed 05/31/2017 15:58:35.22 by rel

Current image running:
Primary Rommon Image

Last reset cause: SoftwareReload
C9X00 platform with 8388608 Kbytes of main memory

```

Alternatively, you can copy the image from TFTP to local flash through Telnet or Management port and then boot the device from local flash.

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 switch or the active switch. As soon as the System LED blinks, press and release the Mode button 2-3 times. The switch enters the ROMMON mode.
5. After recovering the password, reload the switch or the active switch.

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.
 - Connect a PC to the Ethernet management port.

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 switch or the active switch. As soon as the System LED blinks, press and release the Mode button 2-3 times. The switch enters the ROMMON mode.

The following console messages are displayed during the reload:

```

Initializing Hardware...

System Bootstrap, Version 16.6.1r [FC1], RELEASE SOFTWARE (P)
Compiled Sat 07/15/2017 8:31:57.39 by rel

Current image running:
Primary Rommon Image

Last reset cause: SoftwareReload <---- Start pressing and releasing the mode button
C9300-24U platform with 8388608 Kbytes of main memory

attempting to boot from [flash:packages.conf]

Located file packages.conf
#
#####

Unable to load cat9k-rpboot.16.06.02b.SPA.pkg
Failed to boot file flash:user/packages.conf
ERROR: failed to boot from flash:packages.conf (Aborted) <--- will abort
switch:
switch: <---- ROMMON
    
```

Proceed to the *Procedure with Password Recovery Enabled* section, and follow the steps.

Step 5 After recovering the password, reload the switch or the active switch.

On a switch:

```

Switch> reload
Proceed with reload? [confirm] y
    
```

Procedure with Password Recovery Enabled

Step 1 Enable manual boot mode.

```
Device: MANUAL_BOOT=yes
```

Step 2 Ignore the startup configuration with the following command:

```
Device: SWITCH_IGNORE_STARTUP_CFG=1
```

Step 3 Boot the switch with the *packages.conf* file from flash.

```
Device: 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.

```
Device> enable  
Device#
```

Step 6 Copy the startup configuration to running configuration.

```
Device# 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.

```
Device# configure terminal  
Device(config)# enable secret password
```

Step 8 Set the SWITCH_IGNORE_STARTUP_CFG parameter to 0.

```
Device(config)# no system ignore startupconfig switch all  
Device(config)# end
```

Step 9 Write the running configuration to the startup configuration file and save the configuration.

```
Device# copy running-config startup-config
```

```
Device# write memory
```

Step 10 Confirm that manual boot mode is enabled.

```
Device# show boot

BOOT variable = flash:packages.conf;
Manual Boot = yes
Enable Break = yes
```

Step 11 Reload the device.

```
Device# reload
```

Step 12 Boot the device with the *packages.conf* file from flash.

```
Device: boot flash:packages.conf
```

Step 13 After the device boots up, disable manual boot on the device.

```
Device(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 device 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 device 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:

```
Device: dir flash:
```

The device 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
cat9k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin
162196684
```

Step 3 Boot up the system:

```
Device: 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 device prompt, enter privileged EXEC mode:

```
Device> enable
```

Step 5 Enter global configuration mode:

```
Device# configure terminal
```

Step 6 Change the password:

```
Device(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:

```
Device(config)# exit
Device#
```

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:


```
Device# copy running-config startup-config
```

The new password is now in the startup configuration.

- Step 9** You must now reconfigure the device. If the system administrator has the backup device 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 Device 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 Device should be green. Depending on the Device 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 active switch. 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 Device in the stack can make it easier to remotely troubleshoot the switch stack. However, you need to remember that the Device have manually assigned numbers if you add, remove, or rearrange Device 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 Device functions with the exact same configuration as the replaced Device. This is also assuming the new Device is using the same member number as the replaced Device.

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 Device.

For the commands that you can use to monitor the switch stack and its members, see the *Displaying Switch Stack Information* section.

Preventing Autonegotiation Mismatches

The IEEE 802.3ab autonegotiation protocol manages the device 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 the device 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 Device, the Device 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 Device 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 Device, 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 Device brings the interface out of the error-disabled state and retries the operation. For more information about the **errdisable recovery** command, see the command reference for this release.

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. For more information, see the **show interfaces transceiver** command in the command reference for this release.

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 Device.



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 Device:

Command	Purpose
<p>ping ip <i>host address</i></p> <p>Device# ping 172.20.52.3</p>	Pings a remote host through IP or by supplying the hostname or network address.

Monitoring Temperature

The Device 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 Device (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. For more information, see the command reference for this release.

Monitoring the Physical Path

You can monitor the physical path that a packet takes from a source device to a destination device by using one of these privileged EXEC commands:

Table 14: Monitoring the Physical Path

Command	Purpose
<p>tracetroute mac [interface <i>interface-id</i>] {<i>source-mac-address</i>} [interface <i>interface-id</i>] {<i>destination-mac-address</i>} [vlan <i>vlan-id</i>] [detail]</p>	Displays the Layer 2 path taken by the packets from the specified source MAC address to the specified destination MAC address.

Command	Purpose
tracertoute mac ip { <i>source-ip-address</i> <i>source-hostname</i> } { <i>destination-ip-address</i> <i>destination-hostname</i> } [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 **tracertoute** privileged EXEC command, they are not supported in this release.

Command	Purpose
tracertoute ip <i>host</i> Device# <code>tracertoute ip 192.51.100.1</code>	Traces the path that packets take through the network.

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.

For more information about system message logging, see *Configuring System Message Logging*.

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 Device application-specific integrated circuits (ASICs). However, packet forwarding information can also be helpful in troubleshooting.

Using the show debug command

The **show debug** command is entered in privileged EXEC mode. This command displays all debug options available on the switch.

To view all conditional debug options run the command **show debug condition**. The commands can be listed by selecting either a condition identifier <1-1000> or *all* conditions.

To disable debugging, use the **no debug all** command.

**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. Moreover, 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.

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 logging 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 logging onboard switch switch-number** privileged EXEC command.

To clear all the OBFL data in the flash memory except for the uptime, use the **clear logging onboard RP active {application}** privileged EXEC command.

Verifying Troubleshooting of the Software Configuration

Displaying OBFL Information

Table 15: Commands for Displaying OBFL Information

Command	Purpose
show onboard switch <i>switch-number</i> cliilog Device# show onboard switch 1 cliilog	Displays the OBFL CLI commands that were entered on a standalone switch or the specified stack members.
show onboard switch <i>switch-number</i> environment Device# 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 <i>switch-number</i> message Device# show onboard switch 1 message	Displays the hardware-related messages generated by a standalone switch or the specified stack members.
show onboard switch <i>switch-number</i> counter Device# show onboard switch 1 counter	Displays the counter information on a standalone switch or the specified stack members.
show onboard switch <i>switch-number</i> temperature Device# show onboard switch 1 temperature	Displays the temperature of a standalone switch or the specified switch stack members.
show onboard switch <i>switch-number</i> uptime Device# 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 <i>switch-number</i> voltage Device# show onboard switch 1 voltage	Displays the system voltages of a standalone switch or the specified stack members.
show onboard switch <i>switch-number</i> status Device# show onboard switch 1 status	Displays the status of a standalone switch or the specified stack members.

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.

```
Device# 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 16: 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 17: Power over Ethernet Troubleshooting Scenarios

Symptom or Problem	Possible Cause and Solution
<p>Only one port does not have PoE.</p> <p>Trouble is on only one switch port. PoE and non-PoE devices do not work on this port, but do on other ports.</p>	<p>Verify that the powered device works on another PoE port.</p> <p>Use the show run, or show interface status user EXEC commands to verify that the port is not shut down or error-disabled.</p> <p>Note Most switches turn off port power when the port is shut down, even though the IEEE specifications make this optional.</p> <p>Verify that power inline never is not configured on that interface or port.</p> <p>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.</p> <p>Note Cisco powered device works only with straight cable and not with crossover one.</p> <p>Verify that the total cable length from the switch front panel to the powered device is not more than 100 meters.</p> <p>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.</p> <p>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 power inline command to verify the amount of available power.</p>

Symptom or Problem	Possible Cause and Solution
<p>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.</p>	<p>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.</p> <p>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.</p> <p>Use the show log privileged EXEC command to review alarms or system messages that previously reported PoE conditions or status changes.</p> <p>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 reenale the ports.</p> <p>Use the show env power and show power inline privileged EXEC commands to review the PoE status and power budget (available PoE).</p> <p>Review the running configuration to verify that power inline never is not configured on the ports.</p> <p>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.</p> <p>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.</p> <p>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.</p> <p>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.</p> <p>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.</p>

Symptom or Problem	Possible Cause and Solution
<p>Cisco pre-standard powered device disconnects or resets.</p> <p>After working normally, a Cisco phone intermittently reloads or disconnects from PoE.</p>	<p>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.</p> <p>Verify that the cable length is not more than 100 meters from the switch port to the powered device.</p> <p>Notice what changes in the electrical environment at the switch location or what happens at the powered device when the disconnect occurs.</p> <p>Notice whether any error messages appear at the same time a disconnect occurs. Use the show log privileged EXEC command to review error messages.</p> <p>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.)</p> <p>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.</p>
<p>IEEE 802.3af-compliant or IEEE 802.3at-compliant powered devices do not work on Cisco PoE switch.</p> <p>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.</p>	<p>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.</p> <p>Use the show interface status command to verify that the switch detects the connected powered device.</p> <p>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 <i>inrush</i>) current that exceeds a current-limit threshold for the port.</p>

Configuration Examples for Troubleshooting Software

Example: Pinging an IP Host

This example shows how to ping an IP host:

```
Device# 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
Device#
```

Table 18: 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.

Example: Performing a Traceroute to an IP Host

This example shows how to perform a **traceroute** to an IP host:

```
Device# 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 19: 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.

Character	Description
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.

Feature History for Troubleshooting Software Configuration

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Everest 16.5.1a	Troubleshooting Software Configuration	Troubleshooting software configuration describes how to identify and resolve software problems related to the Cisco IOS software on the switch.

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.