Cisco Nexus 3000 Series NX-OS System Management Configuration Guide, Release 5.0(3)U4(1)

First Published: August 26, 2012

Last Modified: August 26, 2012

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

Text Part Number: OL-26558-02

CONTENTS

Preface

This preface contains the following sections:

Audience

This publication is for experienced network administrators who configure and maintain Cisco Nexus Series devices.

Document Conventions

Command descriptions use the following conventions:

Convention	Description
bold	Bold text indicates the commands and keywords that you enter literally as shown.
Italic	Italic text indicates arguments for which the user supplies the values.
[x]	Square brackets enclose an optional element(keyword or argument).
[x | y]	Square brackets enclosing keywords or arguments separated by a vertical bar indicate an optional choice.
{x | y}	Braces enclosing keywords or arguments separated by a vertical bar indicate a required choice.

Convention	Description
[x {y | z}]	Nested set of square brackets or braces indicate optional or required choices within optional or required elements. Braces and a vertical bar within square brackets indicate a required choice within an optional element.
`variable`	Indicates a variable for which you supply values, in context where italics cannot be used.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.

Examples use the following conventions:

Convention	Description
`screen font`	Terminal sessions and information the switch displays are in screen font.
`boldface screen font`	Information you must enter is in boldface screen font.
italic screen font	Arguments for which you supply values are in italic screen font.
< >	Nonprinting characters, such as passwords, are in angle brackets.
[ ]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

This document uses the following conventions:

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

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

Related Documentation for Nexus 3000 Series NX-OS Software

The entire Cisco NX-OS 3000 Series documentation set is available at the following URL:

http://www.cisco.com/en/US/products/ps11541/tsd_products_support_series_home.html

Release Notes

The release notes are available at the following URL:

http://www.cisco.com/en/US/products/ps11541/prod_release_notes_list.html

Installation and Upgrade Guides

The installation and upgrade guides are available at the following URL:

http://www.cisco.com/en/US/products/ps11541/prod_installation_guides_list.html

The documents in this category include:

- *Cisco Nexus 5000 Series, Cisco Nexus 3000 Series, and Cisco Nexus 2000 Series Safety Information and Documentation*
- *Regulatory, Compliance, and Safety Information for the Cisco Nexus 5000 Series, Cisco Nexus 3000 Series, and Cisco Nexus 2000 Series*
- *Cisco Nexus 3000 Series Hardware Installation Guide*

License Information

For information about feature licenses in NX-OS, see the *Cisco NX-OS Licensing Guide*, available at the following URL: http://www.cisco.com/en/US/docs/switches/datacenter/sw/nx-os/licensing/guide/b_Cisco_NX-OS_Licensing_Guide.html.

For the NX-OS end user agreement and copyright information, see *License and Copyright Information for Cisco NX-OS Software*, available at the following URL: http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/license_agreement/nx-ossw_lisns.html.

Configuration Guides

The configuration guides are available at the following URL:

http://www.cisco.com/en/US/products/ps11541/products_installation_and_configuration_guides_list.html

The documents in this category include:

- *Fundamentals Configuration Guide*
- *Interfaces Configuration Guide*
- *Layer 2 Switching Configuration Guide*
- *Multicast Configuration Guide*
- *Quality of Service Configuration Guide*
- *Security Configuration Guide*
- *System Management Configuration Guide*
- *Unicast Routing Configuration Guide*
- *Verified Scalability Guide for Cisco NX-OS*

Technical References

The technical references are available at the following URL:

http://www.cisco.com/en/US/products/ps11541/prod_technical_reference_list.html

Error and System Messages

The error and system message reference guides are available at the following URL:

http://www.cisco.com/en/US/products/ps11541/products_system_message_guides_list.html

Documentation Feedback

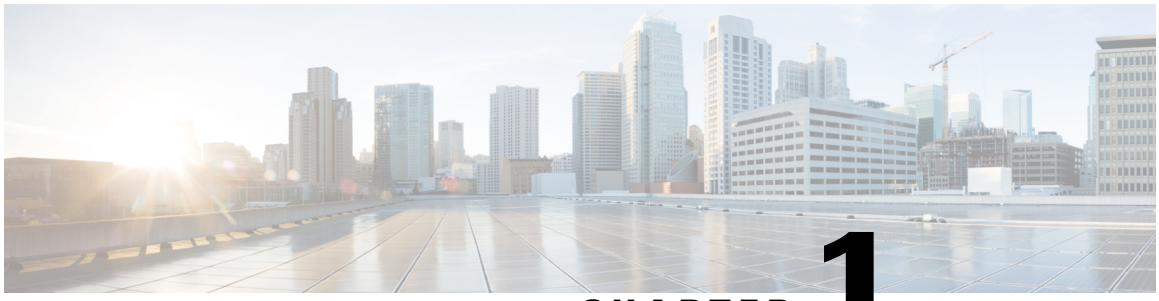
To provide technical feedback on this document, or to report an error or omission, please send your comments to nexus3k-docfeedback@cisco.com. We appreciate your feedback.

Obtaining Documentation and Submitting a Service Request

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

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

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



CHAPTER 1

New and Changed Information for this Release

The following table provides an overview of the significant changes to this guide for this current release. The table does not provide an exhaustive list of all changes made to the configuration guides or of the new features in this release.

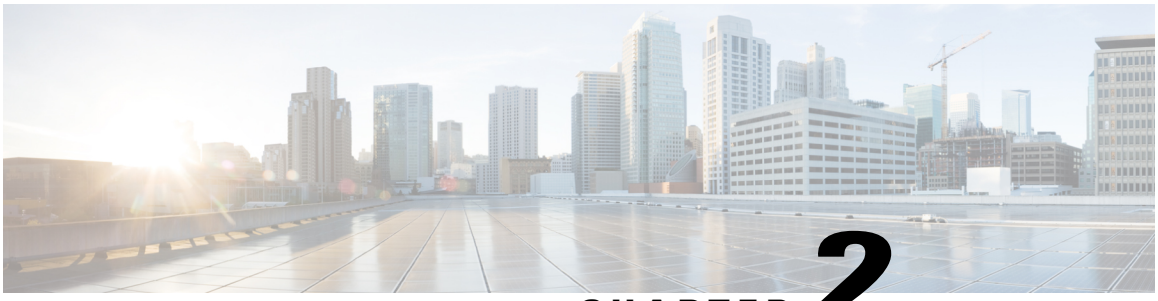
New and Changed Information for this Release

The following table provides an overview of the significant changes to this guide for this current release. The table does not provide an exhaustive list of all changes made to the configuration guides or of the new features in this release.

Table 1: New Features

Feature	Description	Where Documented
sFLOW	Allows the monitoring of real-time traffic in data networks.	Configuring sFLOW, on page 169

New and Changed Information for this Release



CHAPTER 2

Overview

This chapter contains the following sections:

System Management Features

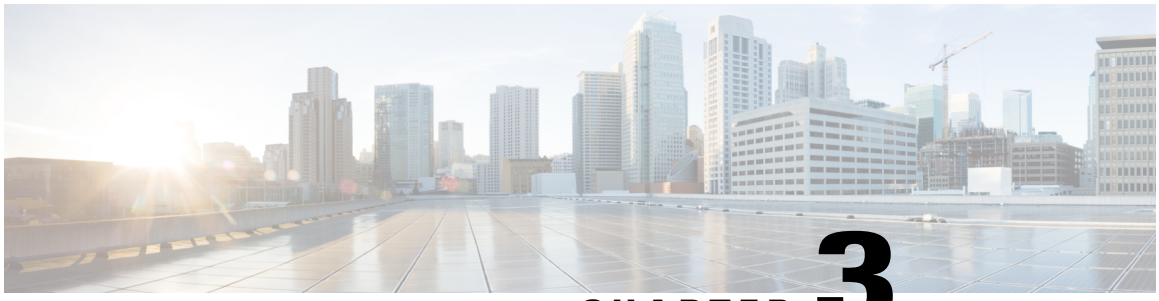
The system management features documented in this guide are described below:

Feature	Description
Switch Profiles	Configuration synchronization allows administrators to make configuration changes on one switch and have the system automatically synchronize the configuration to a peer switch. This feature eliminates misconfigurations and reduces the administrative overhead. The configuration synchronization mode (config-sync) allows users to create switch profiles to synchronize local and peer switch.
Cisco Fabric Services	The Cisco MDS NX-OS software uses the Cisco Fabric Services (CFS) infrastructure to enable efficient database distribution and to promote device flexibility. CFS simplifies SAN provisioning by automatically distributing configuration information to all switches in a fabric.
Precision Time Protocol	The Precision Time Protocol (PTP) is a time synchronization protocol for nodes distributed across a network. Its hardware timestamp feature provides greater accuracy than other time synchronization protocols such as Network Time Protocol (NTP).

Feature	Description
User Accounts and RBAC	User accounts and role-based access control (RBAC) allow you to define the rules for an assigned role. Roles restrict the authorization that the user has to access management operations. Each user role can contain multiple rules and each user can have multiple roles.
Session Manager	Session Manager allows you to create a configuration and apply it in batch mode after the configuration is reviewed and verified for accuracy and completeness.
Online Diagnostics	Cisco Generic Online Diagnostics (GOLD) define a common framework for diagnostic operations across Cisco platforms. The online diagnostic framework specifies the platform-independent fault-detection architecture for centralized and distributed systems, including the common diagnostics CLI and the platform-independent fault-detection procedures for boot-up and run-time diagnostics. The platform-specific diagnostics provide hardware-specific fault-detection tests and allow you to take appropriate corrective action in response to diagnostic test results.
System Message Logging	You can use system message logging to control the destination and to filter the severity level of messages that system processes generate. You can configure logging to a terminal session, a log file, and syslog servers on remote systems. System message logging is based on RFC 3164. For more information about the system message format and the messages that the device generates, see the *Cisco NX-OS System Messages Reference*.
Smart Call Home	Call Home provides an e-mail-based notification of critical system policies. Cisco NX-OS provides a range of message formats for optimal compatibility with pager services, standard e-mail, or XML-based automated parsing applications. You can use this feature to page a network support engineer, e-mail a Network Operations Center, or use Cisco Smart Call Home services to automatically generate a case with the Technical Assistance Center.

Feature	Description
Configuration Rollback	The configuration rollback feature allows users to take a snapshot, or user checkpoint, of the Cisco NX-OS configuration and then reapply that configuration to a switch at any point without having to reload the switch. A rollback allows any authorized administrator to apply this checkpoint configuration without requiring expert knowledge of the features configured in the checkpoint.
SNMP	The Simple Network Management Protocol (SNMP) is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.
RMON	RMON is an Internet Engineering Task Force (IETF) standard monitoring specification that allows various network agents and console systems to exchange network monitoring data. Cisco NX-OS supports RMON alarms, events, and logs to monitor Cisco NX-OS devices.
SPAN	The Switched Port Analyzer (SPAN) feature (sometimes called port mirroring or port monitoring) selects network traffic for analysis by a network analyzer. The network analyzer can be a Cisco SwitchProbe, a Fibre Channel Analyzer, or other Remote Monitoring (RMON) probes.

Feature	Description
ERSPAN	Encapsulated remote switched port analyzer (ERSPAN) is used to transport mirrored traffic in an IP network. ERSPAN supports source ports, source VLANs, and destinations on different switches, which provide remote monitoring of multiple switches across your network. ERSPAN uses a generic routing encapsulation (GRE) tunnel to carry traffic between switches. ERSPAN consists of an ERSPAN source session, routable ERSPAN GRE-encapsulated traffic, and an ERSPAN destination session. You separately configure ERSPAN source sessions and destination sessions on different switches. To configure an ERSPAN source session on one switch, you associate a set of source ports or VLANs with a destination IP address, ERSPAN ID number, and virtual routing and forwarding (VRF) name. To configure an ERSPAN destination session on another switch, you associate the destinations with the source IP address, the ERSPAN ID number, and a VRF name. The ERSPAN source session copies traffic from the source ports or source VLANs and forwards the traffic using routable GRE-encapsulated packets to the ERSPAN destination session. The ERSPAN destination session switches the traffic to the destinations.



CHAPTER 3

Using Cisco Fabric Services

This chapter contains the following sections:

Information About CFS

Some features in the Cisco Nexus Series switch require configuration synchronization with other switches in the network to function correctly. Synchronization through manual configuration at each switch in the network can be a tedious and error-prone process.

Cisco Fabric Services (CFS) provides a common infrastructure for automatic configuration synchronization in the network. It provides the transport function and a set of common services to the features. CFS has the ability to discover CFS-capable switches in the network and to discover feature capabilities in all CFS-capable switches.

Cisco Nexus Series switches support CFS message distribution over Fibre Channel and IPv4 or IPv6 networks. If the switch is provisioned with Fibre Channel ports, CFS over Fibre Channel is enabled by default while CFS over IP must be explicitly enabled.

The configuration synchronization feature has limited support for Cisco Nexus 3000 Series 5.0(3) version.

CFS provides the following features:

- Peer-to-peer protocol with no client-server relationship at the CFS layer.
- CFS message distribution over Fibre Channel and IPv4 networks.
- Three modes of distribution.
  - Coordinated distributions—Only one distribution is allowed in the network at any given time.

- Uncoordinated distributions—Multiple parallel distributions are allowed in the network except when a coordinated distribution is in progress.
- Unrestricted uncoordinated distributions—Multiple parallel distributions are allowed in the network in the presence of an existing coordinated distribution. Unrestricted uncoordinated distributions are allowed to run in parallel with all other types of distributions.

The following features are supported for CFS distribution over IP:

- One scope of distribution over an IP network:
  - Physical scope—The distribution spans the entire IP network.

The following features are supported for CFS distribution over Fibre Channel SANs:

- Three scopes of distribution over SAN fabrics.
  - Logical scope — The distribution occurs within the scope of a VSAN.
  - Physical scope — The distribution spans the entire physical topology.
  - Over a selected set of VSANs — Some features require configuration distribution over some specific VSANs. These features can specify to CFS the set of VSANs over which to restrict the distribution.
- Supports a merge protocol that facilitates the merge of feature configuration during a fabric merge event (when two independent SAN fabrics merge).

CFS Distribution

The CFS distribution functionality is independent of the lower layer transport. Cisco Nexus Series switches support CFS distribution over IP. Features that use CFS are unaware of the lower layer transport.

CFS Distribution Modes

CFS supports three distribution modes to accommodate different feature requirements:

- Uncoordinated Distribution
- Coordinated Distribution
- Unrestricted Uncoordinated Distributions

Only one mode is allowed at any given time.

Uncoordinated Distribution

Uncoordinated distributions are used to distribute information that is not expected to conflict with information from a peer. Parallel uncoordinated distributions are allowed for a feature.

Coordinated Distribution

Coordinated distributions allow only one feature distribution at a given time. CFS uses locks to enforce this feature. A coordinated distribution is not allowed to start if locks are taken for the feature anywhere in the network. A coordinated distribution consists of three stages:

- A network lock is acquired.
- The configuration is distributed and committed.
- The network lock is released.

Coordinated distribution has two variants:

- CFS driven —The stages are executed by CFS in response to feature request without intervention from the feature.
- Feature driven—The stages are under the complete control of the feature.

Coordinated distributions are used to distribute information that can be manipulated and distributed from multiple switches, for example, the port security configuration.

Unrestricted Uncoordinated Distributions

Unrestricted uncoordinated distributions allow multiple parallel distributions in the network in the presence of an existing coordinated distribution. Unrestricted uncoordinated distributions are allowed to run in parallel with all other types of distributions.

Verifying the CFS Distribution Status

The **show cfs status** command displays the status of CFS distribution on the switch:

```
switch# show cfs status
Distribution : Enabled
Distribution over IP : Enabled - mode IPv4
IPv4 multicast address : 239.255.70.83
IPv6 multicast address : ff15::efff:4653
Distribution over Ethernet : Enabled
```

CFS Support for Applications

CFS Application Requirements

All switches in the network must be CFS capable. Switches that are not CFS capable do not receive distributions, which results in part of the network not receiving the intended distribution. CFS has the following requirements:

- Implicit CFS usage—The first time that you issue a CFS task for a CFS-enabled application, the configuration modification process begins and the application locks the network.
- Pending database—The pending database is a temporary buffer to hold uncommitted information. The uncommitted changes are not applied immediately to ensure that the database is synchronized with the

database in the other switches in the network. When you commit the changes, the pending database overwrites the configuration database (also known as the active database or the effective database).

- CFS distribution enabled or disabled on a per-application basis—The default (enable or disable) for the CFS distribution state differs between applications. If CFS distribution is disabled for an application, that application does not distribute any configuration and does not accept a distribution from other switches in the network.
- Explicit CFS commit—Most applications require an explicit commit operation to copy the changes in the temporary buffer to the application database, to distribute the new database to the network, and to release the network lock. The changes in the temporary buffer are not applied if you do not perform the commit operation.

Enabling CFS for an Application

All CFS-based applications provide an option to enable or disable the distribution capabilities.

Applications have the distribution enabled by default.

The application configuration is not distributed by CFS unless distribution is explicitly enabled for that application.

Verifying Application Registration Status

The **show cfs application** command displays the applications that are currently registered with CFS. The first column displays the application name. The second column indicates whether the application is enabled or disabled for distribution (enabled or disabled). The last column indicates the scope of distribution for the application (logical, physical, or both).

Note The **show cfs application** command only displays applications registered with CFS. Conditional services that use CFS do not appear in the output unless these services are running.

```
switch# show cfs application

----------------------------------------------
 Application    Enabled   Scope
----------------------------------------------
 ntp            No        Physical-all
 fscm           Yes       Physical-fc
 rscn           No        Logical
 fctimer        No        Physical-fc
 syslogd        No        Physical-all
 callhome       No        Physical-all
 fcdomain       Yes       Logical
 device-alias   Yes       Physical-fc
Total number of entries = 8
```

The **show cfs application name** command displays the details for a particular application. It displays the enabled/disabled state, timeout as registered with CFS, merge capability (if it has registered with CFS for merge support), and the distribution scope.

```
switch# show cfs application name fscm
```

```
Enabled         : Yes
 Timeout        : 100s
 Merge Capable  : No
 Scope          : Physical-fc
```

Locking the Network

When you configure (first-time configuration) a feature (application) that uses the CFS infrastructure, that feature starts a CFS session and locks the network. When a network is locked, the switch software allows configuration changes to this feature only from the switch that holds the lock. If you make configuration changes to the feature from another switch, the switch issues a message to inform the user about the locked status. The configuration changes are held in a pending database by that application.

If you start a CFS session that requires a network lock but forget to end the session, an administrator can clear the session. If you lock a network at any time, your username is remembered across restarts and switchovers. If another user (on the same machine) tries to perform configuration tasks, that user's attempts are rejected.

Verifying CFS Lock Status

The **show cfs lock** command displays all the locks that are currently acquired by any application. For each application the command displays the application name and scope of the lock taken.

The **show cfs lock name** command displays the lock details for the specified application.

Committing Changes

A commit operation saves the pending database for all application peers and releases the lock for all switches.

The commit function does not start a session; only a lock function starts a session. However, an empty commit is allowed if configuration changes are not previously made. In this case, a commit operation results in a session that acquires locks and distributes the current database.

When you commit configuration changes to a feature using the CFS infrastructure, you receive a notification about one of the following responses:

- One or more external switches report a successful status—The application applies the changes locally and releases the network lock.
- None of the external switches report a successful state—The application considers this state a failure and does not apply the changes to any switch in the network. The network lock is not released.

You can commit changes for a specified feature by entering the **commit** command for that feature.

Discarding Changes

If you discard configuration changes, the application flushes the pending database and releases locks in the network. Both the abort and commit functions are supported only from the switch from which the network lock is acquired.

You can discard changes for a specified feature by using the **abort** command for that feature.

Saving the Configuration

Configuration changes that have not been applied yet (still in the pending database) are not shown in the running configuration. The configuration changes in the pending database overwrite the configuration in the effective database when you commit the changes.

Caution If you do not commit the changes, they are not saved to the running configuration.

Clearing a Locked Session

You can clear locks held by an application from any switch in the network to recover from situations where locks are acquired and not released. This function requires Admin permissions.

Caution Exercise caution when using this function to clear locks in the network. Any pending configurations in any switch in the network is flushed and lost.

CFS Regions

About CFS Regions

A CFS region is a user-defined subset of switches for a given feature or application in its physical distribution scope. When a network spans a vast geography, you might need to localize or restrict the distribution of certain profiles among a set of switches based on their physical proximity. CFS regions allow you to create multiple islands of distribution within the network for a given CFS feature or application. CFS regions are designed to restrict the distribution of a feature's configuration to a specific set or grouping of switches in a network.

Example Scenario

The Call Home application triggers alerts to network administrators when a situation arises or something abnormal occurs. When the network covers many geographies, and there are multiple network administrators who are each responsible for a subset of switches in the network, the Call Home application sends alerts to all network administrators regardless of their location. For the Call Home application to send message alerts selectively to network administrators, the physical scope of the application has to be fine tuned or narrowed down. You can achieve this scenario by implementing CFS regions.

CFS regions are identified by numbers ranging from 0 through 200. Region 0 is reserved as the default region and contains every switch in the network. You can configure regions from 1 through 200. The default region maintains backward compatibility.

If the feature is moved, that is, assigned to a new region, its scope is restricted to that region; it ignores all other regions for distribution or merging purposes. The assignment of the region to a feature has precedence in distribution over its initial physical scope.

You can configure a CFS region to distribute configurations for multiple features. However, on a given switch, you can configure only one CFS region at a time to distribute the configuration for a given feature. Once you assign a feature to a CFS region, its configuration cannot be distributed within another CFS region.

Managing CFS Regions

Creating CFS Regions

You can create a CFS region.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **cfs region** *region-id*	Creates a region.

Assigning Applications to CFS Regions

You can assign an application on a switch to a region.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **cfs region** *region-id*	Creates a region.
Step 3	switch(config-cfs-region)# *application*	Adds application(s) to the region. **Note** You can add any number of applications on the switch to a region. If you try adding an application to the same region more than once, you see the, "Application already present in the same region" error message.

The following example shows how to assign applications to a region:

```
switch# configure terminal
switch(config)# cfs region 1
switch(config-cfs-region)# ntp
switch(config-cfs-region)# callhome
```

Moving an Application to a Different CFS Region

You can move an application from one region to another region.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure**	Enters configuration mode.
Step 2	switch(config)# **cfs region** *region-id*	Enters CFS region configuration submode.
Step 3	switch(config-cfs-region)# *application*	Indicates application(s) to be moved from one region into another. **Note** If you try moving an application to the same region more than once, you see the, "Application already present in the same region" error message.

The following example shows how to move an application into Region 2 that was originally assigned to Region 1:

```
switch# configure terminal
switch(config)# cfs region 2
switch(config-cfs-region)# ntp
```

Removing an Application from a Region

Removing an application from a region is the same as moving the application back to the default region (Region 0), which brings the entire network into the scope of distribution for the application.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure**	Enters configuration mode.
Step 2	switch(config)# **cfs region** *region-id*	Enters CFS region configuration submode.
Step 3	switch(config-cfs-region)# **no** *application*	Removes application(s) that belong to the region.

Deleting CFS Regions

Deleting a region nullifies the region definition. All the applications bound by the region are released back to the default region.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure**	Enters configuration mode.
Step 2	switch(config)# **no cfs region** *region-id*	Deletes the region. **Note** You see the, "All the applications in the region will be moved to the default region" warning.

Configuring CFS over IP

Enabling CFS over IPv4

You can enable or disable CFS over IPv4.

Note CFS cannot distribute over both IPv4 and IPv6 from the same switch.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure**	Enters configuration mode.
Step 2	switch(config)# **cfs ipv4 distribute**	Globally enables CFS over IPv6 for all applications on the switch.
Step 3	switch(config)# **no cfs ipv4 distribute**	(Optional) Disables (default) CFS over IPv6 on the switch.

Enabling CFS over IPv6

You can enable or disable CFS over IPv6.

Note CFS cannot distribute over both IPv4 and IPv6 from the same switch.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure**	Enters configuration mode.
Step 2	switch(config)# **cfs ipv6 distribute**	Globally enables CFS over IPv6 for all applications on the switch.
Step 3	switch(config)# **no cfs ipv6 distribute**	(Optional) Disables (default) CFS over IPv6 on the switch.

Verifying the CFS Over IP Configuration

The following example show how to verify the CFS over IP configuration, use the **show cfs status** command.

```
switch# show cfs status
Distribution : Enabled
Distribution over IP : Enabled - mode IPv4
IPv4 multicast address : 239.255.70.83
IPv6 multicast address : ff15::efff:4653
```

Configuring IP Multicast Address for CFS over IP

All CFS over IP enabled switches with similar multicast addresses form one CFS over IP network. CFS protocol-specific distributions, such as the keepalive mechanism for detecting network topology changes, use the IP multicast address to send and receive information.

Note CFS distributions for application data use directed unicast.

Configuring IPv4 Multicast Address for CFS

You can configure a CFS over IP multicast address value for IPv4. The default IPv4 multicast address is 239.255.70.83.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure**	Enters configuration mode.
Step 2	switch(config)# **cfs ipv4 mcast-address** *ipv4-address*	Configures the IPv4 multicast address for CFS distribution over IPv4. The ranges of valid IPv4 addresses are 239.255.0.0 through 239.255.255.255 and 239.192/16 through 239.251/16.

	Command or Action	Purpose
Step 3	switch(config)# **no cfs ipv4 mcast-address** *ipv4-address*	(Optional) Reverts to the default IPv4 multicast address for CFS distribution over IPv4. The default IPv4 multicast address for CFS is 239.255.70.83.

Configuring IPv6 Multicast Address for CFS

You can configure a CFS over IP multicast address value for IPv6. The default IPv6 multicast address is ff13:7743:4653.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure**	Enters configuration mode.
Step 2	switch(config)# **cfs ipv6 mcast-address** *ipv4-address*	Configures the IPv6 multicast address for CFS distribution over IPv6. The range of valid IPv6 addresses is ff15::/16 (ff15::0000:0000 through ff15::ffff:ffff) and ff18::/16 (ff18::0000:0000 through ff18::ffff:ffff).
Step 3	switch(config)# **no cfs ipv6 mcast-address** *ipv4-address*	(Optional) Reverts to the default IPv6 multicast address for CFS distribution over IPv6. The default IPv6 multicast address for CFS over IP is ff15::efff:4653.

Verifying the IP Multicast Address Configuration for CFS over IP

The following example shows how to verify the IP multicast address configuration for CFS over IP, use the **show cfs status** command:

```
switch# show cfs status
Fabric distribution Enabled
IP distribution Enabled mode ipv4
IPv4 multicast address : 10.1.10.100
IPv6 multicast address : ff13::e244:4754
```

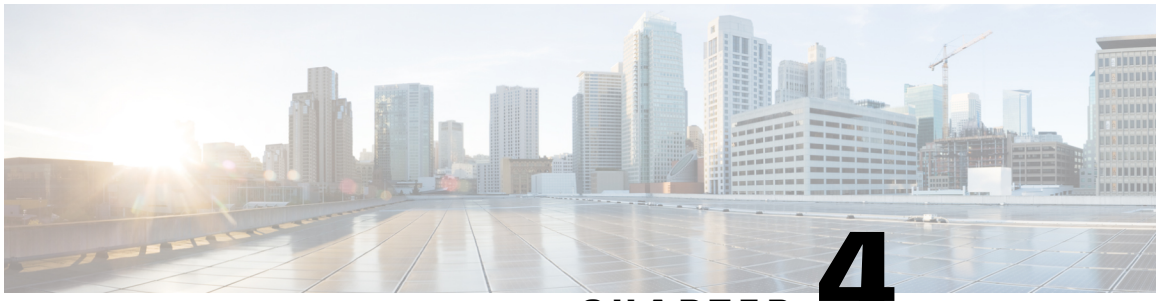
Default Settings for CFS

The following table lists the default settings for CFS configurations.

Table 2: Default CFS Parameters

Parameters	Default
CFS distribution on the switch	Enabled
Database changes	Implicitly enabled with the first configuration change
Application distribution	Differs based on application
Commit	Explicit configuration is required
CFS over IP	Disabled
IPv4 multicast address	239.255.70.83
IPv6 multicast address	ff15::efff:4653

The CISCO-CFS-MIB contains SNMP configuration information for any CFS-related functions. See the *Cisco Nexus 3000 Series MIBs Reference* available at the following URL: http://www.cisco.com/en/US/docs/switches/datacenter/nexus3000/sw/mib/reference/n3k_mib_ref.html.



CHAPTER 4

Configuring PTP

This chapter includes the following sections:

Information About PTP

PTP is a time synchronization protocol for nodes distributed across a network. Its hardware timestamp feature provides greater accuracy than other time synchronization protocols such as the Network Time Protocol (NTP).

A PTP system can consist of a combination of PTP and non-PTP devices. PTP devices include ordinary clocks, boundary clocks, and transparent clocks. Non-PTP devices include ordinary network switches, routers, and other infrastructure devices.

PTP is a distributed protocol that specifies how real-time PTP clocks in the system synchronize with each other. These clocks are organized into a master-slave synchronization hierarchy with the grandmaster clock, which is the clock at the top of the hierarchy, determining the reference time for the entire system. Synchronization is achieved by exchanging PTP timing messages, with the members using the timing information to adjust their clocks to the time of their master in the hierarchy. PTP operates within a logical scope called a PTP domain.

PTP Device Types

The following clocks are common PTP devices:

Ordinary clock

Communicates with the network based on a single physical port, similar to an end host. An ordinary clock can function as a grandmaster clock.

Boundary clock

Typically has several physical ports, with each port behaving like a port of an ordinary clock. However, each port shares the local clock, and the clock data sets are common to all ports. Each port decides its individual state, either master (synchronizing other ports connected to it) or slave (synchronizing to a downstream port), based on the best clock available to it through all of the other ports on the boundary clock. Messages related to synchronization and establishing the master-slave hierarchy terminate in the protocol engine of a boundary clock and are not forwarded.

Transparent clock

Forwards all PTP messages like an ordinary switch or router but measures the residence time of a packet in the switch (the time that the packet takes to traverse the transparent clock) and in some cases the link delay of the ingress port for the packet. The ports have no state because the transparent clock does not need to synchronize to the grandmaster clock.

There are two kinds of transparent clocks:

End-to-end transparent clock

Measures the residence time of a PTP message and accumulates the times in the correction field of the PTP message or an associated follow-up message.

Peer-to-peer transparent clock

Measures the residence time of a PTP message and computes the link delay between each port and a similarly equipped port on another node that shares the link. For a packet, this incoming link delay is added to the residence time in the correction field of the PTP message or an associated follow-up message.

Note PTP operates only in boundary clock mode. Cisco recommends deployment of a Grand Master Clock (GMC) upstream, with servers containing clocks requiring synchronization connected to the switch.

End-to-end transparent clock and peer-to-peer transparent clock modes are not supported.

PTP Process

The PTP process consists of two phases: establishing the master-slave hierarchy and synchronizing the clocks.

Within a PTP domain, each port of an ordinary or boundary clock follows this process to determine its state:

- Examines the contents of all received announce messages (issued by ports in the master state)
- Compares the data sets of the foreign master (in the announce message) and the local clock for priority, clock class, accuracy, and so on
- Determines its own state as either master or slave

After the master-slave hierarchy has been established, the clocks are synchronized as follows:

- The master sends a synchronization message to the slave and notes the time it was sent.
- The slave receives the synchronization message and notes the time it was received.
- The slave sends a delay-request message to the master and notes the time it was sent.
- The master receives the delay-request message and notes the time it was received.
- The master sends a delay-response message to the slave.
- The slave uses these timestamps to adjust its clock to the time of its master.

High Availability for PTP

Stateful restarts are not supported for PTP.

Licensing Requirements for PTP

PTP requires no license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you. For a complete explanation of the Cisco NX-OS licensing scheme, see the *Cisco NX-OS Licensing Guide*.

Guidelines and Limitations for PTP

- PTP operates only in boundary clock mode. End-to-end transparent clock and peer-to-peer transparent clock modes are not supported.
- PTP supports transport over User Datagram Protocol (UDP). Transport over Ethernet is not supported.
- PTP supports only multicast communication. Negotiated unicast communication is not supported.
- PTP is limited to a single domain per network.
- All management messages are forwarded on ports on which PTP is enabled. Handling management messages is not supported.
- PTP-capable ports do not identify PTP packets and do not time-stamp or redirect those packets unless you enable PTP on those ports.
- Cisco Nexus 3000 series switches should be synchronized from the neighboring master using a synchronization log interval that ranges from --2 to --5.
- Do not enable PTP on more than 10 ports if the synchronization log interval is set to -3 or lower on all of those ports.

Default Settings for PTP

The following table lists the default settings for PTP parameters.

Table 3: Default PTP Parameters

Parameters	Default
PTP	Disabled
PTP domain	0
PTP priority 1 value when advertising the clock	255
PTP priority 2 value when advertising the clock	255
PTP announce interval	1 log second
PTP sync interval	--2 log seconds
PTP announce timeout	3 announce intervals
PTP minimum delay request interval	0 log seconds
PTP VLAN	1

Configuring PTP

Configuring PTP Globally

You can enable or disable PTP globally on a device. You can also configure various PTP clock parameters to help determine which clock in the network has the highest priority to be selected as the grandmaster.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # [**no**] **feature ptp**	Enables or disables PTP on the device. **Note** Enabling PTP on the switch does not enable PTP on each interface.
Step 3	switch(config) # [**no**] **ptp source** *ip-address* [**vrf** *vrf*]	Configures the source IP address for all PTP packets. The *ip-address* can be in IPv4 or IPv6 format.
Step 4	switch(config) # [**no**] **ptp domain** *number*	(Optional) Configures the domain number to use for this clock. PTP domains allow you to use multiple independent PTP clocking subdomains on a single network. The range for the *number* is from 0 to 128.

	Command or Action	Purpose
Step 5	switch(config) # [**no**] **ptp priority1** *value*	(Optional) Configures the priority1 value to use when advertising this clock. This value overrides the default criteria (clock quality, clock class, and etc.) for best master clock selection. Lower values take precedence. The range for the *value* is from 0 to 255.
Step 6	switch(config) # [**no**] **ptp priority2** *value*	(Optional) Configures the priority2 value to use when advertising this clock. This value is used to decide between two devices that are otherwise equally matched in the default criteria. For example, you can use the priority2 value to give a specific switch priority over other identical switches. The range for the *value* is from 0 to 255.
Step 7	switch(config) # **show ptp brief**	(Optional) Displays the PTP status.
Step 8	switch(config) # **show ptp clock**	(Optional) Displays the properties of the local clock.
Step 9	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to configure PTP globally on the device, specify the source IP address for PTP communications, and configure a preference level for the clock:

```
switch# configure terminal
switch(config)# feature ptp
switch(config)# ptp source 10.10.10.1
switch(config)# ptp priority1 1
switch(config)# ptp priority2 1
switch(config)# show ptp brief
PTP port status
-----------------------
Port State
------- --------------
switch(config)# show ptp clock
PTP Device Type: Boundary clock
Clock Identity : 0:22:55:ff:ff:79:a4:c1
Clock Domain: 0
Number of PTP ports: 0
Priority1 : 1
Priority2 : 1
Clock Quality:
Class : 248
Accuracy : 254
Offset (log variance) : 65535
Offset From Master : 0
Mean Path Delay : 0
Steps removed : 0
Local clock time:Sun Jul 3 14:13:24 2011
switch(config)#
```

Configuring PTP on an Interface

After you globally enable PTP, it is not enabled on all supported interfaces by default. You must enable PTP interfaces individually.

Before You Begin

Make sure that you have globally enabled PTP on the switch and configured the source IP address for PTP communication.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **interface ethernet** *slot/port*	Specifies the interface on which you are enabling PTP and enters the interface configuration mode.
Step 3	switch(config-if) # [**no**] **feature ptp**	Enables or disables PTP on an interface.
Step 4	switch(config-if) # [**no**] **ptp announce** {**interval** *log seconds* | **timeout** *count*}	(Optional) Configures the interval between PTP announce messages on an interface or the number of PTP intervals before a timeout occurs on an interface. The range for the PTP announcement interval is from 0 to 4 seconds, and the range for the interval timeout is from 2 to 10.
Step 5	switch(config-if) # [**no**] **ptp delay request minimum interval** *log seconds*	(Optional) Configures the minimum interval allowed between PTP delay-request messages when the port is in the master state. The range is from -1 to 6 seconds.
Step 6	switch(config-if) # [**no**] **ptp sync interval** *log seconds*	(Optional) Configures the interval between PTP synchronization messages on an interface. The range for the PTP announcement interval is from -6 to 1 second.
Step 7	switch(config-if) # [**no**] **ptp vlan** *vlan-id*	(Optional) Specifies the VLAN for the interface where PTP is being enabled. You can only enable PTP on one VLAN on an interface. The range is from 1 to 4094.
Step 8	switch(config-if) # **show ptp brief**	(Optional) Displays the PTP status.

	Command or Action	Purpose
Step 9	switch(config-if) # **show ptp port interface** *interface slot/port*	(Optional) Displays the status of the PTP port.
Step 10	switch(config-if)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to configure PTP on an interface and configure the intervals for the announce, delay-request, and synchronization messages:

```
switch# configure terminal
switch(config)# interface ethernet 2/1
switch(config-if)# ptp
switch(config-if)# ptp announce interval 3
switch(config-if)# ptp announce timeout 2
switch(config-if)# ptp delay-request minimum interval 4
switch(config-if)# ptp sync interval -1
switch(config-if)# show ptp brief
PTP port status
-----------------------
Port State
------- --------------
Eth2/1 Master
switch(config-if)# show ptp port interface ethernet 2/1
PTP Port Dataset: Eth2/1
Port identity: clock identity: 0:22:55:ff:ff:79:a4:c1
Port identity: port number: 1028
PTP version: 2
Port state: Master
Delay request interval(log mean): 4
Announce receipt time out: 2
Peer mean path delay: 0
Announce interval(log mean): 3
Sync interval(log mean): -1
Delay Mechanism: End to End
Peer delay request interval(log mean): 0
switch(config-if)#
```

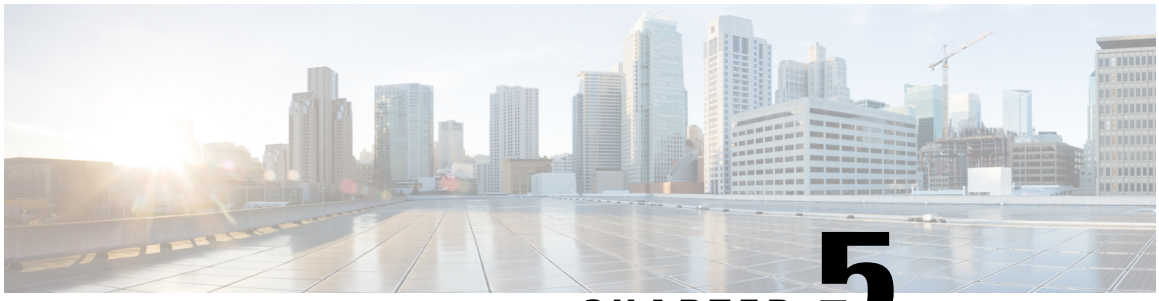
Verifying the PTP Configuration

Use one of the following commands to verify the configuration:

Table 4: PTP Show Commands

Command	Purpose
show ptp brief	Displays the PTP status.
show ptp clock	Displays the properties of the local clock, including clock identity.

Command	Purpose
show ptp clocks foreign-masters-record	Displays the state of foreign masters known to the PTP process. For each foreign master, the output displays the clock identity, basic clock properties, and whether the clock is being used as a grandmaster.
show ptp corrections	Displays the last few PTP corrections.
show ptp parent	Displays the properties of the PTP parent.
show ptp port interface ethernet *slot/port*	Displays the status of the PTP port on the switch.



CHAPTER 5

Configuring User Accounts and RBAC

This chapter contains the following sections:

Information About User Accounts and RBAC

Cisco Nexus Series switches use role-based access control (RBAC) to define the amount of access that each user has when the user logs into the switch.

With RBAC, you define one or more user roles and then specify which management operations each user role is allowed to perform. When you create a user account for the switch, you associate that account with a user role, which then determines what the individual user is allowed to do on the switch.

User Roles

User roles contain rules that define the operations allowed for the user who is assigned the role. Each user role can contain multiple rules and each user can have multiple roles. For example, if role1 allows access only to configuration operations, and role2 allows access only to debug operations, users who belong to both role1 and role2 can access configuration and debug operations. You can also limit access to specific VLANs, and interfaces.

The switch provides the following default user roles:

network-admin (superuser)

Complete read and write access to the entire switch.

network-operator

Complete read access to the switch.

Note If you belong to multiple roles, you can execute a combination of all the commands permitted by these roles. Access to a command takes priority over being denied access to a command. For example, suppose a user has RoleA, which denied access to the configuration commands. However, the user also has RoleB, which has access to the configuration commands. In this case, the user has access to the configuration commands.

Rules

The rule is the basic element of a role. A rule defines what operations the role allows the user to perform. You can apply rules for the following parameters:

Command

A command or group of commands defined in a regular expression.

Feature

Commands that apply to a function provided by the Cisco Nexus 3000 Series switch. Enter the **show role feature** command to display the feature names available for this parameter.

Feature group

Default or user-defined group of features. Enter the **show role feature-group** command to display the default feature groups available for this parameter.

These parameters create a hierarchical relationship. The most basic control parameter is the command. The next control parameter is the feature, which represents all commands associated with the feature. The last control parameter is the feature group. The feature group combines related features and allows you to easily manage the rules.

You can configure up to 256 rules for each role. The user-specified rule number determines the order in which the rules are applied. Rules are applied in descending order. For example, if a role has three rules, rule 3 is applied before rule 2, which is applied before rule 1.

User Role Policies

You can define user role policies to limit the switch resources that the user can access, or to limit access to interfaces and VLANs.

User role policies are constrained by the rules defined for the role. For example, if you define an interface policy to permit access to specific interfaces, the user does not have access to the interfaces unless you configure a command rule for the role to permit the **interface** command.

If a command rule permits access to specific resources (interfaces, VLANs), the user is permitted to access these resources, even if the user is not listed in the user role policies associated with that user.

User Account Configuration Restrictions

The following words are reserved and cannot be used to configure users:

- adm
- bin
- daemon
- ftp
- ftpuser
- games
- gdm
- gopher
- halt
- lp
- mail
- mailnull
- man
- mtsuser
- news
- nobody
- shutdown
- sync
- sys
- uucp
- xfs

User Password Requirements

Cisco Nexus 3000 Series passwords are case sensitive and can contain alphanumeric characters only. Special characters, such as the dollar sign ($) or the percent sign (%), are not allowed.

If a password is trivial (such as a short, easy-to-decipher password), the Cisco Nexus 3000 Series switch rejects the password. Be sure to configure a strong password for each user account. A strong password has the following characteristics:

- At least eight characters long
- Does not contain many consecutive characters (such as "abcd")
- Does not contain many repeating characters (such as "aaabbb")

- Does not contain dictionary words
- Does not contain proper names
- Contains both uppercase and lowercase characters
- Contains numbers

The following are examples of strong passwords:

- If2CoM18
- 2009AsdfLkj30
- Cb1955S21

Note For security reasons, user passwords do not display in the configuration files.

Guidelines and Limitations for User Accounts

Consider the following guidelines and limitations when configuring user accounts and RBAC:

- Up to 256 rules can be added to a user role.
- A maximum of 64 user roles can be assigned to a user account.
- You can assign a user role to more that one user account.
- Predefined roles such as network-admin, network-operator, and san-admin are not editable.
- Add, delete, and editing of rules is not supported for the SAN admin user role.
- The interface, VLAN, and/or VSAN scope cannot be changed for the SAN admin user role.

Note A user account must have at least one user role.

Configuring User Accounts

Note Changes to user account attributes do not take effect until the user logs in and creates a new session.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **show role**	(Optional) Displays the user roles available. You can configure other user roles, if necessary.
Step 3	switch(config) # **username** *user-id* [**password** *password*] [**expire** *date*] [**role** *role-name*]	Configures a user account. The *user-id* is a case-sensitive, alphanumeric character string with a maximum of 28 characters. The default *password* is undefined. **Note** If you do not specify a password, the user might not be able to log into the switch. The **expire** *date* option format is YYYY-MM-DD. The default is no expiry date.
Step 4	switch(config) # **exit**	Exists global configuration mode.
Step 5	switch# **show user-account**	(Optional) Displays the role configuration.
Step 6	switch# **copy running-config startup-config**	(Optional) Copies the running configuration to the startup configuration.

The following example shows how to configure a user account:

```
switch# configure terminal
switch(config)# username NewUser password 4Ty18Rnt
switch(config)# exit
switch# show user-account
```

Configuring RBAC

Creating User Roles and Rules

The rule number that you specify determines the order in which the rules are applied. Rules are applied in descending order. For example, if a role has three rules, rule 3 is applied before rule 2, which is applied before rule 1.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **role name** *role-name*	Specifies a user role and enters role configuration mode. The *role-name* argument is a case-sensitive, alphanumeric character string with a maximum of 16 characters.
Step 3	switch(config-role) # **rule** *number* {**deny** | **permit**} **command** *command-string*	Configures a command rule. The *command-string* can contain spaces and regular expressions. For example, interface ethernet * includes all Ethernet interfaces. Repeat this command for as many rules as needed.
Step 4	switch(config-role)# **rule** *number* {**deny** | **permit**} {**read** | **read-write**}	Configures a read-only or read-and-write rule for all operations.
Step 5	switch(config-role)# **rule** *number* {**deny** | **permit**} {**read** | **read-write**} **feature** *feature-name*	Configures a read-only or read-and-write rule for a feature. Use the **show role feature** command to display a list of features. Repeat this command for as many rules as needed.
Step 6	switch(config-role)# **rule** *number* {**deny** | **permit**} {**read** | **read-write**} **feature-group** *group-name*	Configures a read-only or read-and-write rule for a feature group. Use the **show role feature-group** command to display a list of feature groups. Repeat this command for as many rules as needed.
Step 7	switch(config-role)# **description** *text*	(Optional) Configures the role description. You can include spaces in the description.
Step 8	switch(config-role)# **end**	Exits role configuration mode.
Step 9	switch# **show role**	(Optional) Displays the user role configuration.
Step 10	switch# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to create user roles and specify rules:

```
switch# configure terminal
switch(config)# role name UserA
```

```
switch(config-role)# rule deny command clear users
switch(config-role)# rule deny read-write
switch(config-role)# description This role does not allow users to use clear commands
switch(config-role)# end
switch(config)# show role
```

Creating Feature Groups

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **role feature-group** *group-name*	Specifies a user role feature group and enters role feature group configuration mode. The *group-name* is a case-sensitive, alphanumeric character string with a maximum of 32 characters.
Step 3	switch(config) # **exit**	Exits global configuration mode.
Step 4	switch# **show role feature-group**	(Optional) Displays the role feature group configuration.
Step 5	switch# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to create a feature group:

```
switch# configure terminal
switch(config) # role feature-group group1
switch(config) # exit
switch# show role feature-group
switch# copy running-config startup-config
switch#
```

Changing User Role Interface Policies

You can change a user role interface policy to limit the interfaces that the user can access. Specify a list of interfaces that the role can access. You can specify it for as many interfaces as needed.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.

	Command or Action	Purpose
Step 2	switch(config) # **role name** *role-name*	Specifies a user role and enters role configuration mode.
Step 3	switch(config-role) # **interface policy deny**	Enters role interface policy configuration mode.
Step 4	switch(config-role-interface) # **permit interface** *interface-list*	Specifies a list of interfaces that the role can access. Repeat this command for as many interfaces as needed. For this command, you can specify Ethernet interfaces.
Step 5	switch(config-role-interface) # **exit**	Exits role interface policy configuration mode.
Step 6	switch(config-role) # **show role**	(Optional) Displays the role configuration.
Step 7	switch(config-role) # **copy running-config startup-config**	(Optional) Copies the running configuration to the startup configuration.

The following example shows how to change a user role interface policy to limit the interfaces that the user can access:

```
switch# configure terminal
switch(config)# role name UserB
switch(config-role)# interface policy deny
switch(config-role-interface)# permit interface ethernet 2/1
switch(config-role-interface)# permit interface fc 3/1
switch(config-role-interface)# permit interface vfc 30/1
```

Changing User Role VLAN Policies

You can change a user role VLAN policy to limit the VLANs that the user can access.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **role name** *role-name*	Specifies a user role and enters role configuration mode.
Step 3	switch(config-role)# **vlan policy deny**	Enters role VLAN policy configuration mode.
Step 4	switch(config-role-vlan # **permit vlan** *vlan-list*	Specifies a range of VLANs that the role can access. Repeat this command for as many VLANs as needed.

	Command or Action	Purpose
Step 5	switch(config-role-vlan) # **exit**	Exits role VLAN policy configuration mode.
Step 6	switch# **show role**	(Optional) Displays the role configuration.
Step 7	switch# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Verifying the User Accounts and RBAC Configuration

Use one of the following commands to verify the configuration:

Command	Purpose
show role [*role-name*]	Displays the user role configuration
show role feature	Displays the feature list.
show role feature-group	Displays the feature group configuration.
show startup-config security	Displays the user account configuration in the startup configuration.
show running-config security [**all**]	Displays the user account configuration in the running configuration. The **all** keyword displays the default values for the user accounts.
show user-account	Displays user account information.

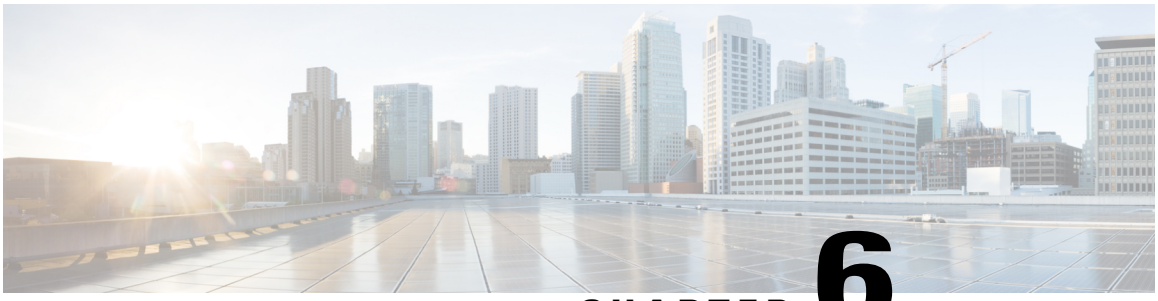
Configuring User Accounts Default Settings for the User Accounts and RBAC

The following table lists the default settings for user accounts and RBAC parameters.

Table 5: Default User Accounts and RBAC Parameters

Parameters	Default
User account password	Undefined.

Parameters	Default
User account expiry date	None.
Interface policy	All interfaces are accessible.
VLAN policy	All VLANs are accessible.
VFC policy	All VFCs are accessible.
VETH policy	All VETHs are accessible.



CHAPTER 6

Configuring Session Manager

This chapter contains the following sections:

Information About Session Manager

Session Manager allows you to implement your configuration changes in batch mode. Session Manager works in the following phases:

- Configuration session—Creates a list of commands that you want to implement in session manager mode.
- Validation—Provides a basic semantic check on your configuration. Cisco NX-OS returns an error if the semantic check fails on any part of the configuration.
- Verification—Verifies the configuration as a whole, based on the existing hardware and software configuration and resources. Cisco NX-OS returns an error if the configuration does not pass this verification phase.
- Commit— Cisco NX-OS verifies the complete configuration and implements the changes atomically to the device. If a failure occurs, Cisco NX-OS reverts to the original configuration.
- Abort—Discards the configuration changes before implementation.

You can optionally end a configuration session without committing the changes. You can also save a configuration session.

Guidelines and Limitations for Session Manager

Session Manager has the following configuration guidelines and limitations:

- Session Manager supports only the ACL feature.
- You can create up to 32 configuration sessions.
- You can configure a maximum of 20,000 commands across all sessions.

Configuring Session Manager

Creating a Session

You can create up to 32 configuration sessions.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure session** *name*	Creates a configuration session and enters session configuration mode. The name can be any alphanumeric string. Displays the contents of the session.
Step 2	switch(config-s)# **show configuration session** [*name*]	(Optional) Displays the contents of the session.
Step 3	switch(config-s)# **save** *location*	(Optional) Saves the session to a file. The location can be in bootflash or volatile.

Configuring ACLs in a Session

You can configure ACLs within a configuration session.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure session** *name*	Creates a configuration session and enters session configuration mode. The name can be any alphanumeric string.
Step 2	switch(config-s)# **ip access-list** *name*	Creates an ACL.
Step 3	switch(config-s-acl)# **permit** *protocol source destination*	(Optional) Adds a permit statement to the ACL.

	Command or Action	Purpose
Step 4	switch(config-s-acl)# **interface** *interface-type number*	Enters interface configuration mode.
Step 5	switch(config-s-if)# **ip port access-group** *name* **in**	Adds a port access group to the interface.
Step 6	switch# **show configuration session** [*name*]	(Optional) Displays the contents of the session.

Verifying a Session

To verify a session, use the following command in session mode:

Command	Purpose
switch(config-s)# **verify** [**verbose**]	Verifies the commands in the configuration session.

Committing a Session

To commit a session, use the following command in session mode:

Command	Purpose
switch(config-s)# **commit** [**verbose**]	Commits the commands in the configuration session.

Saving a Session

To save a session, use the following command in session mode:

Command	Purpose
switch(config-s)# **save** *location*	(Optional) Saves the session to a file. The location can be in bootflash or volatile.

Discarding a Session

To discard a session, use the following command in session mode:

Command	Purpose
switch(config-s)# **abort**	Discards the configuration session without applying the commands.

Configuration Example for Session Manager

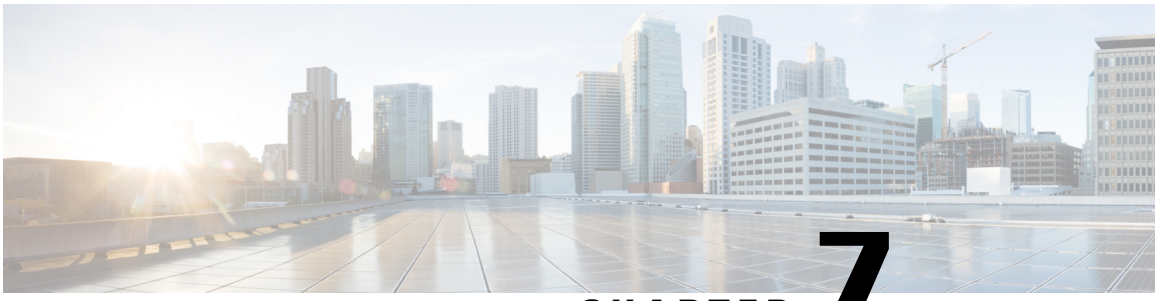
This example shows how to create a configuration session for ACLs:

```
switch# configure session name test2
switch(config-s)# ip access-list acl2
switch(config-s-acl)# permit tcp any any
switch(config-s-acl)# exit
switch(config-s)# interface Ethernet 1/4
switch(config-s-ip)# ip port access-group acl2 in
switch(config-s-ip)# exit
switch(config-s)# verify
switch(config-s)# exit
switch# show configuration session test2
```

Verifying the Session Manager Configuration

To verify Session Manager configuration information, perform one of the following tasks:

Command	Purpose
show configuration session [*name*]	Displays the contents of the configuration session.
show configuration session status [*name*]	Displays the status of the configuration session.
show configuration session summary	Displays a summary of all the configuration sessions.



CHAPTER 7

Configuring the Scheduler

This chapter contains the following sections:

Information About the Scheduler

The scheduler allows you to define and set a timetable for maintenance activities such as the following:

- Quality of service policy changes
- Data backup
- Saving a configuration

Jobs consist of a single command or multiple commands that define routine activities. Jobs can be scheduled one time or at periodic intervals.

The scheduler defines a job and its timetable as follows:

Job

A routine task or tasks defined as a command list and completed according to a specified schedule.

Schedule

The timetable for completing a job. You can assign multiple jobs to a schedule.

A schedule is defined as either periodic or one-time only:

- Periodic mode— A recurring interval that continues until you delete the job. You can configure the following types of intervals:
  - Daily— Job is completed once a day.
  - Weekly— Job is completed once a week.
  - Monthly—Job is completed once a month.
  - Delta—Job begins at the specified start time and then at specified intervals (days:hours:minutes).
- One-time mode—Job is completed only once at a specified time.

Remote User Authentication

Before starting a job, the scheduler authenticates the user who created the job. Because user credentials from a remote authentication are not retained long enough to support a scheduled job, you must locally configure the authentication passwords for users who create jobs. These passwords are part of the scheduler configuration and are not considered a locally configured user.

Before starting the job, the scheduler validates the local password against the password from the remote authentication server.

Scheduler Log Files

The scheduler maintains a log file that contains the job output. If the size of the job output is greater than the size of the log file, the output is truncated.

Licensing Requirements for the Scheduler

This feature does not require a license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you. For a complete explanation of the Cisco NX-OS licensing scheme, see the *Cisco NX-OS Licensing Guide*.

Guidelines and Limitations for the Scheduler

- The scheduler can fail if it encounters one of the following while performing a job:
  - If a feature license is expired when a job for that feature is scheduled.
  - If a feature is disabled at the time when a job for that feature is scheduled.

- Verify that you have configured the time. The scheduler does not apply a default timetable. If you create a schedule, assign jobs, and do not configure the time, the job is not started.
- While defining a job, verify that no interactive or disruptive commands (for example, **copy bootflash:** *file* **ftp:***URI*, **write erase**, and other similar commands) are specified because the job is started and conducted noninteractively.

Default Settings for the Scheduler

Table 6: Default Command Scheduler Parameters

Parameters	Default
Scheduler state	Disabled
Log file size	16 KB

Configuring the Scheduler

Enabling the Scheduler

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **feature scheduler**	Enables the scheduler.
Step 3	switch(config) # **show scheduler config**	(Optional) Displays the scheduler configuration.
Step 4	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to enable the scheduler:

```
switch# configure terminal
switch(config)# feature scheduler
switch(config)# show scheduler config
config terminal
   feature scheduler
   scheduler logfile size 16
end
switch(config)#
```

Defining the Scheduler Log File Size

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **scheduler logfile size** *value*	Defines the scheduler log file size in kilobytes. The range is from 16 to 1024. The default log file size is 16. **Note** If the size of the job output is greater than the size of the log file, the output is truncated.
Step 3	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to define the scheduler log file size:

```
switch# configure terminal
switch(config)# scheduler logfile size 1024
switch(config)#
```

Configuring Remote User Authentication

Remote users must authenticate with their clear text password before creating and configuring jobs.

Remote user passwords are always shown in encrypted form in the output of the **show running-config** command. The encrypted option (**7**) in the command supports the ASCII device configuration.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **scheduler aaa-authentication password** [**0** | **7**] *password*	Configures a password for the user who is currently logged in. To configure a clear text password, enter **0**. To configure an encrypted password, enter **7**.
Step 3	switch(config) # **scheduler aaa-authentication username** *name* **password** [**0** | **7**] *password*	Configures a clear text password for a remote user.

	Command or Action	Purpose
Step 4	switch(config) # **show running-config | include "scheduler aaa-authentication"**	(Optional) Displays the scheduler password information.
Step 5	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to configure a clear text password for a remote user called NewUser:

```
switch# configure terminal
switch(config) # scheduler aaa-authentication
username NewUser password z98y76x54b
switch(config) # copy running-config startup-config
switch(config) #
```

Defining a Job

Once a job is defined, you cannot modify or remove a command. To change the job, you must delete it and create a new one.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **scheduler job name** *name*	Creates a job with the specified name and enters job configuration mode. The *name* is restricted to 31 characters.
Step 3	switch(config-job) # *command1* **;** [*command2* **;***command3* **;** ...	Defines the sequence of commands for the specified job. You must separate commands with a space and a semicolon (;). The filename is created using the current time stamp and switch name.
Step 4	switch(config-job) # **show scheduler job** [*name*]	(Optional) Displays the job information. The *name* is restricted to 31 characters.
Step 5	switch(config-job) # **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to create a scheduler job named backup-cfg, save the running configuration to a file in bootflash, copy the file from bootflash to a TFTP server, and save the change to the startup configuration:

```
switch# configure terminal
switch(config) # scheduler job name backup-cfg
switch(config-job) # cli var name timestamp
$(timestamp) ;copy running-config
bootflash:/$(SWITCHNAME)-cfg.$(timestamp) ;copy
bootflash:/$(SWITCHNAME)-cfg.$(timestamp)
tftp://1.2.3.4/ vrf management
switch(config-job) # copy running-config startup-config
```

Deleting a Job

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **no scheduler job name** *name*	Deletes the specified job and all commands defined within it. The *name* is restricted to 31 characters.
Step 3	switch(config-job) # **show scheduler job** [*name*]	(Optional) Displays the job information.
Step 4	switch(config-job) # **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to delete a job called configsave:

```
switch# configure terminal
switch(config)# no scheduler job name configsave
switch(config-job)# copy running-config startup-config
switch(config-job)#
```

Defining a Timetable

You must configure a timetable. Otherwise, jobs will not be scheduled.

If you do not specify the time for the **time** commands, the scheduler assumes the current time. For example, if the current time is March 24, 2008, 22:00 hours,jobs are started as follows:

- For the **time start 23:00 repeat 4:00:00** command, the scheduler assumes a start time of March 24, 2008, 23:00 hours.
- For the **time daily 55** command, the scheduler assumes a start time every day at 22:55 hours.
- For the **time weekly 23:00** command, the scheduler assumes a start time every Friday at 23:00 hours.

- For the **time monthly 23:00** command, the scheduler assumes a start time on the 24th of every month at 23:00 hours.

Note The scheduler will not begin the next occurrence of a job before the last one completes. For example, you have scheduled a job to be completed at one-minute intervals beginning at 22:00; but the job requires two minutes to complete. The scheduler starts the first job at 22:00, completes it at 22:02, and then observes a one-minute interval before starting the next job at 22:03.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **scheduler schedule name** *name*	Creates a new scheduler and enters schedule configuration mode for that schedule. The *name* is restricted to 31 characters.
Step 3	switch(config-schedule) # **job name** *name*	Associates a job with this schedule. You can add multiple jobs to a schedule. The *name* is restricted to 31 characters.
Step 4	switch(config-schedule) # **time daily** *time*	Indicates the job starts every day at a designated time, specified as HH:MM.
Step 5	switch(config-schedule) # **time weekly** [[*day-of-week*:] *HH*:] *MM*	Indicates that the job starts on a specified day of the week. The day of the week is represented by an integer (for example, **1** for Sunday, **2** for Monday) or as an abbreviation (for example, **sun**, **mon**). The maximum length for the entire argument is 10 characters.
Step 6	switch(config-schedule) # **time monthly** [[*day-of-month*:] *HH*:] *MM*	Indicates that the job starts on a specified day each month. If you specify 29, 30, or 31, the job is started on the last day of each month.
Step 7	switch(config-schedule) # **time start** {**now repeat** *repeat-interval* | *delta-time* [**repeat** *repeat-interval*]}	Indicates the job starts periodically. The start-time format is [[[[yyyy:]mmm:]dd:]HH]:MM. • *delta-time*— Specifies the amount of time to wait after the schedule is configured before starting a job. • **now**— Specifies that the job starts two minutes from now. • **repeat** *repeat-interval*— Specifies the frequency at which the job is repeated.

	Command or Action	Purpose
Step 8	switch(config-schedule) # **show scheduler config**	(Optional) Displays the scheduler information.
Step 9	switch(config-schedule) # **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to define a timetable where jobs start on the 28th of each month at 23:00 hours:

```
switch# configure terminal
switch(config)# scheduler schedule name weekendbackupqos
switch(config-scheduler)# job name offpeakzoning
switch(config-scheduler)# time monthly 28:23:00
switch(config-scheduler)# copy running-config startup-config
switch(config-scheduler)#
```

Clearing the Scheduler Log File

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **clear scheduler logfile**	Clears the scheduler log file.

This example shows how to clear the scheduler log file:

```
switch# configure terminal
switch(config)# clear scheduler logfile
```

Disabling the Scheduler

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **no feature scheduler**	Disables the scheduler.
Step 3	switch(config) # **show scheduler config**	(Optional) Displays the scheduler configuration.

	Command or Action	Purpose
Step 4	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to disable the scheduler:

```
switch# configure terminal
switch(config) # no feature scheduler
switch(config) # copy running-config startup-config
switch(config) #
```

Verifying the Scheduler Configuration

Use one of the following commands to verify the configuration:

Table 7: Scheduler Show Commands

Command	Purpose
show scheduler config	Displays the scheduler configuration.
show scheduler job [**name** *name*]	Displays the jobs configured.
show scheduler logfile	Displays the contents of the scheduler log file.
show scheduler schedule [**name** *name*]	Displays the schedules configured.

Configuration Examples for the Scheduler

Creating a Scheduler Job

This example shows how to create a scheduler job that saves the running configuration to a file in bootflash and then copies the file from bootflash to a TFTP server (the filename is created using the current time stamp and switch name):

```
switch# configure terminal
switch(config)# scheduler job name backup-cfg
switch(config-job)# cli var name timestamp $(TIMESTAMP) ;copy running-config
bootflash:/$(SWITCHNAME)-cfg.$(timestamp) ;copy bootflash:/$(SWITCHNAME)-cfg.$(timestamp)
tftp://1.2.3.4/ vrf management
switch(config-job)# end
switch(config)#
```

Scheduling a Scheduler Job

This example shows how to schedule a scheduler job called backup-cfg to run daily at 1 a.m.:

```
switch# configure terminal
switch(config)# scheduler schedule name daily
switch(config-schedule)# job name backup-cfg
switch(config-schedule)# time daily 1:00
switch(config-schedule)# end
switch(config)#
```

Displaying the Job Schedule

This example shows how to display the job schedule:

```
switch# show scheduler schedule
Schedule Name       : daily
---------------------------
User Name           : admin
Schedule Type       : Run every day at 1 Hrs 00 Mins
Last Execution Time : Fri Jan 2 1:00:00 2009
Last Completion Time: Fri Jan 2 1:00:01 2009
Execution count     : 2
----------------------------------------------
     Job Name            Last Execution Status
----------------------------------------------
back-cfg                            Success (0)
switch(config)#
```

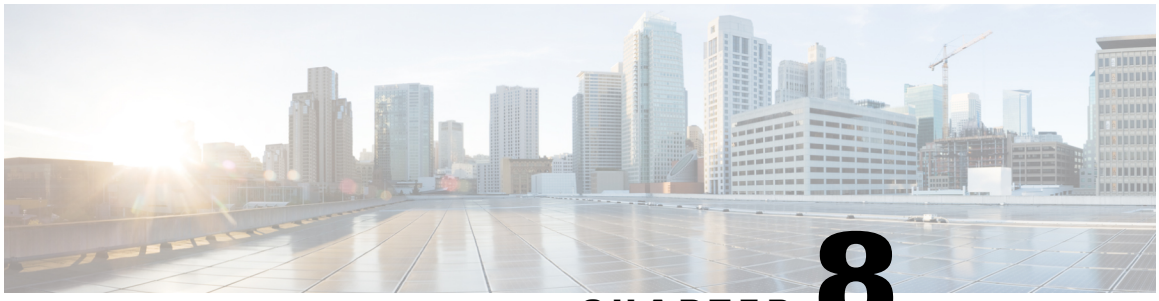
Displaying the Results of Running Scheduler Jobs

This example shows how to display the results of scheduler jobs that have been executed by the scheduler:

```
switch# show scheduler logfile
Job Name       : back-cfg                           Job Status: Failed (1)
Schedule Name  : daily                              User Name : admin
Completion time: Fri Jan 1  1:00:01 2009
-------------------------------- Job Output --------------------------------
`cli var name timestamp 2009-01-01-01.00.00`
`copy running-config bootflash:/$(HOSTNAME)-cfg.$(timestamp)`
`copy bootflash:/switch-cfg.2009-01-01-01.00.00 tftp://1.2.3.4/ vrf management `
copy: cannot access file '/bootflash/switch-cfg.2009-01-01-01.00.00'
==============================================================================
Job Name       : back-cfg                           Job Status: Success (0)
Schedule Name  : daily                              User Name : admin
Completion time: Fri Jan 2  1:00:01 2009
-------------------------------- Job Output --------------------------------
`cli var name timestamp 2009-01-02-01.00.00`
`copy running-config bootflash:/switch-cfg.2009-01-02-01.00.00`
`copy bootflash:/switch-cfg.2009--01-02-01.00.00 tftp://1.2.3.4/ vrf management `
Connection to Server Established.
[                     ]         0.50KBTrying to connect to tftp server......
[######               ]        24.50KB
TFTP put operation was successful
==============================================================================
switch#
```

Standards for the Scheduler

No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.



CHAPTER 8

Configuring Online Diagnostics

This chapter contains the following sections:

Information About Online Diagnostics

Online diagnostics provide verification of hardware components during switch bootup or reset, and they monitor the health of the hardware during normal switch operation.

Cisco Nexus Series switches support bootup diagnostics and runtime diagnostics. Bootup diagnostics include disruptive tests and nondisruptive tests that run during system bootup and system reset.

Runtime diagnostics (also known as health monitoring diagnostics) include nondisruptive tests that run in the background during normal operation of the switch.

Bootup Diagnostics

Bootup diagnostics detect faulty hardware before bringing the switch online. Bootup diagnostics also check the data path and control path connectivity between the supervisor and the ASICs. The following table describes the diagnostics that are run only during switch bootup or reset.

Table 8: Bootup Diagnostics

Diagnostic	Description
PCIe	Tests PCI express (PCIe) access.
NVRAM	Verifies the integrity of the NVRAM.
In band port	Tests connectivity of the inband port to the supervisor.

Diagnostic	Description
Management port	Tests the management port.
Memory	Verifies the integrity of the DRAM.

Bootup diagnostics also include a set of tests that are common with health monitoring diagnostics.

Bootup diagnostics log any failures to the onboard failure logging (OBFL) system. Failures also trigger an LED display to indicate diagnostic test states (on, off, pass, or fail).

You can configure Cisco Nexus 3000 Series switches to either bypass the bootup diagnostics or run the complete set of bootup diagnostics.

Health Monitoring Diagnostics

Health monitoring diagnostics provide information about the health of the switch. They detect runtime hardware errors, memory errors, software faults, and resource exhaustion.

Health monitoring diagnostics are nondisruptive and run in the background to ensure the health of a switch that is processing live network traffic.

The following table describes the health monitoring diagnostics for the switch.

Table 9: Health Monitoring Diagnostics Tests

Diagnostic	Description
LED	Monitors port and system status LEDs.
Power Supply	Monitors the power supply health state.
Temperature Sensor	Monitors temperature sensor readings.
Test Fan	Monitors the fan speed and fan control.

The following table describes the health monitoring diagnostics that also run during system boot or system reset.

Table 10: Health Monitoring and Bootup Diagnostics Tests

Diagnostic	Description
SPROM	Verifies the integrity of backplane and supervisor SPROMs.
Fabric engine	Tests the switch fabric ASICs.
Fabric port	Tests the ports on the switch fabric ASIC.

Diagnostic	Description
Forwarding engine	Tests the forwarding engine ASICs.
Forwarding engine port	Tests the ports on the forwarding engine ASICs.
Front port	Tests the components (such as PHY and MAC) on the front ports.

Expansion Module Diagnostics

During the switch bootup or reset, the bootup diagnostics include tests for the in-service expansion modules in the switch.

When you insert an expansion module into a running switch, a set of diagnostics tests are run. The following table describes the bootup diagnostics for an expansion module. These tests are common with the bootup diagnostics. If the bootup diagnostics fail, the expansion module is not placed into service.

Table 11: Expansion Module Bootup and Health Monitoring Diagnostics

Diagnostic	Description
SPROM	Verifies the integrity of backplane and supervisor SPROMs.
Fabric engine	Tests the switch fabric ASICs.
Fabric port	Tests the ports on the switch fabric ASIC.
Forwarding engine	Tests the forwarding engine ASICs.
Forwarding engine port	Tests the ports on the forwarding engine ASICs.
Front port	Tests the components (such as PHY and MAC) on the front ports.

Health monitoring diagnostics are run on in-service expansion modules. The following table describes the additional tests that are specific to health monitoring diagnostics for expansion modules.

Table 12: Expansion Module Health Monitoring Diagnostics

Diagnostic	Description
LED	Monitors port and system status LEDs.
Temperature Sensor	Monitors temperature sensor readings.

Configuring Online Diagnostics

You can configure the bootup diagnostics to run the complete set of tests, or you can bypass all bootup diagnostic tests for a faster module boot up time.

Note We recommend that you set the bootup online diagnostics level to complete. We do not recommend bypassing the bootup online diagnostics.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **diagnostic bootup level** [**complete** | **bypass**]	Configures the bootup diagnostic level to trigger diagnostics when the device boots, as follows: • complete—Performs all bootup diagnostics. This is the default value. • bypass—Does not perform any bootup diagnostics.
Step 3	switch# **show diagnostic bootup level**	(Optional) Displays the bootup diagnostic level (bypass or complete) that is currently in place on the switch.

The following example shows how to configure the bootup diagnostics level to trigger the complete diagnostics:

```
switch# configure terminal
switch(config)# diagnostic bootup level complete
```

Verifying the Online Diagnostics Configuration

To display online diagnostics configuration information, perform one of the following tasks:

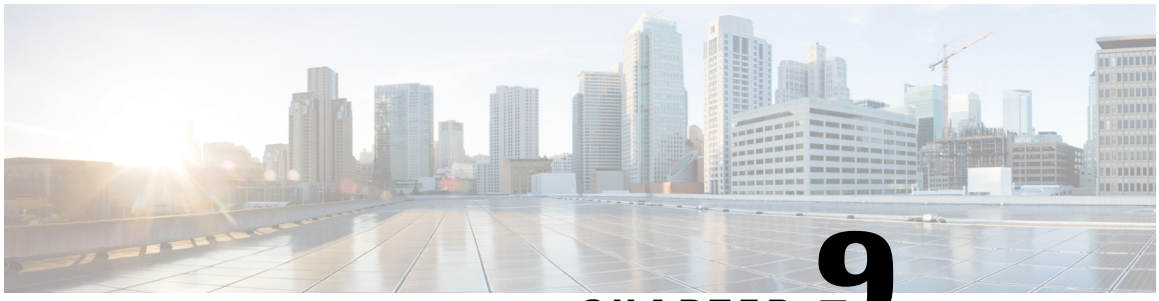
Command	Purpose
show diagnostic bootup level	Displays the bootup diagnostics level.
show diagnostic result module *slot*	Displays the results of the diagnostics tests.

Default Settings for Online Diagnostics

The following table lists the default settings for online diagnostics parameters.

Table 13: Default Online Diagnostics Parameters

Parameters	Default
Bootup diagnostics level	complete



CHAPTER 9

Configuring the Embedded Event Manager

This chapter contains the following sections:

Information About Embedded Event Manager

The ability to detect and handle critical events in the Cisco NX-OS system is important for high availability. The Embedded Event Manager (EEM) provides a central, policy-driven framework to detect and handle events in the system by monitoring events that occur on your device and taking action to recover or troubleshoot these events, based on your configuration..

EEM consists of three major components:

Event statements

Events to monitor from another Cisco NX-OS component that may require some action, workaround, or notification.

Action statements

An action that EEM can take, such as sending an e-mail or disabling an interface, to recover from an event.

Policies

An event paired with one or more actions to troubleshoot or recover from the event.

Without EEM, each individual component is responsible for detecting and handling its own events. For example, if a port flaps frequently, the policy of "putting it into errDisable state" is built into ETHPM.

Embedded Event Manager Policies

An EEM policy consists of an event statement and one or more action statements. The event statement defines the event to look for as well as the filtering characteristics for the event. The action statement defines the action EEM takes when the event occurs.

For example, you can configure an EEM policy to identify when a card is removed from the device and log the details related to the card removal. By setting up an event statement that tells the system to look for all instances of card removal and an then with an action statement that tells the system to log the details.

You can configure EEM policies using the command line interface (CLI) or a VSH script.

EEM gives you a device-wide view of policy management. Once EEM policies are configured, the corresponding actions are triggered. All actions (system or user-configured) for triggered events are tracked and maintained by the system.

Preconfigured System Policies

Cisco NX-OS has a number of preconfigured system policies. These system policies define many common events and actions for the device. System policy names begin with two underscore characters (__).

Some system policies can be overridden. In these cases, you can configure overrides for either the event or the action. The overrides that you configure take the place of the system policy.

Note Override policies must include an event statement. Override policies without event statements override all possible events for the system policy.

To view the preconfigured system polices and determine which polices you can override, use the **show event manager system-policy** command.

User-Created Policies

User-created policies allow you to customize EEM policies for your network. If a user policy is created for an event, actions in the policy are triggered only after EEM triggers the system policy actions related to the same event.

Log Files

The log file that contains data that is related to EEM policy matches is maintained in the event_archive_1 log file located in the /log/event_archive_1 directory.

Event Statements

Any device activity for which some action, such as a workaround or notification, is taken is considered an event by EEM. In many cases, events are related to faults in the device, such as when an interface or a fan malfunctions.

Event statements specify which event or events triggers a policy to run.

Tip YOu can configure EEM to trigger an EEM policy that is based on a combination of events by creating and differentiating multiple EEM events in the policy and then defining a combination of events to trigger a custom action.

EEM defines event filters so that only critical events or multiple occurrences of an event within a specified time period trigger an associated action.

Some commands or internal events trigger other commands internally. These commands are not visible, but will still match the event specification that triggers an action. You cannot prevent these commands from triggering an action, but you can check which event triggered an action.

Supported Events

EEM supports the following events in event statements:

- Counter events
- Fan absent events
- Fan bad events
- Memory thresholds events
- Events being used in overridden system policies.
- SNMP notification events
- Syslog events
- System manager events
- Temperature events
- Track events

Action Statements

Action statements describe the action that is triggered by a policy when an event occurs. Each policy can have multiple action statements. If no action is associated with a policy, EEM still observes events but takes no actions.

In order for triggered events to process default actions, you must configure the EEM policy to allow the default action. For example, if you match a CLI command in a match statement, you must add the event-default action statement to the EEM policy or EEM does not allow the command to execute.

Note When configuring action statements within your user policy or overriding policy, it is important that you confirm that action statements do not negate each other or adversely affect the associated system policy.

Supported Actions

EEM supports the following actions in action statements:

- Execute any CLI commands

- Update a counter
- Reload the device
- Generate a syslog message
- Generate an SNMP notification
- Use the default action for the system policy

VSH Script Policies

You can write policies in a VSH script, by using a text editor. Policies that are written using a VSH script have an event statement and action statement(s) just as other policies, and these policies can either augment or override system policies.

After you define your VSH script policy, copy it to the device and activate it.

Licensing Requirements for Embedded Event Manager

This feature does not require a license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you. For a complete explanation of the Cisco NX-OS licensing scheme, see the *Cisco NX-OS Licensing Guide*.

Prerequisites for Embedded Event Manager

You must have network-admin privileges to configure EEM.

Guidelines and Limitations for Embedded Event Manager

When you plan your EEM configuration, consider the following:

- The maximum number of configurable EEM policies is 500.
- Action statements within your user policy or overriding policy should not negate each other or adversely affect the associated system policy.
- If you want to allow a triggered event to process any default actions, you must configure the EEM policy to allow the default action. For example, if you match a command in a match statement, you must add the event-default action statement to the EEM policy or EEM does not allow the command to execute.
- An override policy that consists of an event statement and no action statement triggers no action and no notification of failures.
- An override policy without an event statement overrides all possible events in the system policy.
- In regular command expressions: all keywords must be expanded, and only the asterisk (*) symbol can be used for replace the arguments.
- EEM event correlation supports up to four event statements in a single policy. The event types can be the same or different, but only these event types are supported: cli, counter, snmp, syslog, and track.

- When more than one event statement is included in an EEM policy, each event statement must have a **tag** keyword with a unique tag argument.
- EEM event correlation does not override the system default policies.
- Default action execution is not supported for policies that are configured with tagged events.
- If your event specification matches a CLI pattern, you can use SSH-style wild card characters.

  For example, if you want to match all show commands, enter the **show *** command. Entering the **show . *** command does not work.
- If your event specification is a regular expression for a matching syslog message, you can use a proper regular expression.

  For example, if you want to detect ADMIN_DOWN events on any port where a syslog is generated, use **.ADMIN_DOWN.**. Entering the **ADMIN_DOWN** command does not work.
- In the event specification for a syslog, the regex does not match any syslog message that is generated as an action of an EEM policy.
- If an EEM event matches a **show** command in the CLI and you want the output for that **show** command to display on the screen (and to not be blocked by the EEM policy), you must specify the **event-default** command for the first action for the EEM policy.

Default Settings for Embedded Event Manager

Table 14: Default EEM Parameters

Parameters	Default
System Policies	Active

Configuring Embedded Event Manager

Defining an Environment Variable

Defining an environment variable is an optional step but is useful for configuring common values for repeated use in multiple policies.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

	Command or Action	Purpose
Step 2	**event manager environment** *variable-name variable-value* **Example:** `switch(config) # event manager environment emailto "admin@anyplace.com"`	Creates an environment variable for EEM. The *variable-name* can be any case-sensitive, alphanumeric string up to 29 characters. The *variable-value* can be any quoted case-sensitive, alphanumeric string up to 39 characters.
Step 3	**show event manager environment** {*variable-name* | **all**} **Example:** `switch(config) # show event manager environment all`	(Optional) Displays information about the configured environment variables.
Step 4	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

What to Do Next

Configure a User Policy.

Defining a User Policy Using the CLI

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**event manager applet** *applet-name*
	
	Example:
	`switch(config)# event manager applet monitorShutdown`
	`switch(config-applet)#`
		Registers the applet with EEM and enters applet configuration mode.
		The applet-name can be any case-sensitive, alphanumeric string up to 29 characters.

Step 3	**description** *policy-description* **Example:** `switch(config-applet)# description "Monitors interface shutdown."`	(Optional) Configures a descriptive string for the policy. The string can be any alphanumeric string up to 80 characters. Enclose the string in quotation marks.

	Command or Action	Purpose
Step 4	**event** *event-statement* **Example:** `switch(config-applet)# event cli match "shutdown"`	Configures the event statement for the policy.
Step 5	**tag** *tag* {**and** | **andnot** | **or**} *tag* [**and** | **andnot** | **or** {*tag*}] {**happens** *occurs* **in** *seconds*} **Example:** `switch(config-applet)# tag one or two happens 1 in 10000`	(Optional) Correlates multiple events in the policy. The range for the *occurs* argument is from 1 to 4294967295. The range for the *seconds* argument is from 0 to 4294967295 seconds.
Step 6	**action** *number*[.*number2*] *action-statement* **Example:** `switch(config-applet)# action 1.0 cli show interface e 3/1`	Configures an action statement for the policy. Repeat this step for multiple action statements.
Step 7	**show event manager policy-state** *name* [**module** *module-id*] **Example:** `switch(config-applet)# show event manager policy-state monitorShutdown`	(Optional) Displays information about the status of the configured policy.
Step 8	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

What to Do Next

Configure event statements and action statements.

Configuring Event Statements

Use one of the following commands in EEM configuration mode (config-applet) to configure an event statement:

Before You Begin

Define a user policy.

Procedure

	Command or Action	Purpose
Step 1	**event cli** [**tag** *tag*] **match** *expression* [**count** *repeats* | **time** *seconds* **Example:** `switch(config-applet) # event cli match "shutdown"`	Triggers an event if you enter a command that matches the regular expression. The **tag** *tag* keyword-argument pair identifies this specific event when multiple events are included in the policy. The *repeats* range is from 1 to 65000. The *time* range is from 0 to 4294967295, where 0 indicates no time limit.
Step 2	**event counter** [**tag** *tag*] **name** *counter* **entry-val** *entry* **entry-op** {**eq** | **ge** | **gt** | **le** | **lt** | **ne**} {**exit-val** *exit* **exit-op** {**eq** | **ge** | **gt** | **le** | **lt** | **ne**} **Example:** `switch(config-applet) # event counter name mycounter entry-val 20 gt`	Triggers an event if the counter crosses the entry threshold based on the entry operation. The event resets immediately. Optionally, you can configure the event to reset after the counter passes the exit threshold. The **tag** *tag* keyword-argument pair identifies this specific event when multiple events are included in the policy. The *counter* name can be any case-sensitive, alphanumeric string up to 28 characters. The *entry* and *exit* value ranges are from 0 to 2147483647.
Step 3	**event fanabsent** [**fan** *number*] **time** *seconds* **Example:** `switch(config-applet) # event fanabsent time 300`	Triggers an event if a fan is removed from the device for more than the configured time, in seconds. The *number* range is is from 1 to 1 and is module-dependent. The *seconds* range is from 10 to 64000.
Step 4	**event fanbad** [**fan** *number*] **time** *seconds* **Example:** `switch(config-applet) # event fanbad time 3000`	Triggers an event if a fan fails for more than the configured time, in seconds. The *number* range is module-dependent. The *seconds* range is from 10 to 64000.
Step 5	**event memory** {**critical** | **minor** | **severe**} **Example:** `switch(config-applet) # event memory critical`	Triggers an event if a memory threshold is crossed.
Step 6	**event policy-default count** *repeats* [**time** *seconds*] **Example:** `switch(config-applet) # event policy-default count 3`	Uses the event configured in the system policy. Use this option for overriding policies. The *repeats* range is from 1 to 65000. The *seconds* range is from 0 to 4294967295, where 0 indicates no time limit.

	Command or Action	Purpose
Step 7	**event snmp** [**tag** *tag*] **oid** *oid* **get-type** {**exact** | **next**} **entry-op** {**eq** | **ge** | **gt** | **le** | **lt** | **ne**} **entry-val** *entry* [**exit-comb** {**and** | **or**}]**exit-op** {**eq** | **ge** | **gt** | **le** | **lt** | **ne**} **exit-val** *exit* **exit-time** *time* **polling-interval** *interval* **Example:** `switch(config-applet) # event snmp oid 1.3.6.1.2.1.31.1.1.1.6 get-type next entry-op lt 300 entry-val 0 exit-op eq 400 exit-time 30 polling-interval 300`	Triggers an event if the SNMP OID crosses the entry threshold based on the entry operation. The event resets immediately, or optionally you can configure the event to reset after the counter passes the exit threshold. The OID is in dotted decimal notation. The **tag** *tag* keyword-argument pair identifies this specific event when multiple events are included in the policy. The *entry* and *exit* value ranges are from 0 to 18446744073709551615. The *time*, in seconds, is from 0 to 2147483647. The *interval*, in seconds, is from 0 to 2147483647.
Step 8	**event sysmgr memory** [**module** *module-num*] **major** *major-percent* **minor** *minor-percent* **clear** *clear-percent* **Example:** `switch(config-applet) # event sysmgr memory minor 80`	Triggers an event if the specified system manager memory threshold is exceeded. The *percent* range is from 1 to 99.
Step 9	**event temperature** [**module** *slot*] [**sensor** *number*] **threshold** {**any** | **down** | **up**} **Example:** `switch(config-applet) # event temperature module 2 threshold any`	Triggers an event if the temperature sensor exceeds the configured threshold. The *sensor* range is from 1 to 18.
Step 10	**event track** [**tag** *tag*] *object-number* **state** {**any** | **down** | **up** **Example:** `switch(config-applet) # event track 1 state down`	Triggers an event if the tracked object is in the configured state. The **tag** *tag* keyword-argument pair identifies this specific event when multiple events are included in the policy. The *object-number* range is from 1 to 500.

What to Do Next

Configure action statements.

If you have already configured action statements or choose not to, complete any of the optional tasks:

- Define a policy using a VSH script. Then, register and activate a VSH script policy.
- Configure memory thresholds
- Configure the syslog as an EEM publisher.
- Verify your EEM configuration.

Configuring Action Statements

You can configure an action by using one of the following commands in EEM configuration mode (config-applet):

Note If you want to allow a triggered event to process any default actions, you must configure the EEM policy to allow the default action.

For example, if you match a command in a match statement, you must add the event-default action statement to the EEM policy or EEM does not allow the command to execute. You can use the **terminal event-manager bypass** command to allow all EEM policies with matches to execute the command.

Before You Begin

Define a user policy.

Procedure

	Command or Action	Purpose
Step 1	**action** *number*[*.number2*] **cli** *command1*[*command2.*] [**local**] **Example:** `switch(config-applet) # action 1.0 cli "show interface e 3/1"`	Runs the configured commands. You can optionally run the commands on the module where the event occurred. The action label is in the format number1.number2. The *number* can be any number from 1 to 16 digits. The range for *number2* is from 0 to 9.
Step 2	**action** *number*[*.number2*] **counter name** *counter* **value** *val* **op** {**dec** | **inc** | **nop** | **set**} **Example:** `switch(config-applet) # action 2.0 counter name mycounter value 20 op inc`	Modifies the counter by the configured value and operation. The action label is in the format number1.number2. The *number* can be any number from 1 to 16 digits. The range for *number2* is from 0 to 9. The *counter* can be any case-sensitive, alphanumeric string up to 28 characters. The *val* can be an integer from 0 to 2147483647 or a substituted parameter.
Step 3	**action** *number*[*.number2*] **event-default** **Example:** `switch(config-applet) # action 1.0 event-default`	Completes the default action for the associated event. The action label is in the format number1.number2. The *number* can be any number from 1 to 16 digits. The range for *number2* is from 0 to 9.
Step 4	**action** *number*[*.number2*] **policy-default** **Example:** `switch(config-applet) # action 1.0 policy-default`	Completes the default action for the policy that you are overriding. The action label is in the format number1.number2. The *number* can be any number from 1 to 16 digits.

	Command or Action	Purpose
		The range for *number2* is from 0 to 9.
Step 5	**action** *number*[.*number2*] **reload** [**module** *slot* [**-** *slot*]] **Example:** `switch(config-applet) # action 1.0 reload module 3-5`	Forces one or more modules to the entire system to reload. The action label is in the format number1.number2. The *number* can be any number from 1 to 16 digits. The range for *number2* is from 0 to 9.
Step 6	**action** *number*[.*number2*] **snmp-trap** [**intdata1** *integer-data1*] [**intdata2** *integer-data2*] [**strdata** *string-data*] **Example:** `switch(config-applet) # action 1.0 snmp-trap strdata "temperature problem"`	Sends an SNMP trap with the configured data. The action label is in the format number1.number2. The *number* can be any number from 1 to 16 digits. The range for *number2* is from 0 to 9. The *data* elements can be any number up to 80 digits. The *string* can be any alphanumeric string up to 80 characters.
Step 7	**action** *number*[.*number2*] **syslog** [**priority** *prio-val*] **msg** *error-message* **Example:** `switch(config-applet) # action 1.0 syslog priority notifications msg "cpu high"`	Sends a customized syslog message at the configured priority. The action label is in the format number1.number2. The *number* can be any number from 1 to 16 digits. The range for *number2* is from 0 to 9. The *error-message* can be any quoted alphanumeric string up to 80 characters.

What to Do Next

Configure event statements.

If you have already configured event statements or choose not to, complete any of the optional tasks:

- Define a policy using a VSH script. Then, register and activate a VSH script policy.
- Configure memory thresholds
- Configure the syslog as an EEM publisher.
- Verify your EEM configuration.

Defining a Policy Using a VSH Script

This is an optional task. Complete the following steps if you are using a VSH script to write EEM policies:

Procedure

Step 1 In a text editor, list the commands that define the policy.

Step 2 Name the text file and save it.

Step 3 Copy the file to the following system directory: bootflash://eem/user_script_policies

What to Do Next

Register and activate a VSH script policy.

Registering and Activating a VSH Script Policy

This is an optional task. Complete the following steps if you are using a VSH script to write EEM policies.

Before You Begin

Define a policy using a VSH script and copy the file to the system directory.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**event manager policy** *policy-script* **Example:** `switch(config)# event manager policy moduleScript`	Registers and activates an EEM script policy. The *policy-script* can be any case-sensitive, alphanumeric string up to 29 characters.
Step 3	**event manager policy internal** *name* **Example:** `switch(config)# event manager policy internal moduleScript`	(Optional) Registers and activates an EEM script policy. The *policy-script* can be any case-sensitive alphanumeric string up to 29 characters.
Step 4	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

What to Do Next

Complete any of the following, depending on your system requirements:

- Configure memory thresholds.
- Configure the syslog as an EEM publisher.
- Verify your EEM configuration.

Overriding a System Policy

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**show event manager policy-state** *system-policy*
	
	Example:
	`switch(config-applet)# show event manager policy-state __ethpm_link_flap`
	`Policy __ethpm_link_flap`
	`Cfg count : 5`
	`Cfg time interval : 10.000000 (seconds)`
	`Hash default, Count 0`
		(Optional)
		Displays information about the system policy that you want to override, including thresholds. Use the **show event manager system-policy** command to find the system policy names.

Step 3	**event manager applet** *applet-name* **override** *system-policy*
	
	Example:
	`switch(config-applet)# event manager applet`
	`ethport override __ethpm_link_flap`
	`switch(config-applet)#`
		Overrides a system policy and enters applet configuration mode.
		The *applet-name* can be any case-sensitive, alphanumeric string up to 80 characters.
		The *system-policy* must be one of the system policies.

Step 4	**description** *policy-description* **Example:** `switch(config-applet)# description "Overrides link flap policy"`	Configures a descriptive string for the policy. The *policy-description* can be any case-sensitive, alphanumeric string up to 80 characters, but it must be enclosed in quotation marks.
Step 5	**event** *event-statement* **Example:** `switch(config-applet)# event policy-default count 2 time 1000`	Configures the event statement for the policy.
Step 6	**section** *number action-statement* **Example:** `switch(config-applet)# action 1.0 syslog priority warnings msg "Link is flapping."`	Configures an action statement for the policy. For multiple action statements, repeat this step.

	Command or Action	Purpose
Step 7	**show event manager policy-state** *name* **Example:** `switch(config-applet)# show event manager policy-state ethport`	(Optional) Displays information about the configured policy.
Step 8	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Configuring Memory Thresholds

Memory thresholds are used to trigger events and set whether the operating system should stop processes if it cannot allocate memory.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**system memory-thresholds minor** *minor* **severe** *severe* **critical** *critical* **Example:** `switch(config)# system memory-thresholds minor 60 severe 70 critical 80`	Configures the system memory thresholds that generate EEM memory events. The default values are as follows: • Minor—85 • Severe—90 • Critical—95 When these memory thresholds are exceeded, the system generates the following syslogs: • 2009 May 7 17:06:30 switch %$ VDC-1 %$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : MINOR • 2009 May 7 17:06:30 switch %$ VDC-1 %$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : SEVERE

	Command or Action	Purpose
		• 2009 May 7 17:06:30 switch %$ VDC-1 %$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : CRITICAL • 2009 May 7 17:06:35 switch %$ VDC-1 %$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : MINOR ALERT RECOVERED • 2009 May 7 17:06:35 switch %$ VDC-1 %$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : SEVERE ALERT RECOVERED • 2009 May 7 17:06:35 switch %$ VDC-1 %$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : CRITICAL ALERT RECOVERED
Step 3	**system memory-thresholds threshold critical no-process-kill** **Example:** `switch(config)# system memory-thresholds threshold critical no-process-kill`	(Optional) Configures the system to not stop processes when the memory cannot be allocated. The default value is to allow the system to stop processes, starting with the one that consumes the most memory.
Step 4	**show running-config | include "system memory"** **Example:** `switch(config)# show running-config | include "system memory"`	(Optional) Displays information about the system memory configuration.
Step 5	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

What to Do Next

Complete any of the following, depending on your system requirements:

- Configure the syslog as an EEM publisher.
- Verify your EEM configuration.

Configuring Syslog as an EEM Publisher

Configuring syslog as an EEM publisher allows you to monitor syslog messages from the switch.

Note The maximum number of searchable strings to monitor syslog messages is 10.

Before You Begin

- Confirm that EEM is available for registration by the syslog.
- Confirm that the syslog daemon is configured and executed.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**event manager applet** *applet-name*
	
	Example:
	`switch(config)# event manager applet abc`
	`switch (config-appliet)#`
		Registers an applet with EEM and enters applet configuration mode.

Step 3	**event syslog** [**tag** *tag*] {**occurs** *number* | **period** *seconds* | **pattern** *msg-text* | **priority** *priority*} **Example:** `switch(config-applet)# event syslog occurs 10`	Registers an applet with EEM and enters applet configuration mode.
Step 4	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

What to Do Next

Verify your EEM configuration.

Verifying the Embedded Event Manager Configuration

Use one of the following commands to verify the configuration:

Command	Purpose
show event manager environment [*variable-name* | **all**]	Displays information about the event manager environment variables.

Command	Purpose
show event manager event-types [*event* | **all** | **module** *slot*]	Displays information about the event manager event types.
show event manager history events [**detail**] [**maximum** *num-events*] [**severity** {**catastrophic** | **minor** | **moderate** | **severe**}]	Displays the history of events for all policies.
show event manager policy internal [*policy-name*] [**inactive**]	Displays information about the configured policies.
show event manager policy-state *policy-name*	Displays information about the policy state, including thresholds.
show event manager script system [*policy-name* | **all**]	Displays information about the script policies.
show event manager system-policy [**all**]	Displays information about the predefined system policies.
show running-config eem	Displays information about the running configuration for EEM.
show startup-config eem	Displays information about the startup configuration for EEM.

Configuration Examples for Embedded Event Manager

The following example shows how to override the __lcm_module_failure system policy by changing the threshold for only module 3 hitless upgrade failures. It also sends a syslog message. The settings in the system policy, __lcm_module_failure, apply in all other cases.

```
event manager applet example2 override __lcm_module_failure
event module-failure type hitless-upgrade-failure module 3 count 2
   action 1 syslog priority errors msg module 3 "upgrade is not a hitless upgrade!"
   action 2 policy-default
```

The following example shows how to override the __ethpm_link_flap system policy and shut down the interface:

```
event manager applet ethport override __ethpm_link_flap
   event policy-default count 2 time 1000
   action 1 cli conf t
   action 2 cli int et1/1
   action 3 cli no shut
```

The following example shows how to create an EEM policy that allows the command to execute but triggers an SNMP notification when a user enters configuration mode on the device:

```
event manager applet TEST
   event cli match "conf t"
   action 1.0 snmp-trap strdata "Configuration change"
   action 2.0 event-default
```

Note You must add the **event-default** action statement to the EEM policy or EEM does not allow the command to execute.

The following example shows how to correlate multiple events in an EEM policy and execute the policy based on a combination of the event triggers. In this example, the EEM policy is triggered if one of the specified syslog patterns occurs within 120 seconds.

```
event manager applet eem-correlate
   event syslog tag one pattern "copy bootflash:.* running-config.*"
   event syslog tag two pattern "copy run start"
   event syslog tag three pattern "hello"
   tag one or two or three happens 1 in 120
   action 1.0 reload module 1
```

Additional References

Related Documents

Related Topic	Document Title
EEM commands	*Cisco Nexus 3000 Series NX-OS System Management Command Reference*

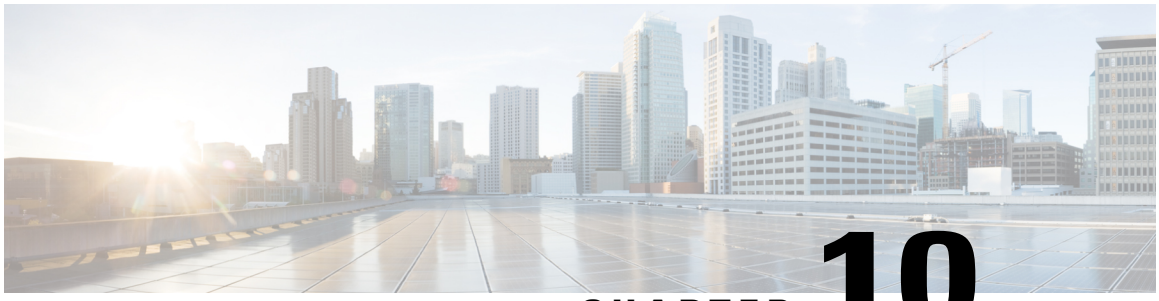
Standards

There are no new or modified standards supported by this feature, and support for existing standards has not been modified by this feature.

Feature History for EEM

Table 15: Feature History for EEM

Feature Name	Release	Feature Information
EEM	5.0(3)U3(1)	Feature added.



CHAPTER 10

Configuring System Message Logging

This chapter contains the following sections:

Information About System Message Logging

You can use system message logging to control the destination and to filter the severity level of messages that system processes generate. You can configure logging to terminal sessions, a log file, and syslog servers on remote systems.

System message logging is based on RFC 3164. For more information about the system message format and the messages that the device generates, see the *Cisco NX-OS System Messages Reference*.

By default, the Cisco Nexus 3000 Series switch outputs messages to terminal sessions.

By default, the switch logs system messages to a log file.

The following table describes the severity levels used in system messages. When you configure the severity level, the system outputs messages at that level and lower.

Table 16: System Message Severity Levels

Level	Description
0 – emergency	System unusable
1 – alert	Immediate action needed
2 – critical	Critical condition

Level	Description
3 – error	Error condition
4 – warning	Warning condition
5 – notification	Normal but significant condition
6 – informational	Informational message only
7 – debugging	Appears during debugging only

The switch logs the most recent 100 messages of severity 0, 1, or 2 to the NVRAM log. You cannot configure logging to the NVRAM.

You can configure which system messages should be logged based on the facility that generated the message and its severity level.

Syslog Servers

Syslog servers run on remote systems that are configured to log system messages based on the syslog protocol. You can configure the Cisco Nexus Series switch to sends logs to up to eight syslog servers.

To support the same configuration of syslog servers on all switches in a fabric, you can use Cisco Fabric Services (CFS) to distribute the syslog server configuration.

Note When the switch first initializes, messages are sent to syslog servers only after the network is initialized.

Licensing Requirements for System Message Logging

Product	License Requirement
Cisco NX-OS	System message logging requires no license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you. For a complete explanation of the Cisco NX-OS licensing scheme, see the *Cisco NX-OS Licensing Guide*.

Guidelines and Limitations for System Message Logging

System messages are logged to the console and the logfile by default.

Default Settings for System Message Logging

The following table lists the default settings for system message logging parameters.

Table 17: Default System Message Logging Parameters

Parameters	Default
Console logging	Enabled at severity level 2
Monitor logging	Enabled at severity level 2
Log file logging	Enabled to log messages at severity level 5
Module logging	Enabled at severity level 5
Facility logging	Enabled
Time-stamp units	Seconds
Syslog server logging	Disabled
Syslog server configuration distribution	Disabled

Configuring System Message Logging

Configuring System Message Logging to Terminal Sessions

You can configure the switch to log messages by their severity level to console, Telnet, and Secure Shell sessions.

By default, logging is enabled for terminal sessions.

Procedure

	Command or Action	Purpose
Step 1	switch# **terminal monitor**	Copies syslog messages from the console to the current terminal session.
Step 2	switch# **configure terminal**	Enters configuration mode.
Step 3	switch(config)# **logging console** [*severity-level*]	Enables the switch to log messages to the console session based on a specified severity level or higher (a lower number value indicates a higher severity level). Severity levels range from 0 to 7:

	Command or Action	Purpose
		• 0 – emergency • 1 – alert • 2 – critical • 3 – error • 4 – warning • 5 – notification • 6 – informational • 7 – debugging If the severity level is not specified, the default of 2 is used.
Step 4	switch(config)# **no logging console** [*severity-level*]	(Optional) Disables logging messages to the console.
Step 5	switch(config)# **logging monitor** [*severity-level*]	Enables the switch to log messages to the monitor based on a specified severity level or higher (a lower number value indicates a higher severity level). Severity levels range from 0 to 7: • 0 – emergency • 1 – alert • 2 – critical • 3 – error • 4 – warning • 5 – notification • 6 – informational • 7 – debugging If the severity level is not specified, the default of 2 is used. The configuration applies to Telnet and SSH sessions.
Step 6	switch(config)# **no logging monitor** [*severity-level*]	(Optional) Disables logging messages to telnet and SSH sessions.
Step 7	switch# **show logging console**	(Optional) Displays the console logging configuration.
Step 8	switch# **show logging monitor**	(Optional) Displays the monitor logging configuration.

	Command or Action	Purpose
Step 9	switch# **copy running-config startup-config**	(Optional) Copies the running configuration to the startup configuration.

The following example shows how to configure a logging level of 3 for the console:

```
switch# configure terminal
switch(config)# logging console 3
```

The following example shows how to display the console logging configuration:

```
switch# show logging console
Logging console:                enabled (Severity: error)
```

The following example shows how to disable logging for the console:

```
switch# configure terminal
switch(config)# no logging console
```

The following example shows how to configure a logging level of 4 for the terminal session:

```
switch# terminal monitor
switch# configure terminal
switch(config)# logging monitor 4
```

The following example shows how to display the terminal session logging configuration:

```
switch# show logging monitor
Logging monitor:                enabled (Severity: warning)
```

The following example shows how to disable logging for the terminal session:

```
switch# configure terminal
switch(config)# no logging monitor
```

Configuring System Message Logging to a File

You can configure the switch to log system messages to a file. By default, system messages are logged to the file log:messages.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **logging logfile** *logfile-name severity-level* [**size** *bytes*]	Configures the name of the log file used to store system messages and the minimum severity level to log. You can optionally specify a maximum file size. The default severity level is 5 and the file size is 4194304. Severity levels range from 0 to 7: • 0 – emergency • 1 – alert

	Command or Action	Purpose
		• 2 – critical • 3 – error • 4 – warning • 5 – notification • 6 – informational • 7 – debugging The file size is from 4096 to 10485760 bytes.
Step 3	switch(config)# **no logging logfile** [*logfile-name severity-level* [**size** *bytes*]]	(Optional) Disables logging to the log file. You can optionally specify a maximum file size. The default severity level is 5 and the file size is 4194304.
Step 4	switch# **show logging info**	(Optional) Displays the logging configuration. You can optionally specify a maximum file size. The default severity level is 5 and the file size is 4194304.
Step 5	switch# **copy running-config startup-config**	(Optional) Copies the running configuration to the startup configuration.

The following example shows how to configure a switch to log system messages to a file:

```
switch# configure terminal
switch(config)# logging logfile my_log 6 size 4194304
```

The following example shows how to display the logging configuration (some of the output has been removed for brevity):

```
switch# show logging info
Logging console:                enabled (Severity: debugging)
Logging monitor:                enabled (Severity: debugging)

Logging timestamp:              Seconds
Logging server:                 disabled
Logging logfile:                enabled
        Name - my_log: Severity - informational Size - 4194304
Facility        Default Severity        Current Session Severity
--------        ----------------        ------------------------
aaa                     3                       3
aclmgr                  3                       3
afm                     3                       3
altos                   3                       3
auth                    0                       0
authpriv                3                       3
bootvar                 5                       5
callhome                2                       2
capability              2                       2
cdp                     2                       2
cert_enroll             2                       2
...
```

Configuring Module and Facility Messages Logging

You can configure the severity level and time-stamp units of messages logged by modules and facilities.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **logging module** [*severity-level*]	Enables module log messages that have the specified severity level or higher. Severity levels range from 0 to 7: • 0 – emergency • 1 – alert • 2 – critical • 3 – error • 4 – warning • 5 – notification • 6 – informational • 7 – debugging If the severity level is not specified, the default of 5 is used.
Step 3	switch(config)# **logging level** *facility severity-level*	Enables logging messages from the specified facility that have the specified severity level or higher. Severity levels from 0 to 7: • 0 – emergency • 1 – alert • 2 – critical • 3 – error • 4 – warning • 5 – notification • 6 – informational • 7 – debugging To apply the same severity level to all facilities, use the all facility. For defaults, see the **show logging level** command.
Step 4	switch(config)# **no logging module** [*severity-level*]	(Optional) Disables module log messages.

	Command or Action	Purpose
Step 5	switch(config)# **no logging level** [*facility severity-level*]	(Optional) Resets the logging severity level for the specified facility to its default level. If you do not specify a facility and severity level, the switch resets all facilities to their default levels.
Step 6	switch# **show logging module**	(Optional) Displays the module logging configuration.
Step 7	switch# **show logging level** [*facility*]	(Optional) Displays the logging level configuration and the system default level by facility. If you do not specify a facility, the switch displays levels for all facilities.
Step 8	switch# **copy running-config startup-config**	(Optional) Copies the running configuration to the startup configuration.

The following example shows how to configure the severity level of module and specific facility messages:

```
switch# configure terminal
switch(config)# logging module 3
switch(config)# logging level aaa 2
```

Configuring Logging Timestamps

You can configure the time-stamp units of messages logged by the Cisco Nexus Series switch.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **logging timestamp** {**microseconds** | **milliseconds** | **seconds**}	Sets the logging time-stamp units. By default, the units are seconds.
Step 3	switch(config)# **no logging timestamp** {**microseconds** | **milliseconds** | **seconds**}	(Optional) Resets the logging time-stamp units to the default of seconds.
Step 4	switch# **show logging timestamp**	(Optional) Displays the logging time-stamp units configured.
Step 5	switch# **copy running-config startup-config**	(Optional) Copies the running configuration to the startup configuration.

The following example shows how to configure the time-stamp units of messages:

```
switch# configure terminal
switch(config)# logging timestamp milliseconds
switch(config)# exit
switch# show logging timestamp
Logging timestamp:              Milliseconds
```

Configuring the ACL Logging Cache

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)#**logging ip access-list cache entries** *num_entries*	Sets the maximum number of log entries cached in software. The range is from 0 to 1000000 entries. The default value is 8000 entries.
Step 3	switch(config)# **logging ip access-list cache interval** *seconds*	Sets the number of seconds between log updates. Also if an entry is inactive for this duration, it is removed from the cache. The range is from 5 to 86400 seconds. The default value is 300 seconds.
Step 4	switch(config)# **logging ip access-list cache threshold** *num_packets*	Sets the number of packet matches before an entry is logged. The range is from 0 to 1000000 packets. The default value is 0 packets, which means that logging is not triggered by the number of packet matches.
Step 5	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

The following example show how to set the maximum number of log entries to 5000, the interval to 120 seconds, and the threshold to 500000:

```
switch# configure terminal
switch(config)# logging ip access-list cache entries 5000
switch(config)# logging ip access-list cache interval 120
switch(config)# logging ip access-list cache threshold 500000
switch(config)# copy running-config startup-config
```

Applying ACL Logging to an Interface

You can apply ACL logging only on the mgmt0 interface.

Before You Begin

- Create an IP access list with at least one access control entry (ACE) configured for logging.

- Configure the ACL logging cache.
- Configure the ACL log match level.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **interface mgmt0**	Specifies the mgmt0 interface.
Step 3	switch(config-if)# **ip access-group** *name* **in**	Enables ACL logging on ingress traffic for the specified interface.
Step 4	switch(config-if)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

The following example shows how to apply the mgmt0 interface with the logging specified in acl1 for all ingress traffic:

```
switch# configure terminal
switch(config)# interface mgmt0
switch(config-if)# ip access-group acl1 in
switch(config-if)# copy running-config startup-config
```

Configuring the ACL Log Match Level

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **acllog match-log-level** *number*	Specifies the logging level to match for entries to be logged in the ACL log (acllog). The *number* is a value from 0 to 7. The default is 6. **Note** For log messages to be entered in the logs, the logging level for the ACL log facility (acllog) and the logging severity level for the logfile must be greater than or equal to the ACL log match log level setting. For more information, see Configuring Module and Facility Messages Logging, on page 83 and Configuring System Message Logging to a File, on page 81.

	Command or Action	Purpose
Step 3	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Configuring Syslog Servers

You can configure up to eight syslog servers that reference remote systems where you want to log system messages.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**logging server** *host* [*severity-level* [**use-vrf** *vrf-name* [**facility** *facility*]]] **Example:** `switch(config)# logging server 172.28.254.254 5 use-vrf default facility local3`	Configures a host to receive syslog messages. • The *host* argument identifies the hostname or the IPv4 or IPv6 address of the syslog server host. • The *severity-level* argument limits the logging of messages to the syslog server to a specified level. Severity levels range from 0 to 7. See Table 16: System Message Severity Levels , on page 77. • The **use vrf** *vrf-name* keyword and argument identify the *default* or *management* values for the virtual routing and forwarding (VRF) name. If a specific VRF is not identified, management is the default. However, if management is configured, it will not be listed in the output of the **show-running** command because it is the default. If a specific VRF is configured, the **show-running** command output will list the VRF for each server. **Note** The current CFS distribution does not support VRF. If CFS distribution is enabled, the logging server configured with the default VRF is distributed as the management VRF. • The facility argument names the syslog facility type. The default outgoing facility is local7. The facilities are listed in the command reference for the Cisco Nexus Series software that you are using. The command references available for Nexus 3000 can be found

	Command or Action	Purpose
		here: http://www.cisco.com/en/US/products/ps11541/prod_command_reference_list.html. **Note** Debugging is a CLI facility but the debug syslogs are not sent to the server.
Step 3	**no logging server** *host* **Example:** `switch(config)# no logging server 172.28.254.254 5`	(Optional) Removes the logging server for the specified host.
Step 4	**show logging server** **Example:** `switch# show logging server`	(Optional) Displays the syslog server configuration.
Step 5	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

The following examples show how to configure a syslog server:

```
switch# configure terminal
switch(config)# logging server 172.28.254.254 5
use-vrf default facility local3

switch# configure terminal
switch(config)# logging server 172.28.254.254 5 use-vrf management facility local3
```

Configuring syslog on a UNIX or Linux System

You can configure a syslog server on a UNIX or Linux system by adding the following line to the /etc/syslog.conf file:

facility.level <five tab characters> *action*

The following table describes the syslog fields that you can configure.

Table 18: syslog Fields in syslog.conf

Field	Description
Facility	Creator of the message, which can be auth, authpriv, cron, daemon, kern, lpr, mail, mark, news, syslog, user, local0 through local7, or an asterisk (*) for all. These facility designators allow you to control the destination of messages based on their origin. **Note** Check your configuration before using a local facility.
Level	Minimum severity level at which messages are logged, which can be debug, info, notice, warning, err, crit, alert, emerg, or an asterisk (*) for all. You can use none to disable a facility.
Action	Destination for messages, which can be a filename, a hostname preceded by the at sign (@), or a comma-separated list of users or an asterisk (*) for all logged-in users.

Procedure

Step 1 Log debug messages with the local7 facility in the file /var/log/myfile.log by adding the following line to the /etc/syslog.conf file:

```
debug.local7            /var/log/myfile.log
```

Step 2 Create the log file by entering these commands at the shell prompt:

```
$ touch /var/log/myfile.log
$ chmod 666 /var/log/myfile.log
```

Step 3 Make sure that the system message logging daemon reads the new changes by checking myfile.log after entering this command:

```
$ kill -HUP ~cat /etc/syslog.pid~
```

Configuring syslog Server Configuration Distribution

You can distribute the syslog server configuration to other switches in the network by using the Cisco Fabric Services (CFS) infrastructure.

After you enable syslog server configuration distribution, you can modify the syslog server configuration and view the pending changes before committing the configuration for distribution. As long as distribution is enabled, the switch maintains pending changes to the syslog server configuration.

Note If the switch is restarted, the syslog server configuration changes that are kept in volatile memory might get lost.

Before You Begin

You must have configured one or more syslog servers.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **logging distribute**	Enables distribution of the syslog server configuration to network switches using the CFS infrastructure. By default, distribution is disabled.
Step 3	switch(config)# **logging commit**	Commits the pending changes to the syslog server configuration for distribution to the switches in the fabric.
Step 4	switch(config)# **logging abort**	Cancels the pending changes to the syslog server configuration.
Step 5	switch(config)# **no logging distribute**	(Optional) Disables the distribution of the syslog server configuration to network switches using the CFS infrastructure. You cannot disable distribution when configuration changes are pending. See the **logging commit** and **logging abort** commands. By default, distribution is disabled.
Step 6	switch# **show logging pending**	(Optional) Displays the pending changes to the syslog server configuration.
Step 7	switch# **show logging pending-diff**	(Optional) Displays the differences from the current syslog server configuration to the pending changes of the syslog server configuration.
Step 8	switch# **show logging internal info**	(Optional) Displays information about the current state of the syslog server distribution and the last action taken.
Step 9	switch# **copy running-config startup-config**	(Optional) Copies the running configuration to the startup configuration.

Displaying and Clearing Log Files

You can display or clear messages in the log file and the NVRAM.

Procedure

	Command or Action	Purpose
Step 1	switch# **show logging last** *number-lines*	Displays the last number of lines in the logging file. You can specify from 1 to 9999 for the last number of lines.
Step 2	switch# **show logging logfile** [**start-time** *yyyy mmm dd hh:mm:ss*] [**end-time** *yyyy mmm dd hh:mm:ss*]	Displays the messages in the log file that have a time stamp within the span entered. If you do not enter an end time, the current time is used. You enter three characters for the month time field and digits for the year and day time fields.
Step 3	switch# **show logging nvram** [**last** *number-lines*]	Displays the messages in the NVRAM. To limit the number of lines displayed, you can enter the last number of lines to display. You can specify from 1 to 100 for the last number of lines.
Step 4	switch# **clear logging logfile**	Clears the contents of the log file.
Step 5	switch# **clear logging nvram**	Clears the logged messages in NVRAM.

The following example shows how to display messages in a log file:

```
switch# show logging last 40
switch# show logging logfile start-time 2007 nov 1 15:10:0
switch# show logging nvram last 10
```

The following example shows how to clear messages in a log file:

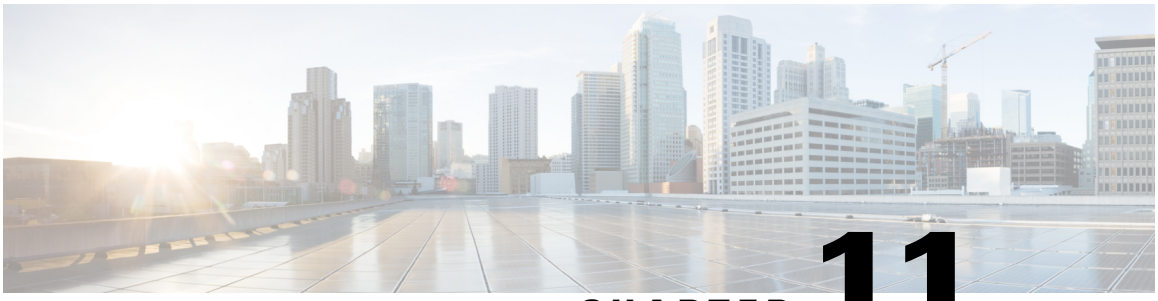
```
switch# clear logging logfile
switch# clear logging nvram
```

Verifying the System Message Logging Configuration

To display system message logging configuration information, perform one of the following tasks:

Command	Purpose
show logging console	Displays the console logging configuration.
show logging info	Displays the logging configuration.
show logging internal info	Displays the syslog distribution information.
show logging ip access-list cache	Displays the IP access list cache.

Command	Purpose
show logging ip access-list cache detail	Displays detailed information about the IP access list cache.
show logging ip access-list status	Displays the status of the IP access list cache.
show logging last *number-lines*	Displays the last number of lines of the log file.
show logging level [*facility*]	Displays the facility logging severity level configuration.
show logging logfile [**start-time** *yyyy mmm dd hh:mm:ss*] [**end-time** *yyyy mmm dd hh:mm:ss*]	Displays the messages in the log file.
show logging module	Displays the module logging configuration.
show logging monitor	Displays the monitor logging configuration.
show logging nvram [**last** *number-lines*]	Displays the messages in the NVRAM log.
show logging pending	Displays the syslog server pending distribution configuration.
show logging pending-diff	Displays the syslog server pending distribution configuration differences.
show logging server	Displays the syslog server configuration.
show logging session	Displays the logging session status.
show logging status	Displays the logging status.
show logging timestamp	Displays the logging time-stamp units configuration.
show running-config acllog	Displays the running configuration for the ACL log file.



CHAPTER 11

Configuring Smart Call Home

This chapter contains the following sections:

Information About Smart Call Home

Smart Call Home provides e-mail-based notification of critical system events. Cisco Nexus Series switches provide a range of message formats for optimal compatibility with pager services, standard e-mail, or XML-based automated parsing applications. You can use this feature to page a network support engineer, e-mail a Network Operations Center, or use Cisco Smart Call Home services to automatically generate a case with the Technical Assistance Center.

If you have a service contract directly with Cisco, you can register your devices for the Smart Call Home service. Smart Call Home provides fast resolution of system problems by analyzing Smart Call Home messages sent from your devices and providing background information and recommendations. For issues that can be identified as known, particularly GOLD diagnostics failures, Automatic Service Requests will be generated by Cisco technical assistance center (TAC).

Smart Call Home offers the following features:

- Continuous device health monitoring and real-time diagnostic alerts.
- Analysis of Smart Call Home messages from your device and, where appropriate, Automatic Service Request generation, routed to the appropriate TAC team, including detailed diagnostic information to speed problem resolution.

- Secure message transport directly from your device or through a downloadable Transport Gateway (TG) aggregation point. You can use a TG aggregation point in cases that require support for multiple devices or in cases where security requirements mandate that your devices may not be connected directly to the Internet.
- Web-based access to Smart Call Home messages and recommendations, inventory and configuration information for all Smart Call Home devices, and field notices, security advisories, and end-of-life information.

Smart Call Home Overview

You can use Smart Call Home to notify an external entity when an important event occurs on your device. Smart Call Home delivers alerts to multiple recipients that you configure in destination profiles.

Smart Call Home includes a fixed set of predefined alerts on your switch. These alerts are grouped into alert groups and CLI commands that are assigned to execute when an alert in an alert group occurs. The switch includes the command output in the transmitted Smart Call Home message.

The Smart Call Home feature offers the following:

- Automatic execution and attachment of relevant CLI command output.
- Multiple message format options such as the following:
  - Short Text—Suitable for pagers or printed reports.
  - Full Text—Fully formatted message information suitable for human reading.
  - XML—Matching readable format that uses the Extensible Markup Language (XML) and the Adaptive Messaging Language (AML) XML schema definition (XSD). The XML format enables communication with the Cisco TAC.
- Multiple concurrent message destinations. You can configure up to 50 e-mail destination addresses for each destination profile.

Smart Call Home Destination Profiles

A Smart Call Home destination profile includes the following information:

- One or more alert groups—The group of alerts that trigger a specific Smart Call Home message if the alert occurs.
- One or more e-mail destinations—The list of receipents for the Smart Call Home messages generated by alert groups assigned to this destination profile.
- Message format—The format for the Smart Call Home message (short text, full text, or XML).
- Message severity level—The Smart Call Home severity level that the alert must meet before the switch generates a Smart Call Home message to all e-mail addresses in the destination profile. The switch does not generate an alert if the Smart Call Home severity level of the alert is lower than the message severity level set for the destination profile.

You can also configure a destination profile to allow periodic inventory update messages by using the inventory alert group that will send out periodic messages daily, weekly, or monthly.

Cisco Nexus switches support the following predefined destination profiles:

- CiscoTAC-1—Supports the Cisco-TAC alert group in XML message format.
- full-text-destination—Supports the full text message format.
- short-text-destination—Supports the short text message format.

Smart Call Home Alert Groups

An alert group is a predefined subset of Smart Call Home alerts that are supported in all Cisco Nexus 3000 Series switches. Alert groups allow you to select the set of Smart Call Home alerts that you want to send to a predefined or custom destination profile. The switch sends Smart Call Home alerts to e-mail destinations in a destination profile only if that Smart Call Home alert belongs to one of the alert groups associated with that destination profile and if the alert has a Smart Call Home message severity at or above the message severity set in the destination profile.

The following table lists the supported alert groups and the default CLI command output included in Smart Call Home messages generated for the alert group.

Table 19: Alert Groups and Executed Commands

Alert Group	Description	Executed Commands
Cisco-TAC	All critical alerts from the other alert groups destined for Smart Call Home.	Execute commands based on the alert group that originates the alert.
Diagnostic	Events generated by diagnostics.	**show diagnostic result module all detail** **show moduleshow version** **show tech-support platform callhome**
Supervisor hardware	Events related to supervisor modules.	**show diagnostic result module all detail** **show moduleshow version** **show tech-support platform callhome**
Linecard hardware	Events related to standard or intelligent switching modules.	**show diagnostic result module all detail** **show moduleshow version** **show tech-support platform callhome**

Alert Group	Description	Executed Commands
Configuration	Periodic events related to configuration.	**show version** **show module** **show running-config all** **show startup-config**
System	Events generated by failure of a software system that is critical to unit operation.	**show system redundancy status** **show tech-support**
Environmental	Events related to power, fan, and environment-sensing elements such as temperature alarms.	**show environment** **show logging last 1000** **show module show version** **show tech-support platform callhome**
Inventory	Inventory status that is provided whenever a unit is cold booted, or when FRUs are inserted or removed. This alert is considered a noncritical event, and the information is used for status and entitlement.	**show module** **show version** **show license usage** **show inventory** **show sprom all** **show system uptime**

Smart Call Home maps the syslog severity level to the corresponding Smart Call Home severity level for syslog port group messages

You can customize predefined alert groups to execute additional CLI **show** commands when specific events occur and send that **show** output with the Smart Call Home message.

You can add **show** commands only to full text and XML destination profiles. Short text destination profiles do not support additional **show** commands because they only allow 128 bytes of text.

Smart Call Home Message Levels

Smart Call Home allows you to filter messages based on their level of urgency. You can associate each destination profile (predefined and user defined) with a Smart Call Home message level threshold. The switch does not generate any Smart Call Home messages with a value lower than this threshold for the destination profile. The Smart Call Home message level ranges from 0 (lowest level of urgency) to 9 (highest level of urgency), and the default is 0 (the switch sends all messages).

Smart Call Home messages that are sent for syslog alert groups have the syslog severity level mapped to the Smart Call Home message level.

Note Smart Call Home does not change the syslog message level in the message text.

The following table shows each Smart Call Home message level keyword and the corresponding syslog level for the syslog port alert group.

Table 20: Severity and Syslog Level Mapping

Smart Call Home Level	Keyword	Syslog Level	Description
9	Catastrophic	N/A	Network-wide catastrophic failure.
8	Disaster	N/A	Significant network impact.
7	Fatal	Emergency (0)	System is unusable.
6	Critical	Alert (1)	Critical conditions that indicate that immediate attention is needed.
5	Major	Critical (2)	Major conditions.
4	Minor	Error (3)	Minor conditions.
3	Warning	Warning (4)	Warning conditions.
2	Notification	Notice (5)	Basic notification and informational messages. Possibly independently insignificant.
1	Normal	Information (6)	Normal event signifying return to normal state.
0	Debugging	Debug (7)	Debugging messages.

Call Home Message Formats

Call Home supports the following message formats:

- Short text message format
- Common fields for all full text and XML messages
- Inserted fields for a reactive or proactive event message
- Inserted fields for an inventory event message

- Inserted fields for a user-generated test message

The following table describes the short text formatting option for all message types.

Table 21: Short Text Message Format

Data Item	Description
Device identification	Configured device name
Date/time stamp	Time stamp of the triggering event
Error isolation message	Plain English description of triggering event
Alarm urgency level	Error level such as that applied to a system message

The following table describes the common event message format for full text or XML.

Table 22: Common Fields for All Full Text and XML Messages

Data Item (Plain Text and XML)	Description (Plain Text and XML)	XML Tag (XML Only)
Time stamp	Date and time stamp of event in ISO time notation: *YYYY-MM-DD HH:MM:SS GMT+HH:MM*	/aml/header/time
Message name	Name of message. Specific event names are listed in the preceding table.	/aml/header/name
Message type	Name of message type, such as reactive or proactive.	/aml/header/type
Message group	Name of alert group, such as syslog.	/aml/header/group
Severity level	Severity level of message.	/aml/header/level
Source ID	Product type for routing.	/aml/header/source

Data Item (Plain Text and XML)	Description (Plain Text and XML)	XML Tag (XML Only)
Device ID	Unique device identifier (UDI) for the end device that generated the message. This field should be empty if the message is nonspecific to a device. The format is *type@Sid@serial*: • *type* is the product model number from backplane IDPROM. • @ is a separator character. • *Sid* is C, identifying the serial ID as a chassis serial number. • *serial* is the number identified by the Sid field. An example is WS-C6509@C@12345678	/aml/ header/deviceID
Customer ID	Optional user-configurable field used for contract information or other ID by any support service.	/aml/ header/customerID
Contract ID	Optional user-configurable field used for contract information or other ID by any support service.	/aml/ header /contractID
Site ID	Optional user-configurable field used for Cisco-supplied site ID or other data meaningful to alternate support service.	/aml/ header/siteID

Data Item (Plain Text and XML)	Description (Plain Text and XML)	XML Tag (XML Only)
Server ID	If the message is generated from the device, this is the unique device identifier (UDI) of the device. The format is *type@Sid@serial*: • *type* is the product model number from backplane IDPROM. • @ is a separator character. • *Sid* is C, identifying the serial ID as a chassis serial number. • *serial* is the number identified by the Sid field. An example is WS-C6509@C@12345678	/aml/header/serverID
Message description	Short text that describes the error.	/aml/body/msgDesc
Device name	Node that experienced the event (hostname of the device).	/aml/body/sysName
Contact name	Name of person to contact for issues associated with the node that experienced the event.	/aml/body/sysContact
Contact e-mail	E-mail address of person identified as the contact for this unit.	/aml/body/sysContactEmail
Contact phone number	Phone number of the person identified as the contact for this unit.	/aml/body/sysContactPhoneNumber
Street address	Optional field that contains the street address for RMA part shipments associated with this unit.	/aml/body/sysStreetAddress
Model name	Model name of the device (the specific model as part of a product family name).	/aml/body/chassis/name
Serial number	Chassis serial number of the unit.	/aml/body/chassis/serialNo
Chassis part number	Top assembly number of the chassis.	/aml/body/chassis/partNo
Fields specific to a particular alert group message are inserted here.		

Data Item (Plain Text and XML)	Description (Plain Text and XML)	XML Tag (XML Only)
The following fields may be repeated if multiple CLI commands are executed for this alert group.		
Command output name	Exact name of the issued CLI command.	/aml/attachments/attachment/name
Attachment type	Specific command output.	/aml/attachments/attachment/type
MIME type	Either plain text or encoding type.	/aml/attachments/attachment/mime
Command output text	Output of command automatically executed.	/aml/attachments/attachment/atdata

The following table describes the reactive event message format for full text or XML.

Table 23: Inserted Fields for a Reactive or Proactive Event Message

Data Item (Plain Text and XML)	Description (Plain Text and XML)	XML Tag (XML Only)
Chassis hardware version	Hardware version of chassis.	/aml/body/chassis/hwVersion
Supervisor module software version	Top-level software version.	/aml/body/chassis/swVersion
Affected FRU name	Name of the affected FRU that is generating the event message.	/aml/body/fru/name
Affected FRU serial number	Serial number of the affected FRU.	/aml/body/fru/serialNo
Affected FRU part number	Part number of the affected FRU.	/aml/body/fru/partNo
FRU slot	Slot number of the FRU that is generating the event message.	/aml/body/fru/slot
FRU hardware version	Hardware version of the affected FRU.	/aml/body/fru/hwVersion
FRU software version	Software version(s) that is running on the affected FRU.	/aml/body/fru/swVersion

The following table describes the inventory event message format for full text or XML.

Table 24: Inserted Fields for an Inventory Event Message

Data Item (Plain Text and XML)	Description (Plain Text and XML)	XML Tag (XML Only)
Chassis hardware version	Hardware version of the chassis.	/aml/body/chassis/hwVersion

Data Item (Plain Text and XML)	Description (Plain Text and XML)	XML Tag (XML Only)
Supervisor module software version	Top-level software version.	/aml/body/chassis/swVersion
FRU name	Name of the affected FRU that is generating the event message.	/aml/body/fru/name
FRU s/n	Serial number of the FRU.	/aml/body/fru/serialNo
FRU part number	Part number of the FRU.	/aml/body/fru/partNo
FRU slot	Slot number of the FRU.	/aml/body/fru/slot
FRU hardware version	Hardware version of the FRU.	/aml/body/fru/hwVersion
FRU software version	Software version(s) that is running on the FRU.	/aml/body/fru/swVersion

The following table describes the user-generated test message format for full text or XML.

Table 25: Inserted Fields for a User-Generated Test Message

Data Item (Plain Text and XML)	Description (Plain Text and XML)	XML Tag (XML Only)
Process ID	Unique process ID.	/aml/body/process/id
Process state	State of process (for example, running or halted).	/aml/body/process/processState
Process exception	Exception or reason code.	/aml/body/process/exception

Guidelines and Limitations for Smart Call Home

- If there is no IP connectivity, or if the interface in the virtual routing and forwarding (VRF) instance to the profile destination is down, the switch cannot send Smart Call Home messages.
- Operates with any SMTP e-mail server.

Prerequisites for Smart Call Home

- E-mail server connectivity.
- Access to contact name (SNMP server contact), phone, and street address information.
- IP connectivity between the switch and the e-mail server.

- An active service contract for the device that you are configuring.

Default Call Home Settings

Table 26: Default Call Home Parameters

Parameters	Default
Destination message size for a message sent in full text format	4000000
Destination message size for a message sent in XML format	4000000
Destination message size for a message sent in short text format	4000
SMTP server port number if no port is specified	25
Alert group association with profile	All for full-text-destination and short-text-destination profiles. The cisco-tac alert group for the CiscoTAC-1 destination profile.
Format type	XML
Call Home message level	0 (zero)

Configuring Smart Call Home

Registering for Smart Call Home

Before You Begin

- SMARTnet contract number for your switch
- Your e-mail address
- Your Cisco.com ID

Procedure

Step 1 In a browser, navigate to the Smart Call Home Web page.
http://www.cisco.com/go/smartcall/

Step 2 Under **Getting Started**, follow the directions to register Smart Call Home.

What to Do Next

Configure contact information.

Configuring Contact Information

You must configure the e-mail, phone, and street address information for Smart Call Home. You can optionally configure the contract ID, customer ID, site ID, and switch priority information.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **snmp-server contact** *sys-contact*	Configures the SNMP sysContact.
Step 3	switch(config)# **callhome**	Enters Smart Call Home configuration mode.
Step 4	switch(config-callhome)# **email-contact** *email-address*	Configures the e-mail address for the primary person responsible for the switch. The *email-address* can be up to 255 alphanumeric characters in e-mail address format. **Note** You can use any valid e-mail address. The address cannot contain spaces.
Step 5	switch(config-callhome)# **phone-contact** *international-phone-number*	Configures the phone number in international phone number format for the primary person responsible for the device. The *international-phone-number* can be up to 17 alphanumeric characters and must be in international phone number format. **Note** The phone number cannot contain spaces. Use the plus (+) prefix before the number.
Step 6	switch(config-callhome)# **streetaddress** *address*	Configures the street address for the primary person responsible for the switch. The *address* can be up to 255 alphanumeric characters. Spaces are accepted.
Step 7	switch(config-callhome)# **contract-id** *contract-number*	(Optional) Configures the contract number for this switch from the service agreement. The *contract-number* can be up to 255 alphanumeric characters.

	Command or Action	Purpose
Step 8	switch(config-callhome)# **customer-id** *customer-number*	(Optional) Configures the customer number for this switch from the service agreement. The *customer-number* can be up to 255 alphanumeric characters.
Step 9	switch(config-callhome)# **site-id** *site-number*	(Optional) Configures the site number for this switch. The *site-number* can be up to 255 alphanumeric characters in free format.
Step 10	switch(config-callhome)# **switch-priority** *number*	(Optional) Configures the switch priority for this switch. The range is from 0 to 7, with 0 being the highest priority and 7 the lowest. The default is 7.
Step 11	switch# **show callhome**	(Optional) Displays a summary of the Smart Call Home configuration.
Step 12	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to configure the contact information for Call Home:

```
switch# configuration terminal
switch(config)# snmp-server contact personname@companyname.com
switch(config)# callhome
switch(config-callhome)# email-contact personname@companyname.com
switch(config-callhome)# phone-contact +1-800-123-4567
switch(config-callhome)# street-address 123 Anystreet St., Anycity, Anywhere
```

What to Do Next

Create a destination profile.

Creating a Destination Profile

You must create a user-defined destination profile and configure the message format for that new destination profile.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.

	Command or Action	Purpose
Step 2	switch(config)# **callhome**	Enters Smart Call Home configuration mode.
Step 3	switch(config-callhome)# **destination-profile** {**ciscoTAC-1** {**alert-group** *group* | **email-addr** *address* | **http** *URL* | **transport-method** {**email** | **http**}} | *profile-name* {**alert-group** *group* | **email-addr** *address* | **format** {**XML** | **full-txt** | **short-txt**} | **http** *URL* | **message-level** *level* | **message-size** *size* | **transport-method** {**email** | **http**}} | **full-txt-destination** {**alert-group** *group* | **email-addr** *address* | **http** *URL* | **message-level** *level* | **message-size** *size* | **transport-method** {**email** | **http**}} | **short-txt-destination** {**alert-group** *group* | **email-addr** *address* | **http** *URL* | **message-level** *level* | **message-size** *size* | **transport-method** {**email** | **http**}}}	Creates a new destination profile and sets the message format for the profile. The profile-name can be any alphanumeric string up to 31 characters. For further details about this command, see the command reference for the Cisco Nexus Series software that you are using. The command references available for Nexus 3000 can be found here: http://www.cisco.com/en/US/products/ps11541/prod_command_reference_list.html.
Step 4	switch# **show callhome destination-profile** [**profile** *name*]	(Optional) Displays information about one or more destination profiles.
Step 5	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to create a destination profile for Smart Call Home:

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# destination-profile Noc101 format full-text
```

Modifying a Destination Profile

You can modify the following attributes for a predefined or user-defined destination profile:

- Destination address—The actual address, pertinent to the transport mechanism, to which the alert should be sent.
- Message formatting—The message format used for sending the alert (full text, short text, or XML).
- Message level—The Call Home message severity level for this destination profile.
- Message size—The allowed length of a Call Home message sent to the e-mail addresses in this destination profile.

Note You cannot modify or delete the CiscoTAC-1 destination profile.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **callhome**	Enters Smart Call Home configuration mode.
Step 3	switch(config-callhome)# **destination-profile** {*name* | **full-txt-destination** | **short-txt-destination**} **email-addr** *address*	Configures an e-mail address for a user-defined or predefined destination profile. You can configure up to 50 e-mail addresses in a destination profile.
Step 4	**destination-profile** {*name* | **full-txt-destination** | **short-txt-destination**} **message-level** *number*	Configures the Call Home message severity level for this destination profile. The switch sends only alerts that have a matching or higher Call Home severity level to destinations in this profile. The range for the *number* is from 0 to 9, where 9 is the highest severity level.
Step 5	switch(config-callhome)# **destination-profile** {*name* | **full-txt-destination** | **short-txt-destination**} **message-size** *number*	Configures the maximum message size for this destination profile. The range is from 0 to 5000000 for full-txt-destination and the default is 2500000. The range is from 0 to 100000 for short-txt-destination and the default is 4000. The value is 5000000 for CiscoTAC-1, which is not changeable.
Step 6	switch# **show callhome destination-profile** [**profile** *name*]	(Optional) Displays information about one or more destination profiles.
Step 7	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to modify a destination profile for Call Home:

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# destination-profile full-text-destination email-addr
person@example.com
switch(config-callhome)# destination-profile full-text-destination message-level 5
switch(config-callhome)# destination-profile full-text-destination message-size 10000
switch(config-callhome)#
```

What to Do Next

Associate an alert group with a destination profile.

Associating an Alert Group with a Destination Profile

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **callhome**	Enters Smart Call Home configuration mode.
Step 3	switch(config-callhome)# **destination-profile** *name* **alert-group** {**All** | **Cisco-TAC** | **Configuration** | **Diagnostic** | **Environmental** | **Inventory** | **License** | **Linecard-Hardware** | **Supervisor-Hardware** | **Syslog-group-port** | **System** | **Test**}	Associates an alert group with this destination profile. Use the **All** keyword to associate all alert groups with the destination profile.
Step 4	switch# **show callhome destination-profile** [**profile** *name*]	(Optional) Displays information about one or more destination profiles.
Step 5	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to associate all alert groups with the destination profile Noc101:

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# destination-profile Noc101 alert-group All
switch(config-callhome)#
```

What to Do Next

Optionally add show commands to an alert group and configure the SMTP e-mail server.

Adding Show Commands to an Alert Group

You can assign a maximum of five user-defined CLI **show** commands to an alert group.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **callhome**	Enters Smart Call Home configuration mode.

	Command or Action	Purpose
Step 3	switch(config-callhome)# **alert-group** {**Configuration** | **Diagnostic** | **Environmental** | **Inventory** | **License** | **Linecard-Hardware** | **Supervisor-Hardware** | **Syslog-group-port** | **System** | **Test**} **user-def-cmd** *show-cmd*	Adds the **show** command output to any Call Home messages sent for this alert group. Only valid **show** commands are accepted. **Note** You cannot add user-defined CLI **show** commands to the CiscoTAC-1 destination profile.
Step 4	switch# **show callhome user-def-cmds**	(Optional) Displays information about all user-defined **show** commands added to alert groups.
Step 5	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to add the **show ip routing** command o the Cisco-TAC alert group:

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# alert-group Configuration user-def-cmd show ip routing
switch(config-callhome)#
```

What to Do Next

Configure Smart Call Home to connect to the SMTP e-mail server.

Configuring E-Mail Server Details

You must configure the SMTP server address for the Call Home functionality to work. You can also configure the from and reply-to e-mail addresses.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **callhome**	Enters Smart Call Home configuration mode.
Step 3	switch(config-callhome)# **transport email smtp-server** *ip-address* [**port** *number*] [**use-vrf** *vrf-name*]	Configures the SMTP server as either the domain name server (DNS) name, IPv4 address, or IPv6 address. The port*number* ranges are from 1 to 65535. The default port number is 25. Optionally, you can configure the VRF to use when communicating with this SMTP server.

	Command or Action	Purpose
Step 4	switch(config-callhome)# **transport email from** *email-address*	(Optional) Configures the e-mail from field for Smart Call Home messages.
Step 5	switch(config-callhome)# **transport email reply-to** *email-address*	(Optional) Configures the e-mail reply-to field for Smart Call Home messages.
Step 6	switch# **show callhome transport-email**	(Optional) Displays information about the e-mail configuration for Smart Call Home.
Step 7	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to configure the e-mail options for Smart Call Home messages:

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# transport email smtp-server 192.0.2.10 use-vrf Red
switch(config-callhome)# transport email from person@example.com
switch(config-callhome)# transport email reply-to person@example.com
switch(config-callhome)#
```

What to Do Next

Configure periodic inventory notifications.

Configuring Periodic Inventory Notifications

You can configure the switch to periodically send a message with an inventory of all software services currently enabled and running on the device with hardware inventory information. The switch generates two Smart Call Home notifications; periodic configuration messages: periodic inventory messages.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **callhome**	Enters Smart Call Home configuration mode.
Step 3	switch(config-callhome)# **periodic-inventory notification** [**interval** *days*] [**timeofday** *time*]	Configures periodic inventory messages. The **interval** *days* range is from 1 to 30 days. The default is 7 days. The **timeofday** *time* is in HH:MM format.

	Command or Action	Purpose
Step 4	switch# **show callhome**	(Optional) Displays information about Smart Call Home.
Step 5	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to configure the periodic inventory messages to generate every 20 days:

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# periodic-inventory notification interval 20
switch(config-callhome)#
```

What to Do Next

Disable duplicate message throttling.

Disabling Duplicate Message Throttling

You can limit the number of duplicate messages received for the same event. By default, the switch limits the number of duplicate messages received for the same event. If the number of duplicate messages sent exceeds 30 messages within a 2-hour time frame, the switch discards further messages for that alert type.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **callhome**	Enters Smart Call Home configuration mode.
Step 3	switch(config-callhome) # **no duplicate-message throttle**	Disables duplicate message throttling for Smart Call Home. Duplicate message throttling is enabled by default.
Step 4	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to disable duplicate message throttling:

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# no duplicate-message throttle
switch(config-callhome)#
```

What to Do Next

Enable Smart Call Home.

Enabling or Disabling Smart Call Home

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **callhome**	Enters Smart Call Home configuration mode.
Step 3	switch(config-callhome) # [**no**] **enable**	Enables or disables Smart Call Home. Smart Call Home is disabled by default.
Step 4	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to enable Smart Call Home:

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# enable
switch(config-callhome)#
```

What to Do Next

Optionally, generate a test message.

Testing the Smart Call Home Configuration

Before You Begin

Verify that the message level for the destination profile is set to 2 or lower.

Important Smart Call Home testing fails when the message level for the destination profile is set to 3 or higher.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.

	Command or Action	Purpose
Step 2	switch(config)# **callhome**	Enters Smart Call Home configuration mode.
Step 3	switch(config-callhome) # **callhome send diagnostic**	Sends the specified Smart Call Home message to all configured destinations.
Step 4	switch(config-callhome) # **callhome test**	Sends a test message to all configured destinations.
Step 5	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to enable Smart Call Home:

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# callhome send diagnostic
switch(config-callhome)# callhome test
switch(config-callhome)#
```

Verifying the Smart Call Home Configuration

Use one of the following commands to verify the configuration:

Command	Purpose
switch# **show callhome**	Displays the status for Call Home.
switch# **show callhome destination-profile** *name*	Displays one or more Call Home destination profiles.
switch# **show callhome pending-diff**	Displays the differences between the pending and running Smart Call Home configuration.
switch# **show callhome status**	Displays the Smart Call Home status.
switch# **show callhome transport-email**	Displays the e-mail configuration for Smart Call Home.
switch# **show callhome user-def-cmds**	Displays CLI commands added to any alert groups.
switch# **show running-config** [**callhome** | **callhome-all**]	Displays the running configuration for Smart Call Home.
switch# **show startup-config callhome**	Displays the startup configuration for Smart Call Home.
switch# **show tech-support callhome**	Displays the technical support output for Smart Call Home.

Sample Syslog Alert Notification in Full-Text Format

This sample shows the full-text format for a syslog port alert-group notification:

```
source:MDS9000
Switch Priority:7
Device Id:WS-C6509@C@FG@07120011
Customer Id:Example.com
Contract Id:123
Site Id:San Jose
Server Id:WS-C6509@C@FG@07120011
Time of Event:2004-10-08T11:10:44
Message Name:SYSLOG_ALERT
Message Type:Syslog
Severity Level:2
System Name:10.76.100.177
Contact Name:User Name
Contact Email:person@example.com
Contact Phone:+1-408-555-1212
Street Address:#1234 Any Street, Any City, Any State, 12345
Event Description:2006 Oct  8 11:10:44 10.76.100.177 %PORT-5-IF_TRUNK_UP:
%$VLAN 1%$ Interface e2/5, vlan 1 is up
syslog_facility:PORT
start chassis information:
Affected Chassis:WS-C6509
Affected Chassis Serial Number:FG@07120011
Affected Chassis Hardware Version:0.104
Affected Chassis Software Version:3.1(1)
Affected Chassis Part No:73-8607-01
end chassis information:
```

Sample Syslog Alert Notification in XML Format

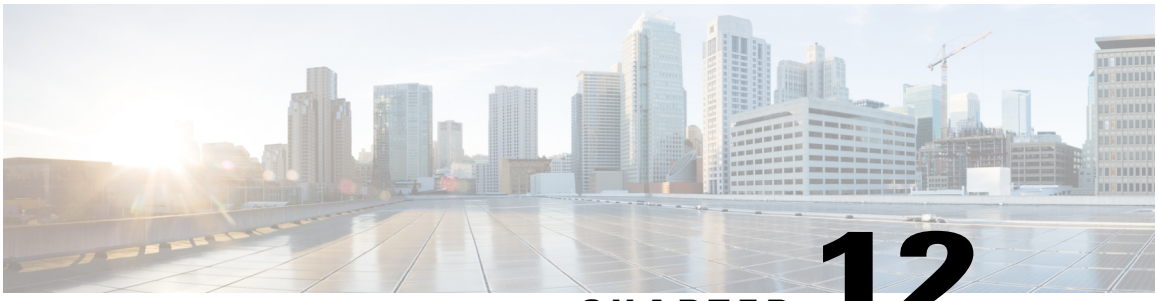
This sample shows the XML format for a syslog port alert-group notification:

```
From: example
Sent: Wednesday, April 25, 2007 7:20 AM
To: User (user)
Subject: System Notification From Router - syslog - 2007-04-25 14:19:55
GMT+00:00
<?xml version="1.0" encoding="UTF-8"?>
<soap-env:Envelope xmlns:soap-env="http://www.w3.org/2003/05/soap-envelope">
<soap-env:Header>
<aml-session:Session xmlns:aml-session="http://www.example.com/2004/01/aml-session"
soap-env:mustUnderstand="true" soap-env:role=
"http://www.w3.org/2003/05/soap-envelope/role/next">
<aml-session:To>http://tools.example.com/services/DDCEService</aml-session:To>
<aml-session:Path>
<aml-session:Via>http://www.example.com/appliance/uri</aml-session:Via>
</aml-session:Path>
<aml-session:From>http://www.example.com/appliance/uri</aml-session:From>
<aml-session:MessageId>M2:69000101:C9D9E20B</aml-session:MessageId>
</aml-session:Session>
</soap-env:Header>
<soap-env:Body>
<aml-block:Block xmlns:aml-block="http://www.example.com/2004/01/aml-block">
<aml-block:Header>
<aml-block:Type>http://www.example.com/2005/05/callhome/syslog</aml-block:Type>
<aml-block:CreationDate>2007-04-25 14:19:55 GMT+00:00</aml-block:CreationDate>
<aml-block:Builder>
<aml-block:Name>Cat6500</aml-block:Name>
<aml-block:Version>2.0</aml-block:Version>
</aml-block:Builder>
<aml-block:BlockGroup>
```

```
<aml-block:GroupId>G3:69000101:C9F9E20C</aml-block:GroupId>
<aml-block:Number>0</aml-block:Number>
<aml-block:IsLast>true</aml-block:IsLast>
<aml-block:IsPrimary>true</aml-block:IsPrimary>
<aml-block:WaitForPrimary>false</aml-block:WaitForPrimary>
</aml-block:BlockGroup>
<aml-block:Severity>2</aml-block:Severity>
</aml-block:Header>
<aml-block:Content>
<ch:Call Home xmlns:ch="http://www.example.com/2005/05/callhome" version="1.0">
<ch:EventTime>2007-04-25 14:19:55 GMT+00:00</ch:EventTime>
<ch:MessageDescription>03:29:29: %CLEAR-5-COUNTERS: Clear counter on all
interfaces by console</ch:MessageDescription>
<ch:Event>
<ch:Type>syslog</ch:Type>
<ch:SubType>
</ch:SubType>
<ch:Brand>Cisco Systems</ch:Brand>
<ch:Series>Catalyst 6500 Series Switches</ch:Series>
</ch:Event>
<ch:CustomerData>
<ch:UserData>
<ch:Email>person@example.com</ch:Email>
</ch:UserData>
<ch:ContractData>
<ch:CustomerId>12345</ch:CustomerId>
<ch:SiteId>building 1</ch:SiteId>
<ch:ContractId>abcdefg12345</ch:ContractId>
<ch:DeviceId>WS-C6509@C@69000101</ch:DeviceId>
</ch:ContractData>
<ch:SystemInfo>
<ch:Name>Router</ch:Name>
<ch:Contact>
</ch:Contact>
<ch:ContactEmail>user@example.com</ch:ContactEmail>
<ch:ContactPhoneNumber>+1-408-555-1212</ch:ContactPhoneNumber>
<ch:StreetAddress>#1234 Any Street, Any City, Any State, 12345
</ch:StreetAddress>
</ch:SystemInfo>
</ch:CustomerData>
<ch:Device>
<rme:Chassis xmlns:rme="http://www.example.com/rme/4.0">
<rme:Model>WS-C6509</rme:Model>
<rme:HardwareVersion>1.0</rme:HardwareVersion>
<rme:SerialNumber>69000101</rme:SerialNumber>
<rme:AdditionalInformation>
<rme:AD name="PartNumber" value="73-3438-03 01" />
<rme:AD name="SoftwareVersion" value="4.0(20080421:012711)" />
</rme:AdditionalInformation>
</rme:Chassis>
</ch:Device>
</ch:Call Home>
</aml-block:Content>
<aml-block:Attachments>
<aml-block:Attachment type="inline">
<aml-block:Name>show logging</aml-block:Name>
<aml-block:Data encoding="plain">
<![CDATA[Syslog logging: enabled (0 messages dropped, 0 messages
rate-limited, 0 flushes, 0 overruns, xml disabled, filtering disabled)
    Console logging: level debugging, 53 messages logged, xml disabled,
filtering disabled    Monitor logging: level debugging, 0 messages logged,
xml disabled,filtering disabled    Buffer logging: level debugging,
53 messages logged, xml disabled,      filtering disabled    Exception
Logging: size (4096 bytes)    Count and timestamp logging messages: disabled
    Trap logging: level informational, 72 message lines logged
Log Buffer (8192 bytes):
00:00:54: curr is 0x20000
00:00:54: RP: Currently running ROMMON from F2 region
00:01:05: %SYS-5-CONFIG_I: Configured from memory by console
00:01:09: %SYS-5-RESTART: System restarted --Cisco IOS Software,
s72033_rp Software (s72033_rp-ADVENTERPRISEK9_DBG-VM), Experimental
Version 12.2(20070421:012711) Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Thu 26-Apr-07 15:54 by xxx
```

```
Firmware compiled 11-Apr-07 03:34 by integ Build [100]00:01:01: %PFREDUN-6-ACTIVE:
 Initializing as ACTIVE processor for this switch00:01:01: %SYS-3-LOGGER_FLUSHED:
System was paused for 00:00:00 to ensure console debugging output.00:03:00: SP: SP:
 Currently running ROMMON from F1 region00:03:07: %C6K_PLATFORM-SP-4-CONFREG_BREAK
_ENABLED: The default factory setting for config register is 0x2102.It is advisable
 to retain 1 in 0x2102 as it prevents returning to ROMMON when break is issued.00:03:18:
 %SYS-SP-5-RESTART: System restarted --Cisco IOS Software, s72033_sp Software
 (s72033_sp-ADVENTERPRISEK9_DBG-VM), Experimental Version 12.2(20070421:012711)Copyright
 (c) 1986-2007 by Cisco Systems, Inc.
Compiled Thu 26-Apr-07 18:00 by xxx
00:03:18: %SYS-SP-6-BOOTTIME: Time taken to reboot after reload =  339 seconds
00:03:18: %OIR-SP-6-INSPS: Power supply inserted in slot 1
00:03:18: %C6KPWR-SP-4-PSOK: power supply 1 turned on.
00:03:18: %OIR-SP-6-INSPS: Power supply inserted in slot00:01:09: %SSH-5-ENABLED:
 SSH 1.99 has been enabled
00:03:18: %C6KPWR-SP-4-PSOK: power supply 2 turned on.
00:03:18: %C6KPWR-SP-4-PSREDUNDANTMISMATCH: power supplies rated outputs do not match.
00:03:18: %C6KPWR-SP-4-PSREDUNDANTBOTHSUPPLY: in power-redundancy mode, system is
 operating on both power supplies.
00:01:10: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is OFF
00:01:10: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is OFF
00:03:20: %C6KENV-SP-4-FANHIOUTPUT: Version 2 high-output fan-tray is in effect
00:03:22: %C6KPWR-SP-4-PSNOREDUNDANCY: Power supplies are not in full redundancy,
 power usage exceeds lower capacity supply
00:03:26: %FABRIC-SP-5-FABRIC_MODULE_ACTIVE: The Switch Fabric Module in slot 6
 became active.
00:03:28: %DIAG-SP-6-RUN_MINIMUM: Module 6: Running Minimal Diagnostics...
00:03:50: %DIAG-SP-6-DIAG_OK: Module 6: Passed Online Diagnostics
00:03:50: %OIR-SP-6-INSCARD: Card inserted in slot 6, interfaces are now online
00:03:51: %DIAG-SP-6-RUN_MINIMUM: Module 3: Running Minimal Diagnostics...
00:03:51: %DIAG-SP-6-RUN_MINIMUM: Module 7: Running Minimal Diagnostics...
00:03:51: %DIAG-SP-6-RUN_MINIMUM: Module 9: Running Minimal Diagnostics...
00:01:51: %MFIB_CONST_RP-6-REPLICATION_MODE_CHANGE: Replication Mode Change Detected.
 Current system replication mode is Ingress
00:04:01: %DIAG-SP-6-DIAG_OK: Module 3: Passed Online Diagnostics
00:04:01: %OIR-SP-6-DOWNGRADE: Fabric capable module 3 not at an appropriate hardware
 revision level, and can only run in flowthrough mode
00:04:02: %OIR-SP-6-INSCARD: Card inserted in slot 3, interfaces are now online
00:04:11: %DIAG-SP-6-DIAG_OK: Module 7: Passed Online Diagnostics
00:04:14: %OIR-SP-6-INSCARD: Card inserted in slot 7, interfaces are now online
00:04:35: %DIAG-SP-6-DIAG_OK: Module 9: Passed Online Diagnostics
00:04:37: %OIR-SP-6-INSCARD: Card inserted in slot 9, interfaces are now online
00:00:09: DaughterBoard (Distributed Forwarding Card 3)
Firmware compiled 11-Apr-07 03:34 by integ Build [100]
00:00:22: %SYS-DFC4-5-RESTART: System restarted --
Cisco DCOS Software, c6lc2 Software (c6lc2-SPDBG-VM), Experimental Version 4.0
(20080421:012711)Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Thu 26-Apr-08 17:20 by xxx
00:00:23: DFC4: Currently running ROMMON from F2 region
00:00:25: %SYS-DFC2-5-RESTART: System restarted --
Cisco IOS Software, c6slc Software (c6slc-SPDBG-VM), Experimental Version 12.2
(20070421:012711)Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Thu 26-Apr-08 16:40 by username1
00:00:26: DFC2: Currently running ROMMON from F2 region
00:04:56: %DIAG-SP-6-RUN_MINIMUM: Module 4: Running Minimal Diagnostics...
00:00:09: DaughterBoard (Distributed Forwarding Card 3)
Firmware compiled 11-Apr-08 03:34 by integ Build [100]
slot_id is 8
00:00:31: %FLASHFS_HES-DFC8-3-BADCARD: /bootflash:: The flash card seems to
 be corrupted
00:00:31: %SYS-DFC8-5-RESTART: System restarted --
Cisco DCOS Software, c6lc2 Software (c6lc2-SPDBG-VM), Experimental Version 4.0
(20080421:012711)Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Thu 26-Apr-08 17:20 by username1
00:00:31: DFC8: Currently running ROMMON from S (Gold) region
00:04:59: %DIAG-SP-6-RUN_MINIMUM: Module 2: Running Minimal Diagnostics...
00:05:12: %DIAG-SP-6-RUN_MINIMUM: Module 8: Running Minimal Diagnostics...
00:05:13: %DIAG-SP-6-RUN_MINIMUM: Module 1: Running Minimal Diagnostics...
00:00:24: %SYS-DFC1-5-RESTART: System restarted --
Cisco DCOS Software, c6slc Software (c6slc-SPDBG-VM), Experimental Version 4.0
(20080421:012711)Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Thu 26-Apr-08 16:40 by username1
00:00:25: DFC1: Currently running ROMMON from F2 region
```

```
00:05:30: %DIAG-SP-6-DIAG_OK: Module 4: Passed Online Diagnostics
00:05:31: %SPAN-SP-6-SPAN_EGRESS_REPLICATION_MODE_CHANGE: Span Egress HW
Replication Mode Change Detected. Current replication mode for unused asic
 session 0 is Centralized
00:05:31: %SPAN-SP-6-SPAN_EGRESS_REPLICATION_MODE_CHANGE: Span Egress HW
Replication Mode Change Detected. Current replication mode for unused asic
 session 1 is Centralized
00:05:31: %OIR-SP-6-INSCARD: Card inserted in slot 4, interfaces are now online
00:06:02: %DIAG-SP-6-DIAG_OK: Module 1: Passed Online Diagnostics
00:06:03: %OIR-SP-6-INSCARD: Card inserted in slot 1, interfaces are now online
00:06:31: %DIAG-SP-6-DIAG_OK: Module 2: Passed Online Diagnostics
00:06:33: %OIR-SP-6-INSCARD: Card inserted in slot 2, interfaces are now online
00:04:30: %XDR-6-XDRIPCNOTIFY: Message not sent to slot 4/0 (4) because of IPC
 error timeout. Disabling linecard. (Expected during linecard OIR)
00:06:59: %DIAG-SP-6-DIAG_OK: Module 8: Passed Online Diagnostics
00:06:59: %OIR-SP-6-DOWNGRADE_EARL: Module 8 DFC installed is not identical to
 system PFC and will perform at current system operating mode.
00:07:06: %OIR-SP-6-INSCARD: Card inserted in slot 8, interfaces are now online
Router#]]>
</aml-block:Data>
</aml-block:Attachment>
</aml-block:Attachments>
</aml-block:Block>
</soap-env:Body>
</soap-env:Envelope>
```



CHAPTER 12

Configuring DNS

This chapter contains the following sections:

DNS Client Overview

If your network devices require connectivity with devices in networks for which you do not control name assignment, you can assign device names that uniquely identify your devices within the entire internetwork using the domain name server (DNS). DNS uses a hierarchical scheme for establishing host names for network nodes, which allows local control of the segments of the network through a client-server scheme. The DNS system can locate a network device by translating the host name of the device into its associated IP address.

On the Internet, a domain is a portion of the naming hierarchy tree that refers to general groupings of networks based on organization type or geography. Domain names are pieced together with periods (.) as the delimiting characters. For example, Cisco is a commercial organization that the Internet identifies by a com domain, so its domain name is cisco.com. A specific host name in this domain, the File Transfer Protocol (FTP) system, for example, is identified as ftp.cisco.com.

Name Servers

Name servers keep track of domain names and know the parts of the domain tree for which they have complete information. A name server may also store information about other parts of the domain tree. To map domain names to IP addresses in Cisco NX-OS, you must first identify the host names, then specify a name server, and enable the DNS service.

Cisco NX-OS allows you to statically map IP addresses to domain names. You can also configure Cisco NX-OS to use one or more domain name servers to find an IP address for a host name.

DNS Operation

A name server handles client-issued queries to the DNS server for locally defined hosts within a particular zone as follows:

- An authoritative name server responds to DNS user queries for a domain name that is under its zone of authority by using the permanent and cached entries in its own host table. If the query is for a domain name that is under its zone of authority but for which it does not have any configuration information, the authoritative name server simply replies that no such information exists.
- A name server that is not configured as the authoritative name server responds to DNS user queries by using information that it has cached from previously received query responses. If no router is configured as the authoritative name server for a zone, queries to the DNS server for locally defined hosts will receive nonauthoritative responses.

Name servers answer DNS queries (forward incoming DNS queries or resolve internally generated DNS queries) according to the forwarding and lookup parameters configured for the specific domain.

High Availability

Cisco NX-OS supports stateless restarts for the DNS client. After a reboot or supervisor switchover, Cisco NX-OS applies the running configuration.

Prerequisites for DNS Clients

The DNS client has the following prerequisites:

- You must have a DNS name server on your network.

Licensing Requirements for DNS Clients

The following table shows the licensing requirements for this feature:

Product	Licence Rquirement
Cicco NX-OS	DNS requires no license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you. For a complete explanation of the Cisco NX-OS licensing scheme, see the *Cisco NX-OS Licensing Guide*.

Default Settings

The following table shows the default settings for DNS client parameters.

Parameter	Default
DNS client	Enabled

Configuring DNS Clients

You can configure the DNS client to use a DNS server on your network.

Before You Begin

- Ensure that you have a domain name server on your network.

Procedure

	Command or Action	Purpose

Step 1	configuration terminal
	
	Example:
	`switch# configuration terminal`
	`switch(config)#`
		Enters the configuration terminal mode.

Step 2	vrf context managment
	
	Example:
	`switch(config)# vrf context management`
	`switch(config)#`
		Specifies a configurable VRF name.

Step 3	ip host *name address1 [address2... address6]*
	
	Example:
	`switch# ip host cisco-rtp 192.0.2.1`
	`switch(config)#`
		Defines up to six static host name-to-address mappings in the host name cache.

Step 4	ip domain name *name* [**use-vrf** *vrf-name*]
	
	Example:
	`switch(config)# ip domain-name myserver.com`
	`switch(config)#`
		(Optional) Defines the default domain name server that Cisco NX-OS uses to complete unqualified host names. You can optionally define a VRF that Cisco NX-OS uses to resolve this domain name server if it cannot be resolved in the VRF that you configured this domain name under.
		
		Cisco NX-OS appends the default domain name to any host name that does not contain a complete domain name before starting a domain-name lookup.

Step 5	ip domain-list *name* [**use-vrf** *vrf-name*]
	
	Example:
	`switch(config)# ip domain-list mycompany.com`
	`switch(config)#`
		(Optional) Defines additional domain name servers that Cisco NX-OS can use to complete unqualified host names. You can optionally define a VRF that Cisco NX-OS uses to resolve this domain name server if it cannot be resolved in the VRF that you configured this domain name under.

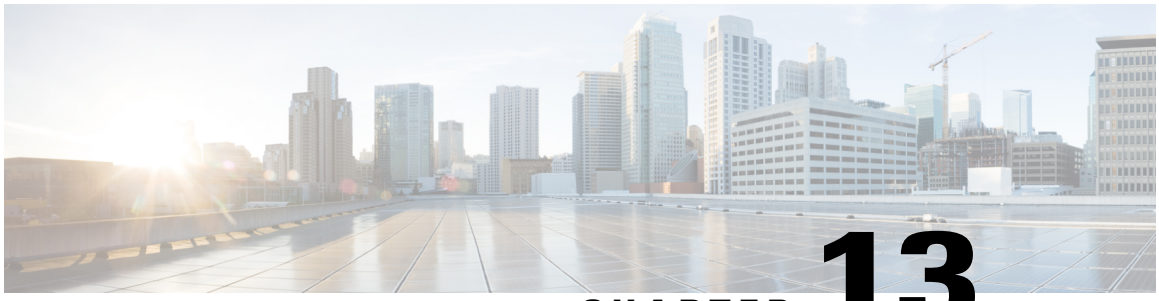
	Command or Action	Purpose
		Cisco NX-OS uses each entry in the domain list to append that domain name to any host name that does not contain a complete domain name before starting a domain-name lookup. Cisco NX-OS continues this for each entry in the domain list until it finds a match.
Step 6	ip name-server *server-address1* [*server-address2... server-address6*] [**use-vrf** *vrf-name*] **Example:** `switch(config)# ip name-server 192.0.2.22`	(Optional) Defines up to six name servers. The address can be either an IPv4 address or an IPv6 address. You can optionally define a VRF that Cisco NX-OS uses to reach this name server if it cannot be reached in the VRF that you configured this name server under.
Step 7	ip domain-lookup **Example:** `switch(config)# ip domain-lookup`	(Optional) Enables DNS-based address translation. Enabled by default.
Step 8	show hosts **Example:** `switch(config)# show hosts`	(Optional) Displays information about DNS.

Step 9	exit
	
	Example:
	`switch(config)# exit`
	`switch#`
		Exits configuration mode and returns to EXEC mode.

Step 10	copy running-config startup-config
	
	Example:
	`switch# copy running-config startup-config`
	`switch#`
		(Optional) Copies the running configuration to the startup configuration.

This example shows how to configure a default domain name and enable DNS lookup:

```
switch# config t
switch(config)# vrf context management
switch(config)# ip domain-name mycompany.com
switch(config)# ip name-server 172.68.0.10
switch(config)# ip domain-lookup
```



CHAPTER 13

Configuring SNMP

This chapter contains the following sections:

Information About SNMP

The Simple Network Management Protocol (SNMP) is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.

SNMP Functional Overview

The SNMP framework consists of three parts:

- An SNMP manager—The system used to control and monitor the activities of network devices using SNMP.
- An SNMP agent—The software component within the managed device that maintains the data for the device and reports these data, as needed, to managing systems. The Cisco Nexus 3000 Series switch supports the agent and MIB. To enable the SNMP agent, you must define the relationship between the manager and the agent.
- A managed information base (MIB)—The collection of managed objects on the SNMP agent

Note Cisco NX-OS does not support SNMP sets for Ethernet MIBs.

The Cisco Nexus 3000 Series switch supports SNMPv1, SNMPv2c and SNMPv3. Both SNMPv1 and SNMPv2c use a community-based form of security.

SNMP is defined in RFC 3410 (http://tools.ietf.org/html/rfc3410), RFC 3411 (http://tools.ietf.org/html/rfc3411), RFC 3412 (http://tools.ietf.org/html/rfc3412), RFC 3413 (http://tools.ietf.org/html/rfc3413), RFC 3414 (http://tools.ietf.org/html/rfc3414), RFC 3415 (http://tools.ietf.org/html/rfc3415), RFC 3416 (http://tools.ietf.org/html/rfc3416), RFC 3417 (http://tools.ietf.org/html/rfc3417), RFC 3418 (http://tools.ietf.org/html/rfc3418), and RFC 3584 (http://tools.ietf.org/html/rfc3584).

SNMP Notifications

A key feature of SNMP is the ability to generate notifications from an SNMP agent. These notifications do not require that requests be sent from the SNMP manager. Notifications can indicate improper user authentication, restarts, the closing of a connection, loss of connection to a neighbor router, or other significant events.

Cisco NX-OS generates SNMP notifications as either traps or informs. A trap is an asynchronous, unacknowledged message sent from the agent to the SNMP managers listed in the host receiver table. Informs are asynchronous messages sent from the SNMP agent to the SNMP manager which the manager must acknowledge receipt of.

Traps are less reliable than informs because the SNMP manager does not send any acknowledgment when it receives a trap. The switch cannot determine if the trap was received. An SNMP manager that receives an inform request acknowledges the message with an SNMP response protocol data unit (PDU). If the Cisco Nexus 3000 Series switch never receives a response, it can send the inform request again.

You can configure Cisco NX-OS to send notifications to multiple host receivers.

SNMPv3

SNMPv3 provides secure access to devices by a combination of authenticating and encrypting frames over the network. The security features provided in SNMPv3 are the following:

- Message integrity—Ensures that a packet has not been tampered with in-transit.
- Authentication—Determines the message is from a valid source.
- Encryption—Scrambles the packet contents to prevent it from being seen by unauthorized sources.

SNMPv3 provides for both security models and security levels. A security model is an authentication strategy that is set up for a user and the role in which the user resides. A security level is the permitted level of security within a security model. A combination of a security model and a security level determines which security mechanism is employed when handling an SNMP packet.

Security Models and Levels for SNMPv1, v2, v3

The security level determines if an SNMP message needs to be protected from disclosure and if the message needs to be authenticated. The various security levels that exist within a security model are as follows:

- noAuthNoPriv—Security level that does not provide authentication or encryption.
- authNoPriv—Security level that provides authentication but does not provide encryption.
- authPriv—Security level that provides both authentication and encryption.

Three security models are available: SNMPv1, SNMPv2c, and SNMPv3. The security model combined with the security level determine the security mechanism applied when the SNMP message is processed.

Table 27: SNMP Security Models and Levels

Model	Level	Authentication	Encryption	What Happens
v1	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
v2c	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
v3	noAuthNoPriv	Username	No	Uses a username match for authentication.
v3	authNoPriv	HMAC-MD5 or HMAC-SHA	No	Provides authentication based on the Hash-Based Message Authentication Code (HMAC) Message Digest 5 (MD5) algorithm or the HMAC Secure Hash Algorithm (SHA).
v3	authPriv	HMAC-MD5 or HMAC-SHA	DES	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides Data Encryption Standard (DES) 56-bit encryption in addition to authentication based on the Cipher Block Chaning (CBC) DES (DES-56) standard.

User-Based Security Model

SNMPv3 User-Based Security Model (USM) refers to SNMP message-level security and offers the following services:

- Message integrity—Ensures that messages have not been altered or destroyed in an unauthorized manner and that data sequences have not been altered to an extent greater than can occur non-maliciously.
- Message origin authentication—Ensures that the claimed identity of the user on whose behalf received data was originated is confirmed.
- Message confidentiality—Ensures that information is not made available or disclosed to unauthorized individuals, entities, or processes.

SNMPv3 authorizes management operations only by configured users and encrypts SNMP messages.

Cisco NX-OS uses two authentication protocols for SNMPv3:

- HMAC-MD5-96 authentication protocol
- HMAC-SHA-96 authentication protocol

Cisco NX-OS uses Advanced Encryption Standard (AES) as one of the privacy protocols for SNMPv3 message encryption and conforms with RFC 3826.

The **priv** option offers a choice of DES or 128-bit AES encryption for SNMP security encryption. The **priv** option along with the **aes-128** token indicates that this privacy password is for generating a 128-bit AES key.The AES priv password can have a minimum of eight characters. If the passphrases are specified in clear text, you can specify a maximum of 64 characters. If you use the localized key, you can specify a maximum of 130 characters.

Note For an SNMPv3 operation using the external AAA server, you must use AES for the privacy protocol in user configuration on the external AAA server.

CLI and SNMP User Synchronization

SNMPv3 user management can be centralized at the Access Authentication and Accounting (AAA) server level. This centralized user management allows the SNMP agent in Cisco NX-OS to leverage the user authentication service of the AAA server. Once user authentication is verified, the SNMP PDUs are processed further. Additionally, the AAA server is also used to store user group names. SNMP uses the group names to apply the access/role policy that is locally available in the switch.

Any configuration changes made to the user group, role, or password results in database synchronization for both SNMP and AAA.

Cisco NX-OS synchronizes user configuration in the following ways:

- The **auth** passphrase specified in the **snmp-server user** command becomes the password for the CLI user.
- The password specified in the **username** command becomes as the **auth** and **priv** passphrases for the SNMP user.

- If you create or delete a user using either SNMP or the CLI, the user is created or deleted for both SNMP and the CLI.
- User-role mapping changes are synchronized in SNMP and the CLI.
- Role changes (deletions or modifications from the CLI are synchronized to SNMP.

Note When you configure passphrase/password in localized key/encrypted format, Cisco NX-OS does not synchronize the user information (passwords, rules, etc.).

Group-Based SNMP Access

Note Because group is a standard SNMP term used industry-wide, roles are referred to as groups in this SNMP section.

SNMP access rights are organized by groups. Each group in SNMP is similar to a role through the CLI. Each group is defined with three accesses: read access, write access, and notification access. Each access can be enabled or disabled within each group.

You can begin communicating with the agent once your user name is created, your roles are set up by your administrator, and you are added to the roles.

Licensing Requirements for SNMP

This feature does not require a license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you. For a complete explanation of the Cisco NX-OS licensing scheme, see the *Cisco NX-OS Licensing Guide*.

Guidelines and Limitations for SNMP

Cisco NX-OS supports read-only access to Ethernet MIBs.

For more information about supported MIBs, see the following URL:

http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

Default SNMP Settings

Table 28: Default SNMP Parameters

Parameters	Default
license notifications	enabled

Parameters	Default
linkUp/Down notification type	ietf-extended

Configuring SNMP

Configuring SNMP Users

Note The commands used to configure SNMP users in Cisco NX-OS are different from those used to configure users in Cisco IOS.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	switch(config)# **snmp-server user** *name* [**auth** {**md5** | **sha**} *passphrase* [**auto**] [**priv** [**aes-128**] *passphrase*] [**engineID** *id*] [**localizedkey**]] **Example:** `switch(config)# snmp-server user Admin auth sha abcd1234 priv abcdefgh`	Configures an SNMP user with authentication and privacy parameters. The passphrase can be any case-sensitive alphanumeric string up to 64 characters. If you use the **localizedkey** keyword, the passphrase can be any case-sensitive alphanumeric string up to 130 characters. The engineID format is a 12-digit, colon-separated decimal number.
Step 3	switch# **show snmp user** **Example:** `switch(config) # show snmp user`	(Optional) Displays information about one or more SNMP users.
Step 4	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

The following example configures an SNMP user:

```
switch# config t
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# snmp-server user Admin auth sha abcd1234 priv abcdefgh
```

Enforcing SNMP Message Encryption

You can configure SNMP to require authentication or encryption for incoming requests. By default the SNMP agent accepts SNMPv3 messages without authentication and encryption. When you enforce privacy, Cisco NX-OS responds with an authorization error for any SNMPv3 PDU request using security level parameter of either **noAuthNoPriv** or **authNoPriv**.

Use the following command in global configuration mode to enforce SNMP message encryption for a specific user.

Command	Purpose
switch(config)# **snmp-server user** *name* **enforcePriv**	Enforces SNMP message encryption for this user.

Use the following command in global configuration mode to enforce SNMP message encryption for all users.

Command	Purpose
switch(config)# **snmp-server globalEnforcePriv**	Enforces SNMP message encryption for all users.

Assigning SNMPv3 Users to Multiple Roles

After you configure an SNMP user, you can assign multiple roles for the user.

Note Only users belonging to a network-admin role can assign roles to other users.

Command	Purpose
switch(config)# **snmp-server user** *name group*	Associates this SNMP user with the configured user role.

Creating SNMP Communities

You can create SNMP communities for SNMPv1 or SNMPv2c.

To create an SNMP community string in a global configuration mode, perform this task:

Command	Purpose
switch(config)# **snmp-server community** *name group* {**ro** | **rw**}	Creates an SNMP community string.

Filtering SNMP Requests

You can assign an access list (ACL) to a community to filter incoming SNMP requests. If the assigned ACL allows the incoming request packet, SNMP processes the request. If the ACL denies the request, SNMP drops the request and sends a system message.

Create the ACL with the following parameters:

- Source IP address
- Destination IP address
- Source port
- Destination port
- Protocol (UDP or TCP)

The ACL applies to both IPv4 and IPv6 over UDP and TCP. After creating the ACL, assign the ACL to the SNMP community.

Tip For more information on creating ACLs, see the *NX-OS Security Configuration Guide* for the Cisco Nexus Series software that you are using. The security configuration guides available for Nexus 3000 can be found here: http://www.cisco.com/en/US/products/ps11541/products_installation_and_configuration_guides_list.html.

Use the following command in global configuration mode to assign an ACL to a community to filter SNMP requests:

Command	Purpose
switch(config)# **snmp-server community** *community name* **use-acl** *acl-name* **Example:** `switch(config)# snmp-server community public use-acl my_acl_for_public`	Assigns an ACL to an SNMP community to filter SNMP requests.

Before You Begin

Create an ACL to assign to the SNMP community.

Assign the ACL to the SNMP community.

Configuring SNMP Notification Receivers

You can configure Cisco NX-OS to generate SNMP notifications to multiple host receivers.

You can configure a host receiver for SNMPv1 traps in a global configuration mode.

Command	Purpose
switch(config)# **snmp-server host** *ip-address* **traps version 1** *community* [**udp_port** *number*]	Configures a host receiver for SNMPv1 traps. The *ip-address* can be an IPv4 or IPv6 address. The community can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535.

You can configure a host receiver for SNMPv2c traps or informs in a global configuration mode.

Command	Purpose
switch(config)# **snmp-server host** *ip-address* {**traps** | **informs**} **version 2c** *community* [**udp_port** *number*]	Configures a host receiver for SNMPv2c traps or informs. The *ip-address* can be an IPv4 or IPv6 address. The community can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535.

You can configure a host receiver for SNMPv3 traps or informs in a global configuration mode.

Command	Purpose
switch(config)# **snmp-server host** *ip-address* {**traps** | **informs**} **version 3** {**auth** | **noauth** | **priv**} *username* [**udp_port** *number*]	Configures a host receiver for SNMPv2c traps or informs. The *ip-address* can be an IPv4 or IPv6 address. The username can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535.

Note The SNMP manager must know the user credentials (authKey/PrivKey) based on the SNMP engineID of the Cisco Nexus 3000 Series switch to authenticate and decrypt the SNMPv3 messages.

The following example shows how to configure a host receiver for an SNMPv1 trap:

```
switch(config)# snmp-server host 192.0.2.1 traps version 1 public
```

The following example shows how to configure a host receiver for an SNMPv2 inform:

```
switch(config)# snmp-server host 192.0.2.1 informs version 2c public
```

The following example shows how to configure a host receiver for an SNMPv3 inform:

```
switch(config)# snmp-server host 192.0.2.1 informs version 3 auth NMS
```

Configuring SNMP Notification Receivers with VRFs

You can configure Cisco NX-OS to use a configured VRF to reach the host receiver. SNMP adds entries into the cExtSnmpTargetVrfTable of the CISCO-SNMP-TARGET-EXT-MIB when you configure the VRF reachability and filtering options for an SNMP notification receiver.

Note You must configure the host before configuring the VRF reachability or filtering options.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch# **snmp-server host** *ip-address* **use-vrf** *vrf_name* [**udp_port** *number*]	Configures SNMP to use the selected VRF to communicate with the host receiver. The ip-address can be an IPv4 or IPv6 address. The VRF name can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535. This command adds an entry into thc ExtSnmpTargetVrfTable of the CISCO-SNMP-TARGET-EXT-MB.
Step 3	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

The following example shows how to configure the SNMP server host with IP address 192.0.2.1 to use the VRF named "Blue:"

```
switch# configuration terminal
switch(config)# snmp-server host 192.0.2.1 use-vrf Blue
switch(config)# copy running-config startup-config
```

Filtering SNMP Notifications Based on a VRF

You can configure Cisco NX-OS filter notifications based on the VRF in which the notification occurred.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **snmp-server host** *ip-address* **filter-vrf** *vrf_name* [**udp_port** *number*]	Filters notifications to the notification host receiver based on the configured VRF. The ip-address can be an IPv4 or IPv6 address. The VRF name can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535.

	Command or Action	Purpose
		This command adds an entry into thc ExtSnmpTargetVrfTable of the CISCO-SNMP-TARGET-EXT-MB.
Step 3	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

The following example shows how to configure filtering of SNMP notifications based on a VRF:

```
switch# configuration terminal
switch(config)# snmp-server host 192.0.2.1 filter-vrf Red
switch(config)# copy running-config startup-config
```

Configuring SNMP for Inband Access

You can configure SNMP for inband access using the following:

- Using SNMP v2 without context—You can use a community which is mapped to a context. In this case the SNMP client does not need to know about the context.
- Using SNMP v2 with context—The SNMP client needs to specify the context by specifying a community, for example, <community>@<context>.
- Using SNMP v3—You can specify the context.

Procedure

	Command or Action	Purpose
Step 1	switch# **configuration terminal**	Enters global configuration mode.
Step 2	switch(config)# **snmp-server context** *context-name* **vrf** *vrf-name*	Maps an SNMP context to the management VRF or default VRF. Custom VRFs are not supported. The names can be any alphanumeric string up to 32 characters.
Step 3	switch(config)# **snmp-server community** *community-name* **group** *group-name*	Maps an SNMPv2c community to an SNMP context and identifies the group that the community belongs. The names can be any alphanumeric string up to 32 characters.
Step 4	switch(config)# **snmp-server mib community-map** *community-name* **context** *context-name*	Maps an SNMPv2c community to an SNMP context. The names can be any alphanumeric string up to 32 characters.

The following SNMPv2 example shows how to map a community named snmpdefault to a context:

```
switch# config t
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# snmp-server context def vrf default
switch(config)# snmp-server community snmpdefault group network-admin
switch(config)# snmp-server mib community-map snmpdefault context def
switch(config)#
```

The following SNMPv2 example shows how to configure and inband access to the community comm which is not mapped:

```
switch# config t
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# snmp-server context def vrf default
switch(config)# snmp-server community comm group network-admin
switch(config)#
```

The following SNMPv3 example shows how to use a v3 username and password:

```
switch# config t
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# snmp-server context def vrf default
switch(config)#
```

Enabling SNMP Notifications

You can enable or disable notifications. If you do not specify a notification name, Cisco NX-OS enables all notifications.

Note The **snmp-server enable traps** CLI command enables both traps and informs, depending on the configured notification host receivers.

The following table lists the CLI commands that enable the notifications for Cisco NX-OS MIBs.

Table 29: Enabling SNMP Notifications

MIB	Related Commands
All notifications	**snmp-server enable traps**
BRIDGE-MIB	**snmp-server enable traps bridge newroot** **snmp-server enable traps bridge topologychange**
CISCO-AAA-SERVER-MIB	**snmp-server enable traps aaa**
ENITY-MIB, CISCO-ENTITY-FRU-CONTROL-MIB, CISCO-ENTITY-SENSOR-MIB	**snmp-server enable traps entity** **snmp-server enable traps entity fru**
CISCO-LICENSE-MGR-MIB	**snmp-server enable traps license**
IF-MIB	**snmp-server enable traps link**
CISCO-PSM-MIB	**snmp-server enable traps port-security**

MIB	Related Commands
SNMPv2-MIB	**snmp-server enable traps snmp** **snmp-server enable traps snmp authentication**
CISCO-FCC-MIB	**snmp-server enable traps fcc**
CISCO-DM-MIB	**snmp-server enable traps fcdomain**
CISCO-NS-MIB	**snmp-server enable traps fcns**
CISCO-FCS-MIB	**snmp-server enable traps fcs discovery-complete** **snmp-server enable traps fcs request-reject**
CISCO-FDMI-MIB	**snmp-server enable traps fdmi**
CISCO-FSPF-MIB	**snmp-server enable traps fspf**
CISCO-PSM-MIB	**snmp-server enable traps port-security**
CISCO-RSCN-MIB	**snmp-server enable traps rscn** **snmp-server enable traps rscn els** **snmp-server enable traps rscn ils**
CISCO-ZS-MIB	**snmp-server enable traps zone** **snmp-server enable traps zone default-zone-behavior-change** **snmp-server enable traps zone enhanced-zone-db-change** **snmp-server enable traps zone merge-failure** **snmp-server enable traps zone merge-success** **snmp-server enable traps zone request-reject** **snmp-server enable traps zone unsupp-mem**

Note The license notifications are enabled by default.

To enable the specified notification in the global configuration mode, perform one of the following tasks:

Command	Purpose
switch(config)# **snmp-server enable traps**	Enables all SNMP notifications.
switch(config)# **snmp-server enable traps aaa** [**server-state-change**]	Enables the AAA SNMP notifications.

Command	Purpose
switch(config)# **snmp-server enable traps entity** [**fru**]	Enables the ENTITY-MIB SNMP notifications.
switch(config)# **snmp-server enable traps license**	Enables the license SNMP notification.
switch(config)# **snmp-server enable traps port-security**	Enables the port security SNMP notifications.
switch(config)# **snmp-server enable traps snmp** [**authentication**]	Enables the SNMP agent notifications.

Configuring Link Notifications

You can configure which linkUp/linkDown notifications to enable on a device. You can enable the following types of linkUp/linkDown notifications:

- Cisco—Cisco NX-OS sends only the Cisco-defined notifications (cieLinkUp, cieLinkDow in CISCO-IF-EXTENSION-MIB.my), if ifLinkUpDownTrapEnable (defined in IF-MIB) is enabled for that interface.
- IETF—Cisco NX-OS sends only the IETF-defined notifications (linkUp, linkDown in IF-MIB) with only the defined varbinds, if ifLinkUpDownTrapEnable (defined in IF-MIB) is enabled for that interface.
- IEFT extended—Cisco NX-OS sends only the IETF-defined notifications (linkUp, linkDown defined in IF-MIB), if ifLinkUpDownTrapEnable (defined in IF-MIB) is enabled for that interface. Cisco NX-OS adds additional varbinds specific to Cisco Systems in addition to the varbinds defined in the IF-MIB. This is the default setting.
- IEFT Cisco—Cisco NX-OS sends the notifications (linkUp, linkDown) defined in IF-MIB and notifications (cieLinkUp, cieLinkDown) defined in CISCO-IF-EXTENSION-MIB.my , if ifLinkUpDownTrapEnable (defined in IF-MIB) is enabled for that interface. Cisco NX-OS sends only the varbinds defined in the linkUp and linkDown notifications.
- IEFT extended Cisco—Cisco NX-OS sends the notifications (linkUp, linkDown) defined in IF-MIB and notifications (cieLinkUp, cieLinkDown) defined in CISCO-IF-EXTENSION-MIB.my, if ifLinkUpDownTrapEnable (defined in IF-MIB) is enabled for that interface. Cisco NX-OS adds additional varbinds specific to Cisco Systems in addition to the varbinds defined in the IF-MIB for the linkUp and linkDown notifications.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

	Command or Action	Purpose
Step 2	**snmp-server enable traps link** [**cisco**] [**ietf** | **ietf-extended**] **Example:** `switch(config)# snmp-server enable traps link cisco`	Enables the link SNMP notifications.

Disabling Link Notifications on an Interface

You can disable linkUp and linkDown notifications on an individual interface. You can use this limit notifications on flapping interface (an interface that transitions between up and down repeatedly).

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **interface** *type slot*/*port*	Specifies the interface to be changed.
Step 3	switch(config -if)# **no snmp trap link-status**	Disables SNMP link-state traps for the interface. Enabled by default.

Enabling One-Time Authentication for SNMP over TCP

You can enable a one-time authentication for SNMP over a TCP session.

Command	Purpose
switch(config)# **snmp-server tcp-session** [**auth**]	Enables a one-time authentication for SNMP over a TCP session. Default is disabled.

Assigning SNMP Switch Contact and Location Information

You can assign the switch contact information, which is limited to 32 characters (without spaces), and the switch location.

Procedure

	Command or Action	Purpose
Step 1	switch# **configuration terminal**	Enters configuration mode.
Step 2	switch(config)# **snmp-server contact** *name*	Configures sysContact, the SNMP contact name.
Step 3	switch(config)# **snmp-server location** *name*	Configures sysLocation, the SNMP location.
Step 4	switch# **show snmp**	(Optional) Displays information about one or more destination profiles.
Step 5	switch# **copy running-config startup-config**	(Optional) Saves this configuration change.

Configuring the Context to Network Entity Mapping

You can configure an SNMP context to map to a logical network entity, such as a protocol instance or VRF.

Procedure

	Command or Action	Purpose
Step 1	switch# **configuration terminal**	Enters configuration mode.
Step 2	switch(config)# **snmp-server context** *context-name* [**instance** *instance-name*] [**vrf** *vrf-name*] [**topology** *topology-name*]	Maps an SNMP context to a protocol instance, VRF, or topology. The names can be any alphanumeric string up to 32 characters.
Step 3	switch(config)# **snmp-server mib community-map** *community-name* **context** *context-name*	Maps an SNMPv2c community to an SNMP context. The names can be any alphanumeric string up to 32 characters.
Step 4	switch(config)# **no snmp-server context** *context-name* [**instance** *instance-name*] [**vrf** *vrf-name*] [**topology** *topology-name*]	(Optional) Deletes the mapping between an SNMP context and a protocol instance, VRF, or topology. The names can be any alphanumeric string up to 32 characters. **Note** Do not enter an instance, VRF, or topology to delete a context mapping. If you use the **instance**, **vrf**, or **topology** keywords, you configure a mapping between the context and a zero-length string.

Disabling SNMP

Procedure

	Command or Action	Purpose

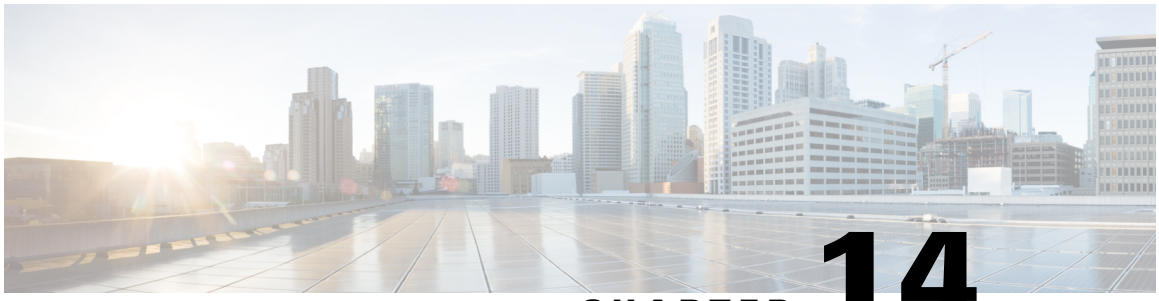
Step 1	**configure terminal**
	
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	switch(config) # **no snmp-server protocol enable** **Example:** `no snmp-server protocol enable`	Disables SNMP. SNMP is disabled by default.

Verifying SNMP Configuration

To display SNMP configuration information, perform one of the following tasks:

Command	Purpose
switch# **show snmp**	Displays the SNMP status.
switch# **show snmp community**	Displays the SNMP community strings.
switch# **show snmp engineID**	Displays the SNMP engineID.
switch# **show snmp group**	Displays SNMP roles.
switch# **show snmp sessions**	Displays SNMP sessions.
switch# **show snmp trap**	Displays the SNMP notifications enabled or disabled.
switch# **show snmp user**	Displays SNMPv3 users.



CHAPTER 14

Configuring RMON

This chapter contains the following sections:

Information About RMON

RMON is an Internet Engineering Task Force (IETF) standard monitoring specification that allows various network agents and console systems to exchange network monitoring data. The Cisco NX-OS supports RMON alarms, events and logs to monitor Cisco Nexus 3000 Series switches

An RMON alarm monitors a specific management information base (MIB) object for a specified interval, triggers an alarm at a specified threshold value (threshold), and resets the alarm at another threshold value. You can use alarms with RMON events to generate a log entry or an SNMP notification when the RMON alarm triggers.

RMON is disabled by default and no events or alarms are configured in Cisco Nexus 3000 Series. You can configure your RMON alarms and events by using the CLI or an SNMP-compatible network management station

RMON Alarms

You can set an alarm on any MIB object that resolves into an SNMP INTEGER type. The specified object must be an existing SNMP MIB object in standard dot notation (for example, 1.3.6.1.2.1.2.2.1.17 represents ifOutOctets.17).

When you create an alarm, you specify the following parameters:

- MIB object to monitor

- Sampling interval—The interval that the Cisco Nexus 3000 Series switch uses to collect a sample value of the MIB object.
- The sample type—Absolute samples take the current snapshot of the MIB object value. Delta samples take two consecutive samples and calculate the difference between them.
- Rising threshold—The value at which the Cisco Nexus 3000 Series switch triggers a rising alarm or resets a falling alarm.
- Falling threshold—The value at which the Cisco Nexus 3000 Series switch triggers a falling alarm or resets a rising alarm.
- Events—The action that the Cisco Nexus 3000 Series switch takes when an alarm (rising or falling) triggers.

Note Use the hcalarms option to set an alarm on a 64-bit integer MIB object.

For example, you can set a delta type rising alarm on an error counter MIB object. If the error counter delta exceeds this value, you can trigger an event that sends an SNMP notification and logs the rising alarm event. This rising alarm will not occur again until the delta sample for the error counter drops below the falling threshold.

Note The falling threshold must be less than the rising threshold.

RMON Events

You can associate a particular event to each RMON alarm. RMON supports the following event types:

- SNMP notification—Sends an SNMP risingAlarm or fallingAlarm notification when the associated alarm triggers.
- Log—Adds an entry in the RMON log table when the associated alarm triggers.
- Both—Sends an SNMP notification and adds an entry in the RMON log table when the associated alarm triggers.

You can specify a different even for a falling alarm and a rising alarm.

Configuration Guidelines and Limitations for RMON

RMON has the following configuration guidelines and limitations:

- You must configure an SNMP user an notification receiver to use the SNMP notification event type.
- You can only configure an RMON alarm on a MIB object that resolves to an integer.

Configuring RMON

Configuring RMON Alarms

You can configure RMON alarms on any integer-based SNMP MIB object.

You can optionally specify the following parameters:

- The event-number to trigger if the rising or falling threshold exceeds the specified limit.
- The owner of the alarm.

Ensure you have configured an SNMP user and enabled SNMP notifications.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **rmon alarm** *index mib-object sample-interval* {**absolute** | **delta**} **rising-threshold** *value* [*event-index*] **falling-threshold** *value* [*event-index*] [**owner** *name*]	Creates an RMON alarm. The value range is from -2147483647 to 2147483647. The owner name can be any alphanumeric string.
Step 3	switch(config)# **rmon hcalarm** *index mib-object sample-interval* {**absolute** | **delta**} **rising-threshold-high** *value* **rising-threshold-low** *value* [*event-index*] **falling-threshold-high** *value* **falling-threshold-low** *value* [*event-index*] [**owner** *name*] [**storagetype** *type*]	Creates an RMON high-capacity alarm. The value range is from -2147483647 to 2147483647. The owner name can be any alphanumeric string. The storage type range is from 1 to 5.
Step 4	switch# **show rmon** {**alarms** | **hcalarms**}	(Optional) Displays information about RMON alarms or high-capacity alarms.
Step 5	switch# **copy running-config startup-config**	(Optional) Saves this configuration change.

The following example shows how to configure RMON alarms:

```
switch# configure terminal
switch(config)# rmon alarm 1 1.3.6.1.2.1.2.2.1.17.83886080 5 delta rising-threshold 5 1
falling-threshold 0 owner test
switch(config)# exit
switch# show rmon alarms
Alarm 1 is active, owned by test
 Monitors 1.3.6.1.2.1.2.2.1.17.83886080 every 5 second(s)
 Taking delta samples, last value was 0
 Rising threshold is 5, assigned to event 1
 Falling threshold is 0, assigned to event 0
 On startup enable rising or falling alarm
```

Configuring RMON Events

You can configure RMON events to associate with RMON alarms. You can reuse the same event with multiple RMON alarms.

Ensure you have configured an SNMP user and enabled SNMP notifications.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **rmon event** *index* [**description** *string*] [**log**] [**trap**] [**owner** *name*]	Configures an RMON event. The description string and owner name can be any alphanumeric string.
Step 3	switch(config)# **show rmon** {**alarms** | **hcalarms**}	(Optional) Displays information about RMON alarms or high-capacity alarms.
Step 4	switch# **copy running-config startup-config**	(Optional) Saves this configuration change.

Verifying RMON Configuration

To display RMON configuration information, perform one of the following tasks:

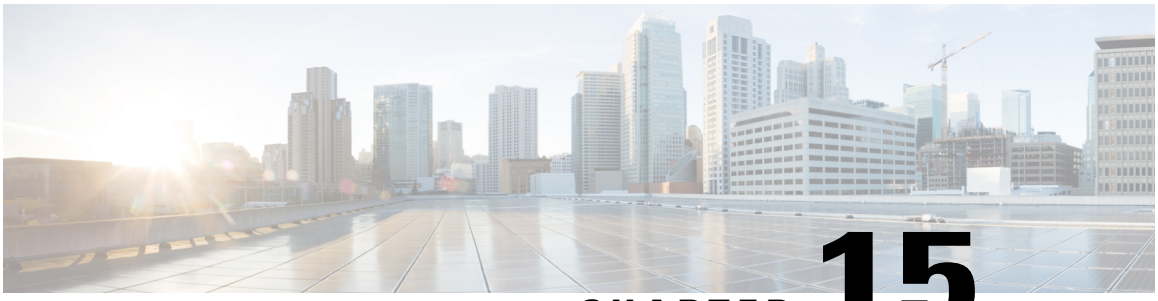
Command	Purpose
switch# **show rmon alarms**	Displays information about RMON alarms.
switch# **show rmon events**	Displays information about RMON events.
switch# **show rmon hcalarms**	Displays information about RMON hcalarms.
switch# **show rmon logs**	Displays information about RMON logs.

Default RMON Settings

The following table lists the default settings for RMON parameters.

Table 30: Default RMON Parameters

Parameters	Default
Alarms	None configured.
Events	None configured.



CHAPTER 15

Configuring SPAN

This chapter contains the following sections:

Information About SPAN

The Switched Port Analyzer (SPAN) feature (sometimes called port mirroring or port monitoring) selects network traffic for analysis by a network analyzer. The network analyzer can be a Cisco SwitchProbe or other Remote Monitoring (RMON) probes.

SPAN Sources

SPAN sources refer to the interfaces from which traffic can be monitored. The Cisco Nexus Series device supports Ethernet, port channels, and VLANs as SPAN sources. With VLANs, all supported interfaces in the specified VLAN are included as SPAN sources. You can choose the SPAN traffic in the ingress direction, the egress direction, or both directions for Ethernet source interfaces:

- Ingress source (Rx)—Traffic entering the device through this source port is copied to the SPAN destination port.
- Egress source (Tx)—Traffic exiting the device through this source port is copied to the SPAN destination port.

Characteristics of Source Ports

A source port, also called a monitored port, is a switched interface that you monitor for network traffic analysis. The switch supports any number of ingress source ports (up to the maximum number of available ports on the switch) and any number of source VLANs.

A source port has these characteristics:

- Can be of Ethernet, port channel, or VLAN port type.
- Cannot be monitored in multiple SPAN sessions.
- Cannot be a destination port.
- Each source port can be configured with a direction (ingress, egress, or both) to monitor. For VLAN sources, the monitored direction can only be ingress and applies to all physical ports in the group. The RX/TX option is not available for VLAN SPAN sessions.
- Source ports can be in the same or different VLANs.

SPAN Destinations

SPAN destinations refer to the interfaces that monitors source ports. The Cisco Nexus Series device supports Ethernet interfaces as SPAN destinations.

Source SPAN	Dest SPAN
Ethernet	Ethernet

Characteristics of Destination Ports

Each local SPAN session must have a destination port (also called a monitoring port) that receives a copy of traffic from the source ports or VLANs. A destination port has these characteristics:

- Can be any physical port. Source ethernet ports cannot be destination ports.

- Cannot be a source port.
- Cannot be a port channel.
- Does not participate in spanning tree while the SPAN session is active.
- Is excluded from the source list and is not monitored if it belongs to a source VLAN of any SPAN session.
- Receives copies of sent and received traffic for all monitored source ports. If a destination port is oversubscribed, it can become congested. This congestion can affect traffic forwarding on one or more of the source ports.

Guidelines and Limitations for SPAN

SPAN has the following guidelines and limitations:

- If you install NX-OS 5.0(3)U2(2) and then downgrade to a lower version of software, the SPAN configuration is lost.

  To avoid this, you need to save the configuration before upgrading to NX-OS 5.0(3)U2(2), and then reapply the local span configurations after the downgrade.

  For information about a similar ERSPAN limitation, see Guidelines and Limitations for ERSPAN, on page 158 for ERSPAN.

Creating or Deleting a SPAN Session

You create a SPAN session by assigning a session number using the **monitor session** command. If the session already exists, any additional configuration information is added to the existing session.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **monitor session** *session-number*	Enters the monitor configuration mode. New session configuration is added to the existing session configuration.

This example shows how to configure a SPAN monitor session:

```
switch# configure terminal
switch(config) # monitor session 2
switch(config) #
```

Configuring an Ethernet Destination Port

You can configure an Ethernet interface as a SPAN destination port.

Note The SPAN destination port can only be a physical port on the switch.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **interface ethernet** *slot*/*port*	Enters interface configuration mode for the Ethernet interface with the specified slot and port.
Step 3	switch(config-if)# **switchport monitor**	Enters monitor mode for the specified Ethernet interface. Priority flow control is disabled when the port is configured as a SPAN destination.
Step 4	switch(config-if)# **exit**	Reverts to global configuration mode.
Step 5	switch(config)# **monitor session** *session-number*	Enters monitor configuration mode for the specified SPAN session.
Step 6	switch(config-monitor)# **destination interface ethernet** *slot*/*port*	Configures the Ethernet SPAN destination port.

The following example shows how to configure an Ethernet SPAN destination port:

```
switch# configure terminal
switch(config)# interface ethernet 1/3
switch(config-if)# switchport monitor
switch(config-if)# exit
switch(config)# monitor session 2
switch(config-monitor)# destination interface ethernet 1/3
switch(config-monitor)#
```

Configuring Source Ports

Source ports can only be Ethernet ports.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **monitor session** *session-number*	Enters monitor configuration mode for the specified monitoring session.
Step 3	switch(config-monitor) # **source interface** *type slot*/*port* [**rx** | **tx** | **both**]	Configures sources and the traffic direction in which to duplicate packets. You can enter a range of Etherne ports. You can specify the traffic direction to duplicate as ingress (rx), egress (tx), or both. By default, the direction is both.

	Command or Action	Purpose

The following example shows how to configure an Ethernet SPAN source port:

```
switch# configure terminal
switch(config)# monitor session 2
switch(config-monitor)# source interface ethernet 1/16
switch(config-monitor)#
```

Configuring Source Port Channels or VLANs

You can configure the source channels for a SPAN session. These ports can be port channels, and VLANs. The monitored direction can be ingress, egress, or both and applies to all physical ports in the group.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **monitor session** *session-number*	Enters monitor configuration mode for the specified SPAN session.
Step 3	switch(config-monitor) # **source** {**interface** {**port-channel**} *channel-number* [**rx** | **tx** | **both**] | **vlan** *vlan-range*}	Configures port channel, or VLAN sources. For VLAN sources, the monitored direction is implicit.

This example shows how to configure a port channel SPAN source:

```
switch# configure terminal
switch(config)# monitor session 2
switch(config-monitor)# source interface port-channel 1 rx
switch(config-monitor)# source interface port-channel 3 tx
switch(config-monitor)# source interface port-channel 5 both
switch(config-monitor)#
```

This example shows how to configure a VLAN SPAN source:

```
switch# configure terminal
switch(config)# monitor session 2
switch(config-monitor)# source vlan 1
switch(config-monitor)#
```

Configuring the Description of a SPAN Session

For ease of reference, you can provide a descriptive name for a SPAN session.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **monitor session** *session-number*	Enters monitor configuration mode for the specified SPAN session.
Step 3	switch(config-monitor) # **description** *description*	Creates descriptive name for the SPAN session.

The following example shows how to configure a SPAN session description:

```
switch# configure terminal
switch(config) # monitor session 2
switch(config-monitor) # description monitoring ports eth2/2-eth2/4
switch(config-monitor) #
```

Activating a SPAN Session

The default is to keep the session state shut. You can open a session that duplicates packets from sources to destinations.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **no monitor session** {**all** | *session-number*} **shut**	Opens the specified SPAN session or all sessions.

The following example shows how to activate a SPAN session:

```
switch# configure terminal
switch(config) # no monitor session 3 shut
```

Suspending a SPAN Session

By default, the session state is **shut**.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.

	Command or Action	Purpose
Step 2	switch(config) # **monitor session** {**all** | *session-number*} **shut**	Suspends the specified SPAN session or all sessions.

The following example shows how to suspend a SPAN session:

```
switch# configure terminal
switch(config) # monitor session 3 shut
switch(config) #
```

Displaying SPAN Information

Procedure

	Command or Action	Purpose
Step 1	switch# **show monitor** [**session** {**all** | *session-number* | **range** *session-range*} [**brief**]]	Displays the SPAN configuration.

This example shows how to display SPAN session information:

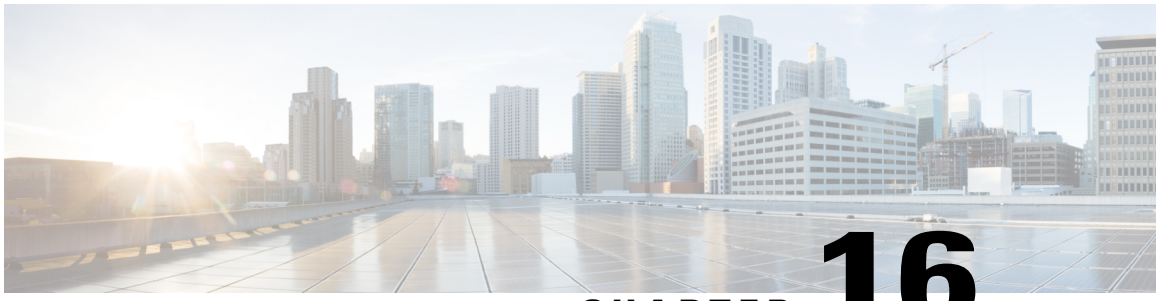
```
switch# show monitor
SESSION  STATE        REASON                  DESCRIPTION
-------  -----------  ----------------------  --------------------------------
2        up           The session is up
3        down         Session suspended
4        down         No hardware resource
```

This example shows how to display SPAN session details:

```
switch# show monitor session 2
   session 2
---------------
type              : local
state             : up
source intf       :

source VLANs      :
    rx            :

destination ports : Eth3/1
```



CHAPTER 16

Configuring ERSPAN

This chapter includes the following sections:

Information About ERSPAN

The Cisco NX-OS system supports the Encapsulated Remote Switching Port Analyser (ERSPAN) feature on both source and destination ports. ERSPAN transports mirrored traffic over an IP network. The traffic is encapsulated at the source router and is transferred across the network. The packet is decapsulated at the destination router and then sent to the destination interface.

ERSPAN consists of an ERSPAN source session, routable ERSPAN generic routing encapsulation (GRE)-encapsulated traffic, and an ERSPAN destination session. You separately configure ERSPAN source sessions and destination sessions on different switches.

ERSPAN Sources

The interfaces from which traffic can be monitored are called ERSPAN sources. Sources designate the traffic to monitor and whether to copy ingress, egress, or both directions of traffic. ERSPAN sources include the following:

- Ethernet ports and port channels.

- VLANs—When a VLAN is specified as an ERSPAN source, all supported interfaces in the VLAN are ERSPAN sources.

ERSPAN source ports have the following characteristics:

- A port configured as a source port cannot also be configured as a destination port.
- ERSPAN does not monitor any packets that are generated by the supervisor, regardless of their source.

ERSPAN Destinations

Destination ports receive the copied traffic from ERSPAN sources.

ERSPAN destination ports have the following characteristics:

- Destinations for an ERSPAN session include Ethernet ports or port-channel interfaces in either access or trunk mode.
- A port configured as a destination port cannot also be configured as a source port.
- A destination port can be configured in only one ERSPAN session at a time.
- Destination ports do not participate in any spanning tree instance or any Layer 3 protocols.
- Ingress and ingress learning options are not supported on monitor destination ports
- HIF port channels, and fabric port channel ports are not supported as SPAN destination ports.

ERSPAN Sessions

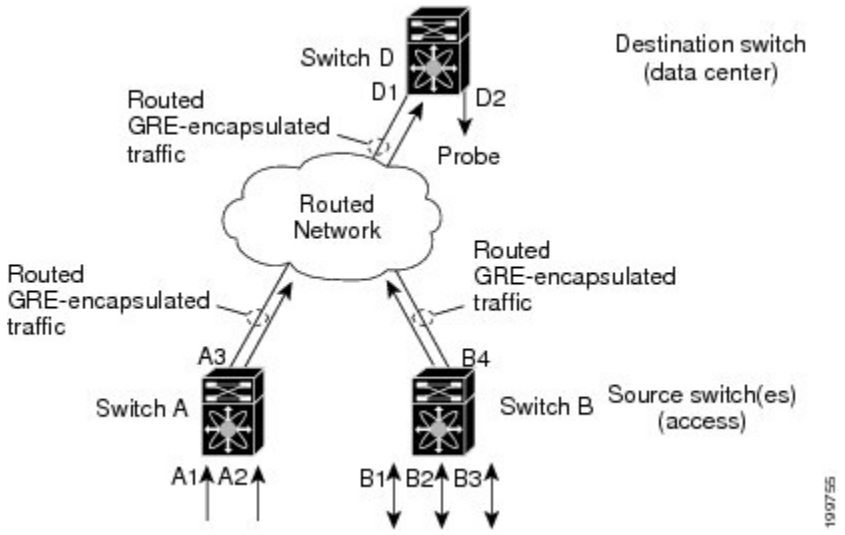
You can create ERSPAN sessions that designate sources and destinations to monitor.

When configuring ERSPAN source sessions, you need to configure the destination IP address. When configuring ERSPAN destination sessions, you need to configure the source IP address. See ERSPAN Sources, on page 155 for the properties of source sessions and ERSPAN Destinations, on page 156 for the properties of destination sessions.

Note Only two ERSPAN or SPAN source sessions can run simultaneously across all switches. Only 23 ERSPAN destination sessions can run simultaneously across all switches.

The following figure shows an ERSPAN configuration.

Figure 1: ERSPAN Configuration



Multiple ERSPAN Sessions

Although you can define up to 48 ERSPAN sessions, only two ERSPAN or SPAN sessions can be running simultaneously. You can shut down any unused ERSPAN sessions.

For information about shutting down ERSPAN sessions, see the Shutting Down or Activating an ERSPAN Session, on page 164.

High Availability

The ERSPAN feature supports stateless and stateful restarts. After a reboot or supervisor switchover, the running configuration is applied.

Licensing Requirements for ERSPAN

The following table shows the licensing requirements for this feature:

Product	License Requirement
Cisco NX-OS	ERSPAN requires no license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you. For a complete explanation of the Cisco NX-OS licensing scheme, see the *License and Copyright Information for Cisco NX-OS Software* available at the following URL: http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/license_agreement/nx-ossw_lisns.html.

Prerequisites for ERSPAN

ERSPAN has the following prerequisite:

•You must first configure the Ethernet interfaces for ports on each device to support the desired ERSPAN configuration.

Guidelines and Limitations for ERSPAN

ERSPAN has the following configuration guidelines and limitations:

- ERSPAN supports the following:
  - From 4 to 6 tunnels
  - Non-tunnel packets
  - IP-in-IP tunnels
  - IPv4 tunnels (limited)
  - ERSPAN source session type (Packets are encapsulated as GRE-tunnel packets and sent on the IP network. However, unlike other Cisco devices, the ERSPAN header is not added to the packet.)
  - ERSPAN destination session type (However, support for decapsulating the ERSPAN packet is not available. The entire encapsulated packet is spanned to a front panel port at the ERSPAN terminating point.)
- ERSPAN packets are dropped if the encapsulated mirror packet fails Layer 2 MTU checks.
- There is a 112-byte limit for egress encapsulation. Packets exceeding this limit are dropped. This scenario might be encountered when tunnels and mirroring are intermixed.
- ERSPAN sessions are shared with local sessions. A maximum of 18 sessions can be configured; however only a maximum of four sessions can be operational at the same time. If both receive and transmit sources are configured in the same session, then only two sessions can be operational.
- If you install NX-OS 5.0(3)U2(2), configure ERSPAN, and then downgrade to a lower version of software, the ERSPAN configuration is lost. This situation occurs because ERSPAN is not supported in versions before NX-OS 5.0(3)U2(2).

For information about a similar SPAN limitation, see Guidelines and Limitations for SPAN, on page 149 for SPAN.

- ERSPAN and ERSPAN ACLs are not supported for packets generated by the supervisor.
- ERSPAN and ERSPAN ACL sessions are terminated identically at the destination router.
- ERSPAN is not supported for management ports.
- A destination port can be configured in only one ERSPAN session at a time.
- You cannot configure a port as both a source and destination port.
- A single ERSPAN session can include mixed sources in any combination of the following:
  - Ethernet ports or port channels but not subinterfaces.
  - VLANs or port channels, which can be assigned to port channel subinterfaces.
  - The port channels to the control plane CPU.

Note ERSPAN does not monitor any packets that are generated by the supervisor, regardless of their source.

- Destination ports do not participate in any spanning tree instance or Layer 3 protocols.
- When an ERSPAN session contains source ports that are monitored in the transmit or transmit and receive direction, packets that these ports receive may be replicated to the ERSPAN destination port even though the packets are not actually transmitted on the source ports. Some examples of this behavior on source ports include:
  - Traffic that results from flooding
  - Broadcast and multicast traffic
- For VLAN ERSPAN sessions with both ingress and egress configured, two packets (one from ingress and one from egress) are forwarded from the destination port if the packets get switched on the same VLAN.
- VLAN ERSPAN monitors only the traffic that leaves or enters Layer 2 ports in the VLAN.
- When packets are mirrored and sent to the ERSPAN destination port, GRE headers are not stripped off. Packets are sent along with the GRE headers as GRE packets with the original packet as the GRE payload.

Default Settings

The following table lists the default settings for ERSPAN parameters.

Table 31: Default ERSPAN Parameters

Parameters	Default
ERSPAN sessions	Created in the shut state.

Configuring ERSPAN

Configuring an ERSPAN Source Session

You can configure an ERSPAN session on the local device only. By default, ERSPAN sessions are created in the shut state.

For sources, you can specify Ethernet ports, port channels, and VLANs. A single ERSPAN session can include mixed sources in any combination of Ethernet ports or VLANs.

Note ERSPAN does not monitor any packets that are generated by the supervisor, regardless of their source.

Procedure

	Command or Action	Purpose

Step 1	**config t**
	
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**monitor erspan origin ip-address** *ip-address* **global** **Example:** `switch(config)# monitor erspan origin ip-address 10.0.0.1 global`	Configures the ERSPAN global origin IP address.
Step 3	**no monitor session** {*session-number* | **all**} **Example:** `switch(config)# no monitor session 3`	Clears the configuration of the specified ERSPAN session. The new session configuration is added to the existing session configuration.

Step 4	**monitor session** {*session-number* | **all**} **type erspan-source**
	
	Example:
	`switch(config)# monitor session 3 type erspan-source`
	`switch(config-erspan-src)#`
		Configures an ERSPAN source session.

Step 5	**description** *description* **Example:** `switch(config-erspan-src)# description erspan_src_session_3`	Configures a description for the session. By default, no description is defined. The description can be up to 32 alphanumeric characters.
Step 6	**source** {[**interface** [*type slot/port*[*-port*][*, type slot/port[-port*]]] [**port-channel**	Configures the sources and traffic direction in which to copy packets. You can enter a range of

	Command or Action	Purpose

	channel-number]] | [**vlan** {*number* | *range*}]} [**rx** | **tx** | **both**]
	
	Example:
	`switch(config-erspan-src)# source interface ethernet 2/1-3, ethernet 3/1 rx`
	
	Example:
	`switch(config-erspan-src)# source interface port-channel 2`
	
	Example:
	`switch(config-erspan-src)# source interface sup-eth 0 both`
	
	Example:
	`switch(config-erspan-src)# source vlan 3, 6-8 tx`
	
	Example:
	`switch(config-monitor)# source interface ethernet 101/1/1-3`
		Ethernet ports, a port channel, or a range of VLANs.
		
		You can configure one or more sources, as either a series of comma-separated entries or a range of numbers. You can specify up to 128 interfaces. For information on the VLAN range, see the *Cisco Nexus 3000 Series NX-OS Layer 2 Switching Configuration Guide, Release 5.x*.
		
		You can specify the traffic direction to copy as ingress, egress, or both. The default direction is both.

Step 7	Repeat Step 6 to configure all ERSPAN sources.	(Optional) —
Step 8	**destination ip** *ip-address* **Example:** `switch(config-erspan-src)# destination ip 10.1.1.1`	Configures the destination IP address in the ERSPAN session. Only one destination IP address is supported per ERSPAN source session.
Step 9	**vrf** *vrf-name* **Example:** `switch(config-erspan-src)# vrf default`	Configures the VRF that the ERSPAN source session uses for traffic forwarding.
Step 10	**ip ttl** *ttl-number* **Example:** `switch(config-erspan-src)# ip ttl 25`	(Optional) Configures the IP time-to-live (TTL) value for the ERSPAN traffic. The range is from 1 to 255.
Step 11	**ip dscp** *dscp-number* **Example:** `switch(config-erspan-src)# ip dscp 42`	(Optional) Configures the differentiated services code point (DSCP) value of the packets in the ERSPAN traffic. The range is from 0 to 63.
Step 12	**no shut** **Example:** `switch(config-erspan-src)# no shut`	Enables the ERSPAN source session. By default, the session is created in the shut state. **Note** Only two ERSPAN source sessions can be running simultaneously.

	Command or Action	Purpose
Step 13	**show monitor session** {**all** | *session-number* | **range** *session-range*} **Example:** `switch(config-erspan-src)# show monitor session 3`	(Optional) Displays the ERSPAN session configuration.
Step 14	**show running-config monitor** **Example:** `switch(config-erspan-src)# show running-config monitor`	(Optional) Displays the running ERSPAN configuration.
Step 15	**show startup-config monitor** **Example:** `switch(config-erspan-src)# show startup-config monitor`	(Optional) Displays the ERSPAN startup configuration.
Step 16	**copy running-config startup-config** **Example:** `switch(config-erspan-src)# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.

Configuring an ERSPAN Destination Session

You can configure an ERSPAN destination session to copy packets from a source IP address to destination ports on the local device. By default, ERSPAN destination sessions are created in the shut state.

Before You Begin

Ensure that you have already configured the destination ports in monitor mode.

Procedure

	Command or Action	Purpose

Step 1	**config t**
	
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface ethernet** *slot/port*[*-port*]
	
	Example:
	`switch(config)# interface ethernet 2/5`
	`switch(config-if)#`
		Enters interface configuration mode on the selected slot and port or range of ports.

	Command or Action	Purpose
Step 3	**switchport** **Example:** `switch(config-if)# switchport`	Configures switchport parameters for the selected slot and port or range of ports.
Step 4	**switchport mode** [**access** | **trunk**] **Example:** `switch(config-if)# switchport mode trunk`	Configures the following switchport modes for the selected slot and port or range of ports: • access • trunk
Step 5	**switchport monitor** **Example:** `switch(config-if)# switchport monitor`	Configures the switchport interface as an ERSPAN destination.
Step 6	Repeat Steps 2 to 5 to configure monitoring on additional ERSPAN destinations.	—
Step 7	**no monitor session** {*session-number* | **all**} **Example:** `switch(config-if)# no monitor session 3`	Clears the configuration of the specified ERSPAN session. The new session configuration is added to the existing session configuration.

Step 8	**monitor session** {*session-number* | **all**} **type erspan-destination**
	
	Example:
	`switch(config-if)# monitor session 3 type erspan-destination`
	`switch(config-erspan-dst)#`
		Configures an ERSPAN destination session.

Step 9	**description** *description* **Example:** `switch(config-erspan-dst)# description erspan_dst_session_3`	Configures a description for the session. By default, no description is defined. The description can be up to 32 alphanumeric characters.
Step 10	**source ip** *ip-address* **Example:** `switch(config-erspan-dst)# source ip 10.1.1.1`	Configures the source IP address in the ERSPAN session. Only one source IP address is supported per ERSPAN destination session.
Step 11	**destination** {[**interface** [*type slot/port*[*-port*][*, type slot/port[-port*]]] [**port-channel** *channel-number*]]} **Example:** `switch(config-erspan-dst)# destination interface ethernet 2/5, ethernet 3/7`	Configures a destination for copied source packets. You can configure one or more interfaces as a series of comma-separated entries. **Note** You can configure destination ports as trunk ports.

	Command or Action	Purpose
Step 12	Repeat Step 11 to configure all ERSPAN destinations.	(Optional) —
Step 13	**no shut** **Example:** `switch(config)# no shut`	Enables the ERSPAN destination session. By default, the session is created in the shut state. **Note** Only 23 ERSPAN destination sessions can be running simultaneously.
Step 14	**show monitor session** {**all** | *session-number* | **range** *session-range*} **Example:** `switch(config)# show monitor session 3`	(Optional) Displays the ERSPAN session configuration.
Step 15	**show running-config monitor** **Example:** `switch(config-erspan-src)# show running-config monitor`	(Optional) Displays the running ERSPAN configuration.
Step 16	**show startup-config monitor** **Example:** `switch(config-erspan-src)# show startup-config monitor`	(Optional) Displays the ERSPAN startup configuration.
Step 17	**copy running-config startup-config** **Example:** `switch(config-erspan-src)# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.

Shutting Down or Activating an ERSPAN Session

You can shut down ERSPAN sessions to discontinue the copying of packets from sources to destinations. Because only two ERSPAN sessions can be running simultaneously, you can shut down a session in order to free hardware resources to enable another session. By default, ERSPAN sessions are created in the shut state.

You can enable ERSPAN sessions to activate the copying of packets from sources to destinations. To enable an ERSPAN session that is already enabled but operationally down, you must first shut it down and then enable it. You can shut down and enable the ERSPAN session states with either a global or monitor configuration mode command.

Procedure

	Command or Action	Purpose

Step 1	**configuration terminal**
	
	Example:
	`switch# configuration terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**monitor session** {*session-range* | **all**} **shut** **Example:** `switch(config)# monitor session 3 shut`	Shuts down the specified ERSPAN sessions. The session range is from 1 to 48. By default, sessions are created in the shut state. Only two sessions can be running at a time.
Step 3	**no monitor session** {*session-range* | **all**} **shut** **Example:** `switch(config)# no monitor session 3 shut`	Resumes (enables) the specified ERSPAN sessions. The session range is from 1 to 48. By default, sessions are created in the shut state. Only two sessions can be running at a time. **Note** If a monitor session is enabled but its operational status is down, then to enable the session, you must first specify the **monitor session shut** command followed by the **no monitor session shut** command.

Step 4	**monitor session** *session-number* **type erspan-source**
	
	Example:
	`switch(config)# monitor session 3 type erspan-source`
	`switch(config-erspan-src)#`
		Enters the monitor configuration mode for the ERSPAN source type. The new session configuration is added to the existing session configuration.

Step 5	**monitor session** *session-number* **type erspan-destination** **Example:** `switch(config-erspan-src)# monitor session 3 type erspan-destination`	Enters the monitor configuration mode for the ERSPAN destination type.
Step 6	**shut** **Example:** `switch(config-erspan-src)# shut`	Shuts down the ERSPAN session. By default, the session is created in the shut state.
Step 7	**no shut** **Example:** `switch(config-erspan-src)# no shut`	Enables the ERSPAN session. By default, the session is created in the shut state.
Step 8	**show monitor session all** **Example:** `switch(config-erspan-src)# show monitor session all`	(Optional) Displays the status of ERSPAN sessions.

	Command or Action	Purpose
Step 9	**show running-config monitor** **Example:** `switch(config-erspan-src)# show running-config monitor`	(Optional) Displays the running ERSPAN configuration.
Step 10	**show startup-config monitor** **Example:** `switch(config-erspan-src)# show startup-config monitor`	(Optional) Displays the ERSPAN startup configuration.
Step 11	**copy running-config startup-config** **Example:** `switch(config-erspan-src)# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.

Verifying the ERSPAN Configuration

To display the ERSPAN configuration, perform one of the following tasks:

Command	Purpose
show monitor session {**all** | *session-number* | **range** *session-range*}	Displays the ERSPAN session configuration.
show running-config monitor	Displays the running ERSPAN configuration.
show startup-config monitor	Displays the ERSPAN startup configuration.

Configuration Examples for ERSPAN

Configuration Example for an ERSPAN Source Session

This example shows how to configure an ERSPAN source session:

```
switch# config t
switch(config)# interface e14/30
switch(config-if)# no shut
switch(config-if)# exit
switch(config)# monitor erspan origin ip-address 3.3.3.3 global
switch(config)# monitor session 1 type erspan-source
switch(config-erspan-src)# source interface e14/30
switch(config-erspan-src)# ip ttl 16
switch(config-erspan-src)# ip dscp 5
switch(config-erspan-src)# vrf default
```

```
switch(config-erspan-src)# destination ip 9.1.1.2
switch(config-erspan-src)# no shut
switch(config-erspan-src)# exit
switch(config)# show monitor session 1
```

Configuration Example for an ERSPAN Destination Session

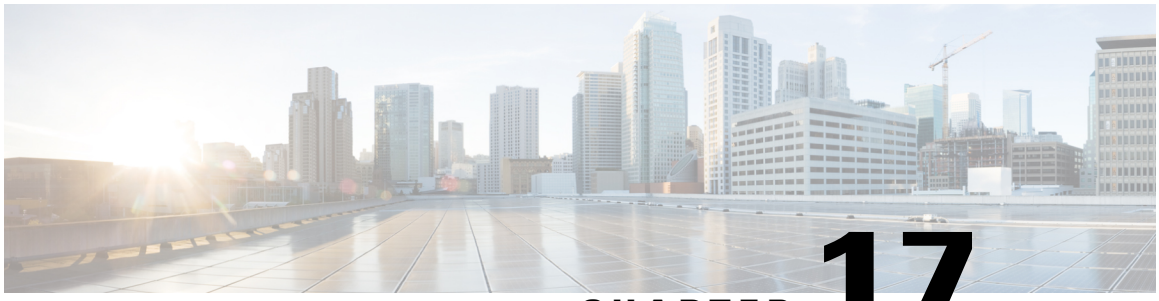
This example shows how to configure an ERSPAN destination session:

```
switch# config t
switch(config)# interface e14/29
switch(config-if)# no shut
switch(config-if)# switchport
switch(config-if)# switchport monitor
switch(config-if)# exit
switch(config)# monitor session 2 type erspan-destination
switch(config-erspan-dst)# source ip 9.1.1.2
switch(config-erspan-dst)# destination interface e14/29
switch(config-erspan-dst)# no shut
switch(config-erspan-dst)# exit
switch(config)# show monitor session 2
```

Additional References

Related Documents

Related Topic	Document Title
ERSPAN commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco Nexus 3000 Series NX-OS System Management Command ReferenceCisco Nexus 5000 Series NX-OS System Management Command Reference*



CHAPTER 17

Configuring sFLOW

This chapter contains the following sections:

Information About sFlow

sFlow allows you to monitor the real-time traffic in data networks that contain switches and routers. It uses the sampling mechanism in the sFlow Agent software on switches and routers for monitoring traffic and to forward the sample data on ingress and egress ports to the central data collector, also called the sFlow Analyzer.

For more information about sFlow, see RFC 3176.

sFlow Agent

The sFlow Agent, which is embedded in the Cisco NX-OS software, periodically samples or polls the interface counters that are associated with a data source of the sampled packets. The data source can be an Ethernet interface, an EtherChannel interface, or a range of Ethernet interfaces. The sFlow Agent queries the Ethernet port manager for the respective EtherChannel membership information and also receives notifications from the Ethernet port manager for membership changes.

When you enable sFlow sampling in the Cisco NX-OS software, based on the sampling rate and the hardware internal random number, the ingress packets and egress packets are sent to the CPU as an sFlow-sampled packet. The sFlow Agent processes the sampled packets and sends an sFlow datagram to the sFlow Analyzer. In addition to the original sampled packet, an sFlow datagram includes the information about the ingress port, egress port, and the original packet length. An sFlow datagram can have multiple sFlow samples.

Licensing Requirements

This feature does not require a license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you. For a complete explanation of the Cisco NX-OS licensing scheme, see the *Cisco NX-OS Licensing Guide*.

Prerequisites

You must enable the sFlow feature using the **feature sflow** command to configure sFlow.

Guidelines and Limitations for sFlow

When you plan your sFlow configuration, consider the following:

- When you enable sFlow for an interface, it is enabled for both ingress and egress. You cannot enable sFlow for only ingress or only egress.
- sFlow egress sampling for multicast, broadcast, or unknown unicast packets is not supported.
- You should configure the sampling rate based on the sFlow configuration and traffic in the system.
- Cisco Nexus 3000 Series supports only one sFlow collector.

Default Settings for sFlow

Table 32: Default sFlow Parameters

Parameters	Default
sFlow sampling-rate	4096
sFlow sampling-size	128
sFlow max datagram-size	1400
sFlow collector-port	6343
sFlow counter-poll-interval	20

Configuring sFLow

Enabling the sFlow Feature

You must enable the sFlow feature before you can configure sFlow on the switch.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	[**no**] **feature sflow**	Enables the sFlow feature.
Step 3	**show feature**	(Optional) Displays enabled and disabled features.
Step 4	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to enable the sFlow feature:

```
switch# configure terminal
switch(config)# feature sflow
switch(config)# copy running-config startup-config
```

Configuring the Sampling Rate

Before You Begin

Ensure that you have enabled the sFlow feature.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	[**no**] **sflow sampling-rate** *sampling-rate*	Configures the sFlow sampling rate for packets. The *sampling-rate* can be an integer between 4096-1000000000. The default value is 4096. **Note** A *sampling-rate* of 0 disables sampling.

	Command or Action	Purpose
Step 3	**show sflow**	(Optional) Displays sFlow information.
Step 4	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to set the sampling rate to 50,000:

```
switch# configure terminal
switch(config)# sflow sampling-rate 50000
switch(config)# copy running-config startup-config
```

Configuring the Maximum Sampled Size

You can configure the maximum number of bytes that should be copied from a sampled packet.

Before You Begin

Ensure that you have enabled the sFlow feature.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	[**no**] **sflow max-sampled-size** *sampling-size*	Configures the sFlow maximum sampling size packets. The range for the *sampling-size* is from 64 to 256 bytes. The default value is 128.
Step 3	**show sflow**	(Optional) Displays sFlow information.
Step 4	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to configure the maximum sampling size for the sFlow Agent:

```
switch# configure terminal
switch(config)# sflow max-sampled-size 200
switch(config)# copy running-config startup-config
```

Configuring the Counter Poll Interval

You can configure the maximum number of seconds between successive samples of the counters that are associated with the data source. A sampling interval of 0 disables counter sampling.

Before You Begin

Ensure that you have enabled the sFlow feature.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	[**no**] **sflow counter-poll-interval** *poll-interval*	Configures the sFlow poll interval for an interface. The range for the *poll-interval* is from 0 to 2147483647 seconds. The default value is 20.
Step 3	**show sflow**	(Optional) Displays sFlow information.
Step 4	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to configure the sFlow poll interval for an interface:

```
switch# configure terminal
switch(config)# sflow counter-poll-interval 100
switch(config)# copy running-config startup-config
```

Configuring the Maximum Datagram Size

You can configure the maximum number of data bytes that can be sent in a single sample datagram.

Before You Begin

Ensure that you have enabled the sFlow feature.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	[**no**] **sflow max-datagram-size** *datagram-size*	Configures the sFlow maximum datagram size. The range for the *datagram-size* is from 200 to 9000 bytes. The default value is 1400.

	Command or Action	Purpose
Step 3	**show sflow**	(Optional) Displays sFlow information.
Step 4	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to configure the sFlow maximum datagram size:

```
switch# configure terminal
switch(config)# sflow max-datagram-size 2000
switch(config)# copy running-config startup-config
[########################################] 100%
```

Configuring the sFlow Analyzer Address

Before You Begin

Ensure that you have enabled the sFlow feature.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	[**no**] **sflow collector-ip** *IP-address vrf-instance*	Configures the IPv4 address for the sFlow Analyzer. *vrf-instance* can be one of the following: • A user-defined VRF name. You can specify a maximum of 32 alphanumeric characters. • **vrf management**. You must use this option if the sFlow data collector is on the network connected to the management port. • **vrf default**. You must use this option if the sFlow data collector is on the network connected to the front panel ports.
Step 3	**show sflow**	(Optional) Displays sFlow information.
Step 4	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to configure the IPv4 address of the sFlow data collector that is connected to the management port:

```
switch# configure terminal
switch(config)# sflow collector-ip 192.0.2.5 vrf management
switch(config)# copy running-config startup-config
```

Configuring the sFlow Analyzer Port

You can configure the destination port for sFlow datagrams.

Before You Begin

Ensure that you have enabled the sFlow feature.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	[**no**] **sflow collector-port** *collector-port*	Configures the UDP port of the sFlow Analyzer. The range for the *collector-port* is from 0 to 65535. The default value is 6343.
Step 3	**show sflow**	(Optional) Displays sFlow information.
Step 4	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to configure the destination port for sFlow datagrams:

```
switch# configure terminal
switch(config)# sflow collector-port 7000
switch(config)# copy running-config startup-config
[########################################] 100%
switch(config)#
```

Configuring the sFlow Agent Address

Before You Begin

Ensure that you have enabled the sFlow feature.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	[**no**] **sflow agent-ip** *ip-address*	Configures the IPv4 address of the sFlow Agent. The default *ip-address* is 0.0.0.0, which means that all sampling is disabled on the switch. You must specify a valid IP address to enable sFlow functionality.
Step 3	**show sflow**	(Optional) Displays sFlow information.
Step 4	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to configure the IPv4 address of the sFlow Agent:

```
switch# configure terminal
switch(config)# sflow agent-ip 192.0.2.3
switch(config)# copy running-config startup-config
```

Configuring the sFlow Sampling Data Source

The sFlow sampling data source can be an Ethernet port, a range of Ethernet ports, or a port channel.

Before You Begin

- Ensure that you have enabled the sFlow feature.
- If you want to use a port channel as the data source, ensure that you have already configured the port channel and you know the port channel number.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# [**no**] **sflow data-source interface** [**ethernet** *slot*/*port*[*-port*] |**port-channel** *channel-number*]	Configures the sFlow sampling data source. For an Ethernet data source, *slot* is the slot number and *port* can be either a single port number or a range of ports designated as *port-port*.
Step 3	switch(config)# **show sflow**	(Optional) Displays sFlow information.

	Command or Action	Purpose
Step 4	switch(config)# **copy running-config startup-config**	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to configure Ethernet ports 5 through 12 for the sFlow sampler:

```
switch# configure terminal
switch(config)# sflow data-source interface ethernet 1/5-12
switch(config)# copy running-config startup-config
[########################################] 100%
switch(config)#
```

This example shows how to configure port channel 100 for the sFlow sampler:

```
switch# configure terminal
switch(config)# sflow data-source interface port-channel 100
switch(config)# copy running-config startup-config
[########################################] 100%
switch(config)#
```

sFLOW Show Commands

To display the sFlow configuration information, perform one of the following tasks:

Command	Purpose
show sflow	Displays the sFlow global configuration.
show sflow statistics	Displays the sFlow statistics.
clear sflow statistics	Clears the sFlow statistics.
show running-config sflow [**all**]	Displays the current running sFlow configuration.

Configuration Examples for sFlow

This example shows how to configure sFlow:

```
feature sflow
sflow sampling-rate 5000
sflow max-sampled-size 200
sflow counter-poll-interval 100
sflow max-datagram-size 2000
sflow collector-ip 192.0.2.5 vrf management
sflow collector-port 7000
sflow agent-ip 192.0.2.3
sflow data-source interface ethernet 1/5
```

Additional References for sFlow

Table 33: Related Documents for sFlow

Related Topic	Document Title
sFlow CLI commands	*Cisco Nexus 3000 Series NX-OS System Management Command Reference.*
RFC 3176	Defines the sFlow packet format and SNMP MIB. http://www.sflow.org/rfc3176.txt

Feature History for sFlow

This table includes only the updates for those releases that have resulted in additions or changes to the feature.

Feature Name	Releases	Feature Information
sFlow	5.0(3)U4(1)	This feature was introduced.

INDEX

A

C

D

E

F

G

H

I

J

L

M

N

O

P

R

S

T

U

V