Cisco Nexus 5500 Series NX-OS System Management Configuration Guide, Release 7.x

First Published: 2014-01-29

Last Modified: 2022-02-18

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

Text Part Number: OL-30897-01

Preface

The preface contains the following sections:

Audience

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

Audience

This publication is for data center and cloud administrators who configure and maintain Cisco software, Cisco software appliances, and virtualization infrastructure.

This guide is for target for network and server administrators with the following experience and knowledge:

- An understanding of virtualization.
- Using Virtual Machine Manager (VMM) software to create a virtual machine.
- Configuration of a virtual switch, such as Cisco Nexus 1000V, VMware vSwitch.
- An understanding of public cloud provider utilization for supported providers such as Amazon Web Services (AWS), Microsoft Azure, Cisco Intercloud Services – V, Cisco Intercloud Services, and Cisco Intercloud Services.

Document Conventions

Note As part of our constant endeavor to remodel our documents to meet our customers' requirements, we have modified the manner in which we document configuration tasks. As a result of this, you may find a deviation in the style used to describe these tasks, with the newly included sections of the document following the new format.

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.
[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 Cisco 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

License Information

For information about feature licenses in NX-OS, see the *Cisco NX-OS Licensing Guide*, available at the following URL: https://www.cisco.com/c/en/us/td/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

Programming Guides

The XML Interface User Guide and other programming guides are available at the following URL:

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

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

Related Documentation for Cisco Nexus 5500 Series NX-OS Software

The entire Cisco NX-OS 5500 Series documentation set is available at the following URL:

http://www.cisco.com/c/en/us/support/switches/nexus-5000-series-switches/tsd-products-support-series-home.html

Release Notes

The release notes are available at the following URL:

http://www.cisco.com/en/US/products/ps9670/prod_release_notes_list.html

Configuration Guides

These guides are available at the following URL:

http://www.cisco.com/en/US/products/ps9670/products_installation_and_configuration_guides_list.html

The documents in this category include:

- *Cisco Nexus 5500 Series NX-OS Adapter-FEX Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS FabricPath Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS FCoE Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS Fundamentals Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS Interfaces Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS Layer 2 Switching Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS Multicast Routing Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS Quality of Service Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS SAN Switching Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS Security Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS System Management Configuration Guide*
- *Cisco Nexus 5500 Series NX-OS Unicast Routing Configuration Guide*

Installation and Upgrade Guides

These guides are available at the following URL:

http://www.cisco.com/en/US/products/ps9670/prod_installation_guides_list.html

The document in this category include:

- *Cisco Nexus 5500 Series NX-OS Software Upgrade and Downgrade Guides*

Licensing Guide

The *License and Copyright Information for Cisco NX-OS Software* is available at http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/license_agreement/nx-ossw_lisns.html.

Command References

These guides are available at the following URL:

http://www.cisco.com/en/US/products/ps9670/prod_command_reference_list.html

The documents in this category include:

- *Cisco Nexus 5500 Series NX-OS Fabric Extender Command Reference*
- *Cisco Nexus 5500 Series NX-OS FabricPath Command Reference*
- *Cisco Nexus 5500 Series NX-OS Fundamentals Command Reference*
- *Cisco Nexus 5500 Series NX-OS Interfaces Command Reference*
- *Cisco Nexus 5500 Series NX-OS Layer 2 Interfaces Command Reference*
- *Cisco Nexus 5500 Series NX-OS Multicast Routing Command Reference*
- *Cisco Nexus 5500 Series NX-OS Quality of Service Command Reference*
- *Cisco Nexus 5500 Series NX-OS Security Command Reference*
- *Cisco Nexus 5500 Series NX-OS System Management Command Reference*
- *Cisco Nexus 5500 Series NX-OS TrustSec Command Reference*
- *Cisco Nexus 5500 Series NX-OS Unicast Routing Command Reference*
- *Cisco Nexus 5500 Series NX-OS Virtual Port Channel Command Reference*

Technical References

The *Cisco Nexus 5500 Series NX-OS MIB Reference* is available at http://www.cisco.com/en/US/docs/switches/datacenter/nexus5500/sw/mib/reference/NX5500_MIBRef.html.

Error and System Messages

The *Cisco Nexus 5500 Series NX-OS System Message Guide* is available at http://www.cisco.com/en/US/docs/switches/datacenter/nexus5500/sw/system_messages/reference/sl_nxos_book.html.

Troubleshooting Guide

The *Cisco Nexus 5500 Series NX-OS Troubleshooting Guide* is available at http://www.cisco.com/en/US/docs/switches/datacenter/nexus5500/sw/troubleshooting/guide/N5K_Troubleshooting_Guide.html.

Related Documentation for Cisco Nexus 6000 Series NX-OS Software

The entire Cisco NX-OS 6000 Series documentation set is available at the following URL:

http://www.cisco.com/en/US/products/ps12806/tsd_products_support_series_home.html

Release Notes

The release notes are available at the following URL:

http://www.cisco.com/c/en/us/support/switches/nexus-6000-series-switches/products-release-notes-list.html

Configuration Guides

These guides are available at the following URL:

http://www.cisco.com/c/en/us/support/switches/nexus-6000-series-switches/products-installation-and-configuration-guides-list.html

The documents in this category include:

- *Cisco Nexus 6000 Series NX-OS Adapter-FEX Configuration Guide*
- *Cisco Nexus 6000 Series NX-OS FabricPath Configuration Guide*
- *Cisco Nexus 6000 Series NX-OS FCoE Configuration Guide*
- *Cisco Nexus 6000 Series NX-OS Fundamentals Configuration Guide*
- *Cisco Nexus 6000 Series NX-OS Interfaces Configuration Guide*
- *Cisco Nexus 6000 Series NX-OS Layer 2 Switching Configuration Guide*
- *Cisco Nexus 6000 Series NX-OS Multicast Routing Configuration Guide*
- *Cisco Nexus 6000 Series NX-OS Quality of Service Configuration Guide*
- *Cisco Nexus 6000 Series NX-OS SAN Switching Configuration Guide*
- *Cisco Nexus 6000 Series NX-OS Security Configuration Guide*
- *Cisco Nexus 6000 Series NX-OS System Management Configuration Guide*
- *Cisco Nexus 6000 Series NX-OS Unicast Routing Configuration Guide*

Installation and Upgrade Guides

These guides are available at the following URL:

http://www.cisco.com/c/en/us/support/switches/nexus-6000-series-switches/products-installation-guides-list.html

The document in this category include:

- *Cisco Nexus 6000 Series NX-OS Software Upgrade and Downgrade Guides*

Licensing Guide

The *License and Copyright Information for Cisco NX-OS Software* is available at http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/license_agreement/nx-ossw_lisns.html.

Command References

These guides are available at the following URL:

http://www.cisco.com/c/en/us/support/switches/nexus-6000-series-switches/products-command-reference-list.html

The documents in this category include:

- *Cisco Nexus 6000 Series NX-OS Fabric Extender Command Reference*
- *Cisco Nexus 6000 Series NX-OS FabricPath Command Reference*
- *Cisco Nexus 6000 Series NX-OS Fundamentals Command Reference*
- *Cisco Nexus 6000 Series NX-OS Interfaces Command Reference*
- *Cisco Nexus 6000 Series NX-OS Layer 2 Interfaces Command Reference*
- *Cisco Nexus 6000 Series NX-OS Multicast Routing Command Reference*
- *Cisco Nexus 6000 Series NX-OS Quality of Service Command Reference*
- *Cisco Nexus 6000 Series NX-OS Security Command Reference*
- *Cisco Nexus 6000 Series NX-OS System Management Command Reference*
- *Cisco Nexus 6000 Series NX-OS TrustSec Command Reference*
- *Cisco Nexus 6000 Series NX-OS Unicast Routing Command Reference*
- *Cisco Nexus 6000 Series NX-OS Virtual Port Channel Command Reference*

Technical References

The *Cisco Nexus 6000 Series NX-OS MIB Reference* is available at http://www.cisco.com/en/US/docs/switches/datacenter/nexus6000/sw/mib/reference/NX6000_MIBRef.html.

Error and System Messages

The *Cisco Nexus 6000 Series NX-OS System Message Guide* is available at http://www.cisco.com/c/en/us/td/docs/switches/datacenter/nexus6000/sw/system_messages/reference/sl_nxos_book.html.

Troubleshooting Guide

The *Cisco Nexus 6000 Series NX-OS Troubleshooting Guide* is available at http://www.cisco.com/c/en/us/support/switches/nexus-6000-series-switches/tsd-products-support-troubleshoot-and-alerts.html.

Related Documentation for Cisco Intercloud Fabric

This section lists the documents used with Cisco Intercloud Fabric and available at the following URL:

http://www.cisco.com/c/en/us/support/cloud-systems-management/intercloud-fabric/tsd-products-support-series-home.html

General Information

Cisco Intercloud Fabric Release Notes

Install and Upgrade

Cisco Intercloud Fabric Getting Started Guide

User Guides

Cisco Intercloud Fabric User Guide

Configuration Guides

Cisco Intercloud Fabric Configuration Guide

Cisco Intercloud Fabric Firewall Configuration Guide

Cisco vPath and vServices Reference Guide for Intercloud Fabric

Programming Guide

Cisco Intercloud Fabric Director REST API Guide

Troubleshooting and Alerts

Cisco Intercloud Fabric Troubleshooting Guide

Cisco Intercloud Fabric Provider Platform

The documentation listed below is available for use with Cisco Intercloud Fabric Provider Platform at the following URL:

http://www.cisco.com/c/en/us/support/cloud-systems-management/intercloud-fabric/tsd-products-support-series-home.html

Cisco Intercloud Fabric Provider Platform Release Notes

Cisco Intercloud Fabric Provider Platform Installation Guide

Cisco Intercloud Fabric Provider Platform Administrator Guide

Cisco Intercloud Fabric Provider Platform Troubleshooting Guide

Cisco Nexus 1000V Documentation

Cisco Nexus 1000V for VMware vSphere

Cisco Nexus 1000V for KVM

Cisco Nexus 1000V for Microsoft Hyper-V

Cisco Virtual Security Gateway Documentation

Cisco Virtual Security Gateway

Cisco Prime Network Services Controller Documentation

Cisco Prime Network Services Controller

Cisco Cloud Services Router Documentation

Cisco Cloud Services Router 1000V

Documentation Feedback

To provide technical feedback on this document, or to report an error or omission, please send your comments to: intercloud-fabric-doc-feedback@cisco.com.

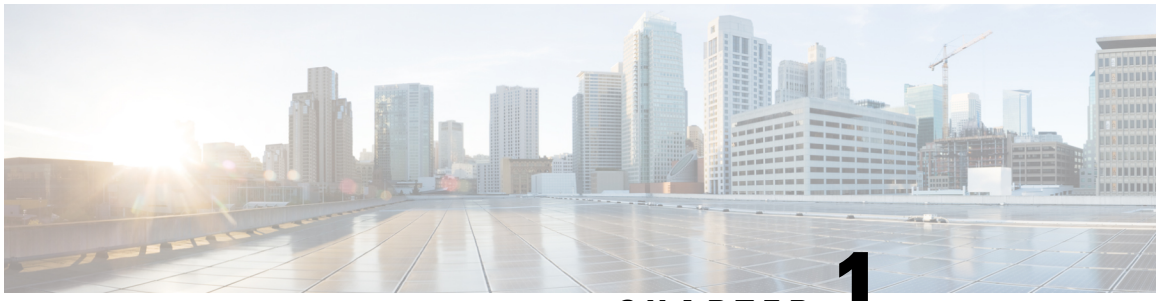
We appreciate your feedback.

Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco Marketplace.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.



CHAPTER 1

New and Changed Information

New and Changed Information

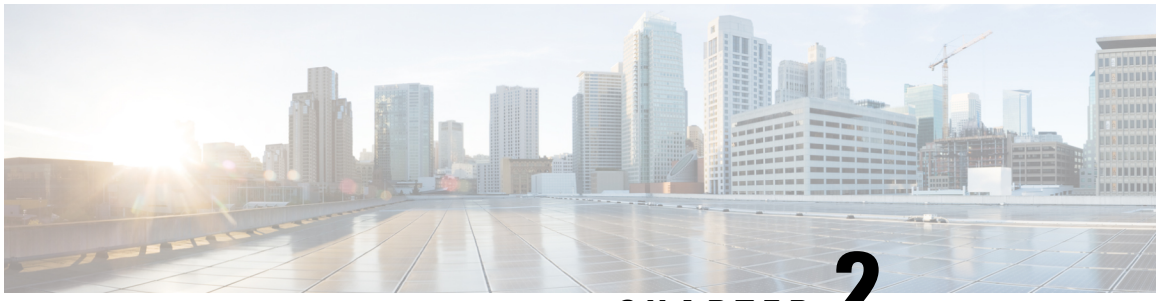
The following table provides an overview of the significant changes made to this configuration guide. The table does not provide an exhaustive list of all changes made to this guide or all new features in a particular release.

Feature	Description	Release	Where Documented
Secure Erase	Secure Erase is an operation to remove all the identifiable customer information on Cisco NX-OS devices in conditions of product removal due to Return Merchandise Authorization (RMA), or upgrade or replacement, or system end-of-life.	7.3(11)N1(1)	Configuring Secure Erase
Soft Reload	The Soft Reload feature provides a best effort mechanism for the switch to be gracefully brought up with minimal impact to production traffic when a process crash occurs. You can also use the **soft-reload** command to trigger a manual soft reload of the switch.	7.3(2)N1(1)	Soft Reload

Feature	Description	Release	Where Documented
GIR Enhancement	Starting with Cisco NX-OS Release 7.3(0)N1(1), the default mode for GIR is “isolate”. Provides support for Unplanned Maintenance, Maintenance Mode timer, Supress FIB Pending, Adding Show commands to snapshots and dumping snapshot sections. You can use GIR to perform maintenance and software upgrade of the switches and the connected FEXs. A FEX group is added to optimize the procedure to bring up or take down the FEX.	7.3(0)N1(1)	Configuring GIR
Class-based Quality-of-Service MIB Phase 2	Starting with Cisco NX-OS Release 7.3(0)N1(1), the following cbQoSMIB tables are also supported by QoS policies: cbQosClassMapStats, cbQosMatchStmtStats and cbQosQueueingStats	7.3(0)N1(1)	Class-based Quality-of-Service MIB
Performing Software Maintenance Upgrades	A software maintenance upgrade (SMU) is a package file that contains fixes for specific defects. SMUs are created to respond to immediate issues and do not include new features.	7.2(1)N1(1)	Performing Software Maintenance Upgrades
Class-based Quality-of-Service MIB	Provides the Simple Network Management Protocol (SNMP) MIB that enables retrieval of class-map and policy-map configuration and statistics.	7.1(1)N1(1)	Class-based Quality-of-Service MIB
Isolate and Maintenance Mode Enhancement	Provides the ability to gracefully eject a switch and isolate it from the network so that debugging or an upgrade can be performed. The switch is removed from the regular switching path and put into a maintenance mode. Once maintenance on the switch is complete, you can bring the switch into full operational mode.	7.1(0)N1(1)	Configuring GIR
OpenFlow	OpenFlow allows a controller to direct the forwarding functions of a switch through a secure channel.	7.0(0)N1(1)	OpenFlow

Feature	Description	Release	Where Documented
SPAN with ACL	The SPAN with ACL filtering feature allows you to filter SPAN traffic so that you can reduce bandwidth congestion.	7.0(0)N1(1)	SPAN with ACL

New and Changed Information



CHAPTER 2

Overview

This chapter contains the following sections:

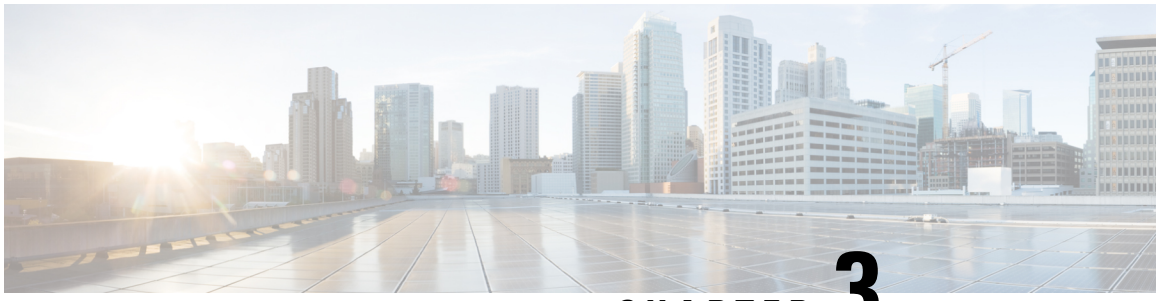
System Management Features

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

Feature	Description
Active Buffer Monitoring	The Active Buffer Monitoring feature provides detailed buffer occupancy data to help you detect network congestion, review past events to understand when and how network congestion is affecting network operations, understand historical trending, and identify patterns of application traffic flow.
Warp Mode	In warp mode, the access path is shortened by consolidating the forwarding table into single table, resulting in faster processing of frames and packets. In warp mode, latency is reduced by up to 20 percent.
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.

Feature	Description
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.
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.

Feature	Description
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 or other Remote Monitoring (RMON) probes.



CHAPTER 3

Configuring Switch Profiles

This chapter contains the following sections:

Information About Switch Profiles

Several applications require consistent configuration across Cisco Nexus Series switches in the network. Mismatched configurations can cause errors or misconfigurations that can result in service disruptions.

The configuration synchronization (config-sync) feature allows you to configure one switch profile and have the configuration be automatically synchronized to the peer switch. A switch profile provides the following benefits:

- Allows configurations to be synchronized between switches.
- Merges configurations when connectivity is established between two switches.
- Provides control of exactly which configuration gets synchronized.

- Ensures configuration consistency across peers through merge and mutual-exclusion checks.
- Provides verify and commit semantics.

Switch Profile Configuration Modes

The switch profile feature includes the following configuration modes:

- Configuration Synchronization Mode
- Switch Profile Mode
- Switch Profile Import Mode

Configuration Synchronization Mode

The configuration synchronization mode (config-sync) allows you to create switch profiles using the **config sync** command on the local switch that you want to use as the master. After you create the profile, you can enter the **config sync** command on the peer switch that you want to synchronize.

Switch Profile Mode

The switch profile mode allows you to add supported configuration commands to a switch profile that is later synchronized with a peer switch. Commands that you enter in the switch profile mode are buffered until you enter the **commit** command.

Switch Profile Import Mode

When you upgrade from an earlier release, you have the option to enter the **import** command to copy supported running-configuration commands to a switch profile. After entering the **import** command, the switch profile mode (config-sync-sp) changes to the switch profile import mode (config-sync-sp-import). The switch profile import mode allows you to import existing switch configurations from the running configuration and specify which commands you want to include in the switch profile.

Because different topologies require different commands that are included in a switch profile, the **import** command mode allows you to modify the imported set of commands to suit a specific topology.

You need to enter the **commit** command to complete the import process and move the configuration into the switch profile. Because configuration changes are not supported during the import process, if you added new commands before entering the **commit** command, the switch profile remains unsaved and the switch remains in the switch profile import mode. You can remove the added commands or abort the import. Unsaved configurations are lost if the process is aborted. You can add new commands to the switch profile after the import is complete.

Configuration Validation

Two types of configuration validation checks can identify two types of switch profile failures:

- Mutual Exclusion Checks
- Merge Checks

Mutual Exclusion Checks

To reduce the possibility of overriding configuration settings that are included in a switch profile, mutual exclusion (mutex) checks the switch profile commands against the commands that exist on the local switch and the commands on the peer switch. A command that is included in a switch profile cannot be configured outside of the switch profile or on a peer switch. This requirement reduces the possibility that an existing command is unintentionally overwritten.

As a part of the commit process, the mutex-check occurs on both switches if the peer switch is reachable; otherwise, the mutex-check is performed locally. Configuration changes made from the configuration terminal occur only on the local switch.

If a mutex-check identifies errors, they are reported as mutex failures and they must be manually corrected.

The following exceptions apply to the mutual exclusion policy:

- Interface configuration—Port channel interfaces must be configured fully in either switch profile mode or global configuration mode.

Note Several port channel subcommands are not configurable in switch profile mode. These commands can be configured from global configuration mode even if the port channel is created and configured in switch profile mode.

For example, the following command can only be configured in global configuration mode:

switchport private-vlan association trunk *primary-vlan secondary-vlan*

- Shutdown/no shutdown
- System QoS

Merge Checks

Merge checks are done on the peer switch that is receiving a configuration. The merge checks ensure that the received configuration does not conflict with the switch profile configuration that already exists on the receiving switch. The merge check occurs during the merge or commit process. Errors are reported as merge failures and must be manually corrected.

When one or both switches are reloaded and the configurations are synchronized for the first time, the merge check verifies that the switch profile configurations are identical on both switches. Differences in the switch profiles are reported as merge errors and must be manually corrected.

Software Upgrades and Downgrades with Switch Profiles

When you downgrade to an earlier release, you are prompted to remove an existing switch profile that is not supported on earlier releases.

When you upgrade from an earlier release, you have the option to move some of the running-configuration commands to a switch profile. The **import** command allows you to import relevant switch profile commands. An upgrade can occur if there are buffered configurations (uncommitted); however, the uncommitted configurations are lost.

When you perform an In Service Software Upgrade (ISSU) on one of the switches included in a switch profile, a configuration synchronization cannot occur because the peer is unreachable.

Prerequisites for Switch Profiles

Switch profiles have the following prerequisites:

- You must enable Cisco Fabric Series over IP (CFSoIP) distribution over mgmt0 on both switches by entering the **cfs ipv4 distribute** command.
- You must configure a switch profile with the same name on both peer switches by entering the **config sync** and **switch-profile** commands.
- Configure each switch as peer switch by entering the **sync-peers destination** command

Guidelines and Limitations for Switch Profiles

The Switch profile has the following guidelines and limitations:

- You can only enable configuration synchronization using the mgmt0 interface.
- Configuration synchronization is performed using the mgmt 0 interface and cannot be performed using a management SVI.
- You must configure synchronized peers with the same switch profile name.
- Commands that are qualified for a switch profile configuration are allowed to be configured in the configuration switch profile (config-sync-sp) mode.
- Supported switch profile commands relate to virtual port channel (vPC) commands. Fiber Channel over Ethernet (FCoE) commands are not supported.
- One switch profile session can be in progress at a time. Attempts to start another session will fail.
- Supported command changes made from the configuration terminal mode are blocked when a switch profile session is in progress. You should not make unsupported command changes from the configuration terminal mode when a switch profile session is in progress.
- When you enter the **commit** command and a peer switch is reachable, the configuration is applied to both peer switches or neither switch. If there is a commit failure, the commands remain in the switch profile buffer. You can then make necessary corrections and try the commit again.
- We recommend that you enable preprovisioning for all Generic Expansion Modules (GEMs) and Cisco Nexus Fabric Extender modules whose interface configurations are synchronized using the configuration synchronization feature. Follow these guidelines in Cisco Nexus Fabric Extender active/active topologies where the Fabric Extenders might not be online on one switch and its configuration is changed and synchronized on the other switch. In this scenario, if you do not enable preprovisioning, a commit fails and the configuration is rolled back on both switches.
- Once a port channel is configured using switch profile mode, it cannot be configured using global configuration (config terminal) mode.

Note Several port channel subcommands are not configurable in switch profile mode. These commands can be configured from global configuration mode even if the port channel is created and configured in switch profile mode.

For example, the following command can only be configured in global configuration mode:

switchport private-vlan association trunk *primary-vlan secondary-vlan*

- Shutdown and no shutdown can be configured in either global configuration mode or switch profile mode.
- If a port channel is created in global configuration mode, channel groups including member interfaces must also be created using global configuration mode.
- Port channels that are configured within switch profile mode may have members both inside and outside of a switch profile.
- If you want to import a member interface to a switch profile, the port channel including the member interface must also be present within the switch profile.

Guidelines for Synchronizing After Reboot, Connectivity Loss, or Failure

- Synchronizing configurations after vPC peer link failure— If both switches are operational when a peer link fails, the secondary switch shuts down its vPC ports. In a Fabric Extender A/A topology, the A/A Fabric Extender disconnects from the secondary switch. If the configuration is changed using a switch profile on the primary switch, configurations are not accepted on the secondary switch unless the A/A Fabric Extender is preprovisioned. When using the configuration synchronization feature, we recommend that you preprovision all A/A Fabric Extenders.
- Synchronizing configurations after mgmt0 interface connectivity loss—When mgmt0 interface connectivity is lost and configuration changes are required, apply the configuration changes on both switches using the switch profile. When connectivity to the mgmt0 interface is restored, both switches synchronize automatically.

  If a configuration change is made on only one switch, a merge occurs when the mgmt0 interface comes up and the configuration is applied on the other switch.
- Synchronizing configurations when an ISSU is performed on one switch and a configuration change is made on the peer switch—In a vPC topology, configuration changes on the peer switch are not allowed when an ISSU is performed on the other switch. In topologies without vPCs, configuration changes are allowed and the switch undergoing an ISSU synchronizes new configurations when the upgrade is complete.

Configuring Switch Profiles

You can create and configure a switch profile. Enter the **switch-profile** *name* command in the configuration synchronization mode (config-sync).

Before you begin

You must create the switch profile with the same name on each switch and the switches must configure each other as a peer. When connectivity is established between switches with the same active switch profile, the switch profiles are synchronized.

Procedure

	Command or Action	Purpose

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

Step 2	**cfs ipv4 distribute**
	Example:
	`switch(config)# cfs ipv4 distribute`
	`switch(config)#`
		Enables CFS distribution between the peer switches.

Step 3	**config sync**
	Example:
	`switch# config sync`
	`switch(config-sync)#`
		Enters configuration synchronization mode.

Step 4	**switch-profile** *name*
	Example:
	`switch(config-sync)# switch-profile abc`
	`switch(config-sync-sp)#`
		Configures the switch profile, names the switch profile, and enters switch profile synchronization configuration mode.

Step 5	**sync-peers destination** *IP-address*
	Example:
	`switch(config-sync-sp)# sync-peers destination 10.1.1.1`
	`switch(config-sync-sp)#`
		Configures the peer switch.

Step 6	(Optional) **show switch-profile** *name* **status**
	Example:
	`switch(config-sync-sp)# show switch-profile abc status`
	`switch(config-sync-sp)#`
		Views the switch profile on the local switch and the peer switch information.

Step 7	**exit**
	Example:
	`switch(config-sync-sp)# exit`
	`switch#`
		Exits the switch profile configuration mode and returns to EXEC mode.

Step 8	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

The following example shows how to configure a switch profile and shows the switch profile status.

```
switch# configuration terminal
switch(config)# cfs ipv4 distribute
switch(config-sync)# switch-profile abc
switch(config-sync-sp)# sync-peers destination 10.1.1.1
switch(config-sync-sp)# show switch-profile abc status
Start-time:  15801 usecs after Mon Aug 23 06:21:08 2010
End-time:   6480 usecs after Mon Aug 23 06:21:13 2010

Profile-Revision: 1
Session-type: Initial-Exchange
Peer-triggered: Yes
Profile-status: Sync Success

Local information:
----------------
Status: Commit Success
Error(s):

Peer information:
----------------
IP-address: 10.1.1.1
Sync-status: In Sync.
Status: Commit Success
Error(s):
switch(config-sync-sp)# exit
switch#
```

Adding a Switch to a Switch Profile

Enter the **sync-peers destination** *destination IP* command in switch profile configuration mode to add the switch to a switch profile.

Follow these guidelines when adding switches:

- Switches are identified by their IP address.
- Destination IPs are the IP addresses of the switches that you want to synchronize.
- The committed switch profile is synchronized with the newly added peers (when they are online) if the peer switch is also configured with configuration synchronization.

  If you want to import a member interface to a switch profile, the port channel including the member interface must also be present within the switch profile.

Before you begin

After creating a switch profile on the local switch, you must add the second switch that will be included in the synchronization.

Procedure

	Command or Action	Purpose

Step 1	**config sync**
	Example:
	`switch# config sync`
	`switch(config-sync)#`
		Enters configuration synchronization mode.

Step 2	**switch-profile** *name*
	Example:
	`switch(config-sync)# switch-profile abc`
	`switch(config-sync-sp)#`
		Configures switch profile, names the switch profile, and enters switch profile synchronization configuration mode.

Step 3	**sync-peers destination** *destination IP*
	Example:
	`switch(config-sync-sp)# sync-peers destination 10.1.1.1`
	`switch(config-sync-sp)#`
		Adds a switch to the switch profile.

Step 4	**exit**
	Example:
	`switch(config-sync-sp)# exit`
	`switch#`
		Exits switch profile configuration mode.

Step 5	(Optional) **show switch-profile peer** **Example:** `switch# show switch-profile peer`	Displays the switch profile peer configuration.
Step 6	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Adding or Modifying Switch Profile Commands

To modify a command in a switch profile, add the modified command to the switch profile and enter the **commit** command to apply the command and synchronize the switch profile to the peer switch if it is reachable.

Follow these guidelines when adding or modifying switch profile commands:

- Commands that are added or modified are buffered until you enter the **commit** command.
- Commands are executed in the same order in which they are buffered. If there is an order-dependency for certain commands, for example, a QoS policy must be defined before being applied, you must maintain that order; otherwise, the commit might fail. You can use utility commands, such as the **show switch-profile name buffer** command, the **buffer-delete** command, or the **buffer-move** command, to change the buffer and correct the order of already entered commands.

Before you begin

After configuring a switch profile on the local and the peer switch, you must add and commit the supported commands to the switch profile. The commands are added to the switch profile buffer until you enter the **commit** command. The **commit** command does the following:

- Triggers the mutex check and the merge check to verify the synchronization.
- Creates a checkpoint with a rollback infrastructure.
- Applies the configuration on the local switch and the peer switch.
- Executes a rollback on all switches if there is a failure with an application on any of the switches in the switch profile.
- Deletes the checkpoint.

Procedure

	Command or Action	Purpose

Step 1	**config sync**
	Example:
	`switch# config sync`
	`switch(config-sync)#`
		Enters configuration synchronization mode.

Step 2	**switch-profile** *name*
	Example:
	`switch(config-sync)# switch-profile abc`
	`switch(config-sync-sp)#`
		Configures the switch profile, names the switch profile, and enters switch profile synchronization configuration mode.

Step 3	*Command argument*
	Example:
	`switch(config-sync-sp)# interface Port-channel100`
	`switch(config-sync-sp-if)# speed 1000`
	`switch(config-sync-sp-if)# interface Ethernet1/1`
	`switch(config-sync-sp-if)# speed 1000`
	`switch(config-sync-sp-if)# channel-group 100`
		Adds a command to the switch profile.

Step 4	(Optional) **show switch-profile** *name* **buffer**
	Example:
	`switch(config-sync-sp)# show switch-profile abc buffer`
	`switch(config-sync-sp)#`
		Displays the configuration commands in the switch profile buffer.

Step 5	**verify** **Example:** `switch(config-sync-sp)# verify`	Verifies the commands in the switch profile buffer.

	Command or Action	Purpose
Step 6	**commit** **Example:** `switch(config-sync-sp)# commit`	Saves the commands in the switch profile and synchronizes the configuration with the peer switch.

Step 7	(Optional) **show switch-profile** *name* **status**
	Example:
	`switch(config-sync-sp)# show switch-profile abc status`
	`switch(config-sync-sp)#`
		Displays the status of the switch profile on the local switch and the status on the peer switch.

Step 8	**exit**
	Example:
	`switch(config-sync-sp)# exit`
	`switch#`
		Exits the switch profile configuration mode.

Step 9	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

The following example shows how to create a switch profile, configure a peer switch, and add commands to the switch profile.

```
switch# configuration terminal
switch(config)# cfs ipv4 distribute
switch(config-sync)# switch-profile abc
switch(config-sync-sp)# sync-peers destination 10.1.1.1
switch(config-sync-sp)# interface port-channel100
switch(config-sync-sp-if)# speed 1000
switch(config-sync-sp-if)# interface Ethernet1/1
switch(config-sync-sp-if)# speed 1000
switch(config-sync-sp-if)# channel-group 100
switch(config-sync-sp)# verify
switch(config-sync-sp)# commit
switch(config-sync-sp)# exit
switch#
```

The following example shows an existing configuration with a defined switch profile. The second example shows how the switch profile command changed by adding the modified command to the switch profile.

```
switch# show running-config
switch-profile abc
  interface Ethernet1/1
    switchport mode trunk
    switchport trunk allowed vlan 1-10

switch# config sync
switch(config-sync)# switch-profile abc
switch(config-sync-sp)# interface Ethernet1/1
switch(config-sync-sp-if)# switchport trunk allowed vlan 5-10
```

```
switch(config-sync-sp-if)# commit

switch# show running-config
switch-profile abc
  interface Ethernet1/1
    switchport mode trunk
    switchport trunk allowed vlan 5-10
```

Importing a Switch Profile

You can import a switch profile based on the set of commands that you want to import. Using the configuration terminal mode, you can do the following:

- Add selected commands to the switch profile.
- Add supported commands that were specified for an interface.
- Add supported system-level commands.
- Add supported system-level commands excluding the physical interface commands.

When you import commands to a switch profile, the switch profile buffer must be empty.

If new commands are added during the import, the switch profile remains unsaved and the switch remains in the switch profile import mode. You can enter the **abort** command to stop the import. For additional information importing a switch profile, see the "Switch Profile Import Mode" section.

Procedure

	Command or Action	Purpose

Step 1	**config sync**
	Example:
	`switch# config sync`
	`switch(config-sync)#`
		Enters configuration synchronization mode.

Step 2	**switch-profile** *name*
	Example:
	`switch(config-sync)# switch-profile abc`
	`switch(config-sync-sp)#`
		Configures the switch profile, names the switch profile, and enters switch profile synchronization configuration mode.

Step 3	**import** {*interface port/slot* | *running-config* [**exclude interface ethernet**]}
	Example:
	`switch(config-sync-sp)# import ethernet 1/2`
	`switch(config-sync-sp-import)#`
		Identifies the commands that you want to import and enters switch profile import mode.
		• <CR>—Adds selected commands.
		• **interface**—Adds the supported commands for a specified interface.
		• **running-config**—Adds supported system-level commands.
		• **running-config exclude interface ethernet**—Adds supported system-level

	Command or Action	Purpose
		commands excluding the physical interface commands. **Note** If this is a QSFP+GEM or a breakout port, the *port* syntax is *QSFP-module*/*port*.
Step 4	**commit** **Example:** `switch(config-sync-sp-import)# commit`	Imports the commands and saves the commands to the switch profile.
Step 5	(Optional) **abort** **Example:** `switch(config-sync-sp-import)# abort`	Aborts the import process.

Step 6	**exit**
	Example:
	`switch(config-sync-sp)# exit`
	`switch#`
		Exits switch profile import mode.

Step 7	(Optional) **show switch-profile** **Example:** `switch# show switch-profile`	Displays the switch profile configuration.
Step 8	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

The following example shows how to import supported system-level commands excluding the Ethernet interface commands into the switch profile named sp:

```
switch(config-vlan)# conf sync
switch(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch(config-sync-sp)# show switch-profile buffer

switch-profile  : sp
----------------------------------------------------------
Seq-no  Command
----------------------------------------------------------

switch(config-sync-sp)# import running-config exclude interface ethernet
switch(config-sync-sp-import)#
switch(config-sync-sp-import)# show switch-profile buffer

switch-profile  : sp
----------------------------------------------------------
```

```
Seq-no  Command
-------------------------------------------------------------
3       vlan 100-299
4       vlan 300
4.1       state suspend
5       vlan 301-345
6       interface port-channel100
6.1       spanning-tree port type network
7       interface port-channel105

switch(config-sync-sp-import)#
```

Importing Configurations in a vPC Topology

You can import configurations in a two-switch vPC topology.

Note For specific information about the following steps, see the appropriate sections in this chapter.

- Configure the switch profile with the same name on both switches.
- Import the configurations to both switches independently.

Note Ensure that the configuration moved to the switch profile on both switches is identical; otherwise, a merge-check failure might occur.

- Configure the switches by entering the **sync-peer destination** command.
- Verify that the switch profiles are the same by entering the appropriate show commands.

Verifying Commands in a Switch Profile

You can verify the commands that are included in a switch profile by entering the **verify** command in switch profile mode.

Procedure

	Command or Action	Purpose

Step 1	**config sync**
	Example:
	`switch# config sync`
	`switch(config-sync)#`
		Enters configuration synchronization mode.

	Command or Action	Purpose

Step 2	**switch-profile** *name*
	Example:
	`switch(config-sync)# switch-profile abc`
	`switch(config-sync-sp)#`
		Configures the switch profile, names the switch profile, and enters switch profile synchronization configuration mode.

Step 3	**verify** **Example:** `switch(config-sync-sp)# verify`	Verifies the commands in the switch profile buffer.

Step 4	**exit**
	Example:
	`switch(config-sync-sp)# exit`
	`switch#`
		Exits the switch profile configuration mode.

Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Isolating a Peer Switch

You can isolate a peer switch in order to make changes to a switch profile. This process can be used when you want to block a configuration synchronization or when you want to debug configurations.

Isolating a peer switch requires that you remove the switch from the switch profile and then add the peer switch back to the switch profile.

To temporarily isolate a peer switch, follow these steps:

1. Remove a peer switch from a switch profile.
2. Make changes to the switch profile and commit the changes.
3. Enter debug commands.
4. Undo the changes that were made to the switch profile in Step 2 and commit.
5. Add the peer switch back to the switch profile.

Deleting a Switch Profile

You can delete a switch profile by selecting the **all-config** or the **local-config** option:

- **all-config**—Deletes the switch profile on both peer switches (when both are reachable). If you choose this option and one of the peers is unreachable, only the local switch profile is deleted. The **all-config** option completely deletes the switch profile on both peer switches.
- **local-config**—Deletes the switch profile on the local switch only.

Procedure

	Command or Action	Purpose

Step 1	**config sync**
	Example:
	`switch# config sync`
	`switch(config-sync)#`
		Enters configuration synchronization mode.

Step 2	**no switch-profile name** {**all-config** | **local-config**}
	Example:
	`switch(config-sync)# no switch-profile abc local-config`
	`switch(config-sync-sp)#`
		Deletes the switch profile as follows:
		• **all-config**—Deletes the switch profile on the local and peer switch. If the peer switch is not reachable, only the local switch profile is deleted.
		• **local-config**—Deletes the switch profile and local configuration.

Step 3	**exit**
	Example:
	`switch(config-sync-sp)# exit`
	`switch#`
		Exits configuration synchronization mode.

Step 4	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Deleting a Switch from a Switch Profile

You can delete a switch from a switch profile.

Procedure

	Command or Action	Purpose

Step 1	**config sync**
	Example:
	`switch# config sync`
	`switch(config-sync)#`
		Enters configuration synchronization mode.

Step 2	**switch-profile** *name*
	Example:
	`switch(config-sync)# switch-profile abc`
	`switch(config-sync-sp)#`
		Configures the switch profile, names the switch profile, and enters the switch profile synchronization configuration mode.

Step 3	**no sync-peers destination** *destination IP* **Example:**	Removes the specified switch from the switch profile.

	Command or Action	Purpose
	`switch(config-sync-sp)# no sync-peers destination 10.1.1.1` `switch(config-sync-sp)#`	

Step 4	**exit**
	Example:
	`switch(config-sync-sp)# exit`
	`switch#`
		Exits the switch profile configuration mode.

Step 5	(Optional) **show switch-profile** **Example:** `switch# show switch-profile`	Displays the switch profile configuration.
Step 6	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Displaying the Switch Profile Buffer

Procedure

	Command or Action	Purpose
Step 1	switch# **configure sync**	Enters configuration synchronization mode.
Step 2	switch(config-sync) # **switch-profile** *profile-name*	Enters switch profile synchronization configuration mode for the specified switch profile.
Step 3	switch(config-sync-sp) # **show switch-profile***profile-name* **buffer**	Enters interface switch profile synchronization configuration mode for the specified interface.

Example

The following example shows how to display the switch profile buffer for a service profile called sp:

```
switch# configure sync
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch(config-sync-sp)# show switch-profile sp buffer
------------------------------------------------------------
Seq-no  Command
------------------------------------------------------------
1       vlan 101
1.1       ip igmp snooping querier 10.101.1.1
2       mac address-table static 0000.0000.0001 vlan 101 drop
```

```
3       interface Ethernet1/2
3.1       switchport mode trunk
3.2       switchport trunk allowed vlan 101

switch(config-sync-sp)# buffer-move 3 1
switch(config-sync-sp)# show switch-profile sp buffer
----------------------------------------------------------
Seq-no  Command
----------------------------------------------------------
1       interface Ethernet1/2
1.1       switchport mode trunk
1.2       switchport trunk allowed vlan 101
2       vlan 101
2.1       ip igmp snooping querier 10.101.1.1
3       mac address-table static 0000.0000.0001 vlan 101 drop
switch(config-sync-sp)#
```

Synchronizing Configurations After a Switch Reboot

If a Cisco Nexus Series switch reboots while a new configuration is being committed on a peer switch using a switch profile, complete the following steps to synchronize the peer switches after reload:

Procedure

Step 1 Reapply configurations that were changed on the peer switch during the reboot.

Step 2 Enter the **commit** command.

Step 3 Verify that the configuration is applied correctly and both peers are back synchronized.

Example

Switch Profile Configuration show Commands

The following **show** commands display information about the switch profile.

Command	Purpose
show switch-profile *name*	Displays the commands in a switch profile.
show switch-profile *name* **buffer**	Displays the uncommitted commands in a switch profile, the commands that were moved, and the commands that were deleted.
show switch-profile *name* **peer** *IP-address*	Displays the synchronization status for a peer switch.
show switch-profile *name* **session-history**	Displays the status of the last 20 switch profile sessions.
show switch-profile *name* **status**	Displays the configuration synchronization status of a peer switch.

Command	Purpose
show running-config exclude-provision	Displays the configurations for offline preprovisioned interfaces that are hidden.
show running-config switch-profile	Displays the running configuration for the switch profile on the local switch.
show startup-config switch-profile	Displays the startup configuration for the switch profile on the local switch.

For detailed information about the fields in the output from these commands, see the system management command reference for your platform.

Configuration Examples for Switch Profiles

Creating a Switch Profile on a Local and Peer Switch Example

The following example shows how to create a successful switch profile configuration on a local and peer switch.

Procedure

	Command or Action	Purpose

Step 1	Enable CFSoIP distribution on the local and the peer switch.
	Example:
	`switch# configuration terminal`
	`switch(config)# cfs ipv4 distribute`	

Step 2	Create a switch profile on the local and the peer switch.
	Example:
	`switch(config-sync)# switch-profile abc`
	
	`switch(config-sync-sp)# sync-peers destination 10.1.1.1`	

Step 3	Verify that the switch profiles are the same on the local and the peer switch.
	Example:
	`switch(config-sync-sp)# show switch-profile abc status`
	
	`Start-time: 15801 usecs after Mon Aug 23 06:21:08 2010`
	`End-time: 6480 usecs after Mon Aug 23 06:21:13 2010`
	
	`Profile-Revision: 1`	

	Command or Action	Purpose
	Session-type: Initial-Exchange Peer-triggered: Yes Profile-status: Sync Success Local information: ---------- Status: Commit Success Error(s): Peer information: ---------- IP-address: 10.1.1.1 Sync-status: In Sync. Status: Commit Success Error(s):	
Step 4	Add the configuration commands to the switch profile on the local switch. The commands will be applied to the peer switch when the commands are committed. **Example:** switch(config-sync-sp)# **class-map type qos c1**	
Step 5	Verify the commands in the switch profile. **Example:** switch(config-sync-sp-if)# **verify** Verification Successful	
Step 6	Apply the commands to the switch profile and to synchronize the configurations between the local and the peer switch. **Example:** switch(config-sync-sp)# **commit** Commit Successful switch(config-sync)#	

Verifying the Synchronization Status Example

The following example shows how to verify the synchronization status between the local and the peer switch:

```
switch(config-sync)# show switch-profile switch-profile status
Start-time: 804935 usecs after Mon Aug 23 06:41:10 2010
End-time: 956631 usecs after Mon Aug 23 06:41:20 2010

Profile-Revision: 2
Session-type: Commit
Peer-triggered: No
Profile-status: Sync Success

Local information:
----------
Status: Commit Success
```

```
Error(s):

Peer information:
----------------
IP-address: 10.1.1.1
Sync-status: In Sync.
Status: Commit Success
Error(s):

switch(config-sync)#
```

Displaying the Running Configuration

The following example shows how to display the running configuration of the switch profile on the local switch:

```
switch# configure sync
switch(config-sync)# show running-config switch-profile

switch(config-sync)#
```

Displaying the Switch Profile Synchronization Between Local and Peer Switches

This example shows how to display the synchronization status for two peer switches:

```
switch1# show switch-profile sp status

Start-time: 491815 usecs after Thu Aug 12 11:54:51 2010
End-time: 449475 usecs after Thu Aug 12 11:54:58 2010

Profile-Revision: 1
Session-type: Initial-Exchange
Peer-triggered: No
Profile-status: Sync Success

Local information:
----------------
Status: Commit Success
Error(s):

Peer information:
----------------
IP-address: 10.193.194.52
Sync-status: In Sync.
Status: Commit Success
Error(s):

switch1#


switch2# show switch-profile sp status

Start-time: 503194 usecs after Thu Aug 12 11:54:51 2010
End-time: 532989 usecs after Thu Aug 12 11:54:58 2010

Profile-Revision: 1
Session-type: Initial-Exchange
Peer-triggered: Yes
Profile-status: Sync Success
```

```
Local information:
----------------
Status: Commit Success
Error(s):

Peer information:
----------------
IP-address: 10.193.194.51
Sync-status: In Sync.
Status: Commit Success
Error(s):

switch2#
```

Displaying Verify and Commit on Local and Peer Switches

This example shows how to configure a successful verify and commit of the local and peer switch:

```
switch1# configure sync
Enter configuration commands, one per line.  End with CNTL/Z.
switch1(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch1(config-sync-sp)# interface ethernet1/1
switch1(config-sync-sp-if)# description foo
switch1(config-sync-sp-if)# verify
Verification Successful
switch1(config-sync-sp)# commit
Commit Successful
switch1(config-sync)# show running-config switch-profile
switch-profile sp
  sync-peers destination 10.193.194.52
  interface Ethernet1/1
    description foo
switch1(config-sync)# show switch-profile sp status

Start-time: 171513 usecs after Wed Aug 11 17:51:28 2010
End-time: 676451 usecs after Wed Aug 11 17:51:43 2010

Profile-Revision: 3
Session-type: Commit
Peer-triggered: No
Profile-status: Sync Success

Local information:
----------------
Status: Commit Success
Error(s):

Peer information:
----------------
IP-address: 10.193.194.52
Sync-status: In Sync.
Status: Commit Success
Error(s):

switch1(config-sync)#


switch2# show running-config switch-profile
switch-profile sp
  sync-peers destination 10.193.194.51
  interface Ethernet1/1
```

```
    description foo
switch2# show switch-profile sp status

Start-time: 265716 usecs after Wed Aug 11 16:51:28 2010
End-time: 734702 usecs after Wed Aug 11 16:51:43 2010

Profile-Revision: 3
Session-type: Commit
Peer-triggered: Yes
Profile-status: Sync Success

Local information:
----------------
Status: Commit Success
Error(s):

Peer information:
----------------
IP-address: 10.193.194.51
Sync-status: In Sync.
Status: Commit Success
Error(s):

switch2#
```

Successful and Unsuccessful Synchronization Examples

The following example shows a successful synchronization of the switch profile on the peer switch:

```
switch# show switch-profile abc peer

switch# show switch-profile sp peer 10.193.194.52
Peer-sync-status          : In Sync.
Peer-status               : Commit Success
Peer-error(s)             :
switch1#
```

The following example shows an unsuccessful synchronization of a switch profile on the peer switch, with a peer not reachable status:

```
switch# show switch-profile sp peer 10.193.194.52
Peer-sync-status          : Not yet merged. pending-merge:1 received_merge:0
Peer-status               : Peer not reachable
Peer-error(s)             :
switch#
```

Configuring the Switch Profile Buffer, Moving the Buffer, and Deleting the Buffer

This example shows how to configure the switch profile buffer, the buffer-move configuration, and the buffer-delete configuration:

```
switch# configure sync
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch(config-sync-sp)# vlan 101
switch(config-sync-sp-vlan)# ip igmp snooping querier 10.101.1.1
switch(config-sync-sp-vlan)# exit
switch(config-sync-sp)# mac address-table static 0000.0000.0001 vlan 101 drop
```

```
switch(config-sync-sp)# interface ethernet1/2
switch(config-sync-sp-if)# switchport mode trunk
switch(config-sync-sp-if)# switchport trunk allowed vlan 101
switch(config-sync-sp-if)# exit
switch(config-sync-sp)# show switch-profile sp buffer
-----------------------------------------------------------
Seq-no  Command
-----------------------------------------------------------
1       vlan 101
1.1       ip igmp snooping querier 10.101.1.1
2       mac address-table static 0000.0000.0001 vlan 101 drop
3       interface Ethernet1/2
3.1       switchport mode trunk
3.2       switchport trunk allowed vlan 101

switch(config-sync-sp)# buffer-move 3 1
switch(config-sync-sp)# show switch-profile sp buffer
-----------------------------------------------------------
Seq-no  Command
-----------------------------------------------------------
1       interface Ethernet1/2
1.1       switchport mode trunk
1.2       switchport trunk allowed vlan 101
2       vlan 101
2.1       ip igmp snooping querier 10.101.1.1
3       mac address-table static 0000.0000.0001 vlan 101 drop

switch(config-sync-sp)# buffer-delete 1
switch(config-sync-sp)# show switch-profile sp buffer
-----------------------------------------------------------
Seq-no  Command
-----------------------------------------------------------
2       vlan 101
2.1       ip igmp snooping querier 10.101.1.1
3       mac address-table static 0000.0000.0001 vlan 101 drop

switch(config-sync-sp)# buffer-delete all
switch(config-sync-sp)# show switch-profile sp buffer
switch(config-sync-sp)#
```

Sample Migrations Using the Import Command

Migrating Cisco NX-OS Release 5.0(2)N1(1) in a Fabric Extender A-A Topology Example

This examples shows the tasks used to migrate to Cisco NX-OS Release 5.0(2)N1(1) in a Fabric Extender A-A topology. For details on the tasks, see the appropriate sections in this chapter.

Procedure

Step 1 Ensure configurations are the same on both switches.

Step 2 Configure the switch-profile with same name on both switches.

Step 3 Enter the **import running config** command on both switches.

Step 4 Enter the **switch-profile** *name* **buffer** command to ensure all configurations are correctly imported on both switches.

Step 5 Remove unwanted configuration settings by editing the buffer.

For details, see Configuring the Switch Profile Buffer, Moving the Buffer, and Deleting the Buffer, on page 30.

Step 6 Enter the **commit** command on both switches.

Step 7 Enter the **sync-peers destination** *IP-address* command to configure the peer switch on both switches.

Step 8 Enter the **switch-profile** *name* **status** command to ensure both switches are synchronized.

Migrating Cisco NX-OS Release 5.0(2)N1(1) in a Fabric Extender Fabric Extender Straight-Through Topology Example

This example shows the tasks used to migrate to Cisco NX-OS Release 5.0(2)N1(1) in a Fabric Extender Straight-Through topology. For details on the tasks, see the appropriate sections in this chapter.

Procedure

Step 1 Ensure the vPC port-channel configurations are the same on both switches.

Step 2 Configure the switch-profile with the same name on both switches.

Step 3 Enter the **import interface port-channel** *x-y*, **port-channel** *z* command for all vPC port-channels on both switches.

Step 4 Enter the show switch-profile *name* buffer command to ensure all configurations are correctly imported on both switches.

Step 5 Remove unwanted configuration settings by editing the buffer.

For details, see Configuring the Switch Profile Buffer, Moving the Buffer, and Deleting the Buffer, on page 30.

Step 6 Enter the **commit** command on both switches.

Step 7 Enter the **sync-peers destination** *IP-address* command to configure the peer switch on both switches.

Step 8 Enter the **show switch-profile** *name* **status** command to ensure both switches are synchronized.

Replacing a Cisco Nexus 5000 Series Switch

When a Cisco Nexus 5000 Series switch has been replaced, perform the following configuration steps on the replacement switch to synchronize it with the existing Cisco Nexus 5000 Series switch. The procedure can be done in a hybrid Fabric Extender A/A topology and Fabric Extender Straight-Through topology.

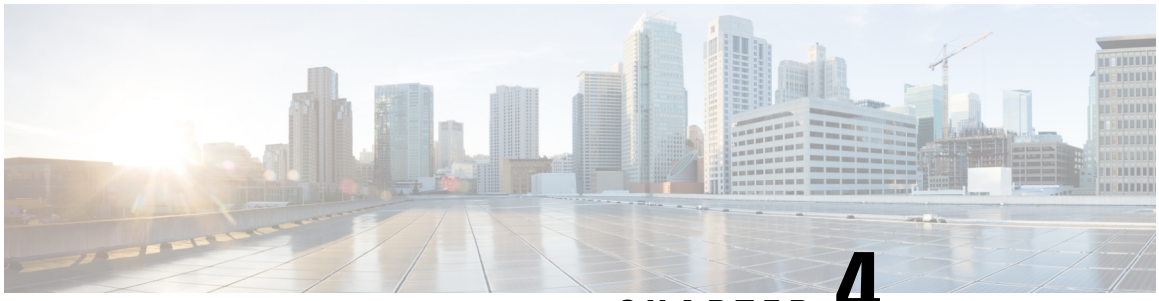
1. Do not connect any peer-link, vPC, A/A or Straight-Through topology fabric ports to the replacement switch.
2. Boot the replacement switch. The switch comes up with no configuration.
3. Enable pre-provisioning on all Fabric Extender A/A and ST modules.
4. Configure the replacement switch:

   If the running-configuration was saved offline, follow steps 5-9 to apply the configuration.

If the running-configuration was not saved offline, you can obtain it from the peer switch if the configuration synchronization feature is enabled. (See Steps 1 and 2 from "Creating a Switch Profile on a Local and Peer Switch" then begin with step 10 below).

If neither condition is met, manually add the configuration and then begin with step 10 below.

5. Edit the configuration file to remove the **sync-peer** command if using the configuration synchronization feature.
6. Configure the mgmt port IP address and download the configuration file.
7. Copy the saved configuration file to the running configuration.
8. Verify the configuration is correct by entering the **show running-config** command and the **show provision failed-config** *slot* command.
9. If switch-profile configuration changes were made on the peer switch while the replacement switch was out-of-service, apply those configurations in the switch-profile and then enter the commit command.
10. Shutdown all Fabric Extender ST topology ports that are included in a vPC topology.
11. Connect the Fabric Extender ST topology fabric ports.
12. Wait for Fabric Extender ST topology switches to come online.
13. Ensure the vPC role priority of the existing switch is better than the replacement switch.
14. Connect the peer-link ports to the peer switch.
15. Connect the Fabric Extender A/A topology fabric ports.
16. Connect the switch vPC ports.
17. Enter the **no shutdown** command on all Fabric Extender ST vPC ports.
18. Verify that all vPC switches and the Fabric Extenders on the replacement switch come online and that there is no disruption in traffic.
19. If you are using the configuration synchronization feature, add the sync-peer configuration to the switch-profile if this wasn’t enabled in Step 4.
20. If you are using the configuration synchronization feature, enter the **show switch-profile** *name* **status** command to ensure both switches are synchronized.



CHAPTER 4

Configuring Module Pre-Provisioning

This chapter contains the following sections:

Information About Module Pre-Provisioning

The pre-provisioning feature allows you to preconfigure interfaces before inserting or attaching a module. If a module goes offline, you can also use pre-provisioning to make changes to the interface configurations for the offline module. When a pre-provisioned module comes online, the pre-provisioning configurations are applied. If any configurations were not applied, a syslog is generated. The syslog lists the configurations that were not accepted.

In some Virtual Port Channel (vPC) topologies, pre-provisioning is required for the configuration synchronization feature. Pre-provisioning allows you to synchronize the configuration for an interface that is online with one peer but offline with another peer.

Guidelines and Limitations

Pre-provisioning has the following configuration guidelines and limitations:

- When a module comes online, commands that are not applied are listed in the syslog.
- If a slot is pre-provisioned for module A and if you insert module B into the slot, module B does not come online.
- There is no MIB support for pre-provisioned interfaces.
- Cisco DCNM is not supported.

Enabling Module Pre-Provisioning

You can enable pre-provisioning on a module that is offline. Enter the **provision model** *model* command in module pre-provision mode.

Note After enabling pre-provisioning, you can configure the interfaces as though they are online.

Procedure

	Command or Action	Purpose

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

Step 2	**slot** *slot*
	Example:
	`switch(config)# slot 101`
	`switch(config-slot)#`
		Selects the slot to pre-provision and enters slot configuration mode.

Step 3	**provision model** *model*
	Example:
	`switch(config-slot)# provision model N2K-C2248T`
	`switch(config-slot)#`
		Selects the module that you want to pre-provision.

Step 4	**exit**
	Example:
	`switch(config-slot)# exit`
	`switch#`
		Exits slot configuration mode.

Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to select slot 101 and the N2K-C2232P module to pre-provision.

```
switch# configure terminal
switch(config)# slot 101
switch(config-slot)# provision model N2K-C2232P
switch(config-slot)# exit
```

Removing Module Pre-Provisioning

You can remove a module that has been pre-provisioned.

Procedure

	Command or Action	Purpose

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

Step 2	**slot** *slot*
	Example:
	`switch(config)# slot 101`
	`switch(config-slot)#`
		Selects the slot to pre-provision and enters slot configuration mode.

Step 3	**no provision model** *model*
	Example:
	`switch(config-slot)# no provision model N2K-C2248T`
	`switch(config-slot)#`
		Removes pre-provisioning from the module.

Step 4	**exit**
	Example:
	`switch(config-slot)# exit`
	`switch#`
		Exits slot configuration mode.

Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

This example shows how to remove a preprovisioned module from a chassis slot:

```
switch(config)# slot 2
switch(config-slot)# no provision model N5K-M1404
switch(config-slot)#
```

Verifying the Pre-Provisioned Configuration

To display the pre-provisioned configuration, perform one of the following tasks:

Command	Purpose
show module	Displays module information.
show switch-profile	Displays switch profile information.
show running-config exclude-provision	Displays the running configuration without the pre-provisioned interfaces or modules that are offline.
show provision failed-config	Displays the pre-provisioned commands that were not applied to the configuration when the interface or module came online. This command also displays a history of failed commands.
show running-config	Displays the running configuration including the pre-provisioned configuration.
show startup-config	Displays the startup configuration including the pre-provisioned configuration.

Configuration Examples for Pre-Provisioning

The following example shows how to enable pre-provisioning on slot 110 on the Cisco Nexus 2232P Fabric Extender and how to pre-provision interface configuration commands on the Ethernet 110/1/1 interface.

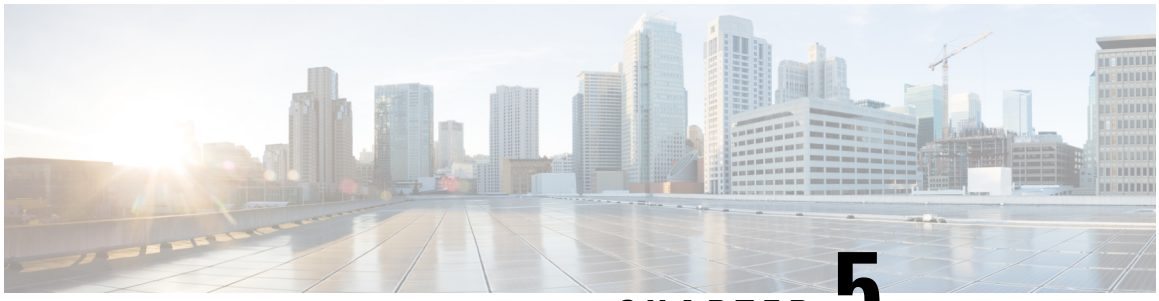
```
switch# configure terminal
switch(config)# slot 110
switch(config-slot)# provision model N2K-C2232P
switch(config-slot)# exit

switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# interface Ethernet110/1/1
switch(config-if)# description module is preprovisioned
switch(config-if)# show running-config interface Ethernet110/1/1
Time: Wed Aug 25 21:29:44 2010

version 5.0(2)N1(1)

interface Ethernet110/1/1
  description module is preprovisioned
```

The following example shows the list of pre-provisioned commands that were not applied when the module came online.

```
switch(config-if-range)# show provision failed-config 101
The following config was not applied for slot 33
================================================

interface Ethernet101/1/1
  service-policy input test


interface Ethernet101/1/2
  service-policy input test


interface Ethernet101/1/3
  service-policy input test
```

This example shows how to remove all pre-provisioned modules from a slot:

```
switch(config)# slot 2
switch(config-slot)# no provision model
switch(config-slot)#
```



CHAPTER 5

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 IPv4 networks.

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 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.

Cisco Fabric Services over Ethernet

The Cisco Fabric Services over Ethernet (CFSoE) is a reliable state transport mechanism that you can use to synchronize the actions of the vPC peer devices. CFSoE carries messages and packets for many features linked with vPC, such as STP and IGMP. Information is carried in CFS/CFSoE protocol data units (PDUs).

When you enable the vPC feature, the device automatically enables CFSoE, and you do not have to configure anything. CFSoE distributions for vPCs do not need the capabilities to distribute over IP or the CFS regions. You do not need to configure anything for the CFSoE feature to work correctly on vPCs.

You can use the show mac address-table command to display the MAC addresses that CFSoE synchronizes for the vPC peer link.

Note Do not enter the **no cfs eth distribute** or the **no cfs distribute** command. CFSoE must be enabled for vPC functionality. If you do enter either of these commands when vPC is enabled, the system displays an error message.

When you enter the **show cfs application** command, the output displays "Physical-eth," which shows the applications that are using CFSoE.

Guidelines and Limitations for CFS

CFS has the following configuration guidelines and limitations:

- If the virtual port channel (vPC) feature is enabled for your device, do not disable CFSoE.

Note CFSoE must be enabled for the vPC feature to work.

- All CFSoIP-enabled devices with similar multicast addresses form one CFSoIP fabric.
- Make sure that CFS is enabled for the applications that you want to configure.
- Anytime you lock a fabric, your username is remembered across restarts and switchovers.
- Anytime you lock a fabric, configuration changes attempted by anyone else are rejected.
- While a fabric is locked, the application holds a working copy of configuration changes in a pending database or temporary storage area, not in the running configuration.

- Configuration changes that have not been committed yet (still saved as a working copy) are not in the running configuration and do not display in the output of **show** commands.
- If you start a CFS session that requires a fabric lock but forget to end the session, an administrator can clear the session.
- An empty commit is allowed if configuration changes are not previously made. In this case, the **commit** command results in a session that acquires locks and distributes the current database.
- You can use the **commit** command only on the specific device where the fabric lock was acquired.
- CFSoIP and CFSoE are not supported for use together.
- CFS regions can be applied only to CFSoIP applications.

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 a 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

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 a lock held by an application from any device in the fabric.

Caution When you clear a lock in the fabric, any pending configurations in any device in the fabric are discarded.

Before you begin

You must have administrator permissions to release a lock.

Procedure

	Command or Action	Purpose
Step 1	(Optional) switch# **show** *application-name* **status**	Shows the current application state.
Step 2	Required: switch# **clear** *application-name* **session**	Clears the application configuration session and releases the lock on the fabric. All pending changes are discarded.
Step 3	(Optional) switch# **show** *application-name* **status**	Shows the current application state.

Example

```
switch# show ntp status
Distribution : Enabled
Last operational state: Fabric Locked
switch# clear ntp session
switch# show ntp status
Distribution : Enabled
Last operational state: No session
```

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 Smart 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 Smart Call Home application sends alerts to all network administrators regardless of their location. For the Smart 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 global 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 global 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.

	Command or Action	Purpose
		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.

Example

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 terminal**	Enters global 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.

Example

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 terminal**	Enters global 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 terminal**	Enters global 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.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **cfs ipv4 distribute**	Globally enables CFS over IPv4 for all applications on the switch.

	Command or Action	Purpose
Step 3	(Optional) switch(config)# **no cfs ipv4 distribute**	Disables (default) CFS over IPv4 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	(Optional) switch(config)# **no cfs ipv6 distribute**	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:

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

Configuring IP Multicast Addresses 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 terminal**	Enters global 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.
Step 3	(Optional) switch(config)# **no cfs ipv4 mcast-address** *ipv4-address*	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	(Optional) switch(config)# **no cfs ipv6 mcast-address** *ipv4-address*	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:

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

Default Settings for CFS

The following table lists the default settings for CFS configurations.

Table 1: 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

The CISCO-CFS-MIB contains SNMP configuration information for any CFS-related functions. See the MIB reference for your platform.

Enabling CFS to Distribute Smart Call Home Configurations

You can enable CFS to distribute Call Home configurations to all Cisco NX-OS devices in the network. The entire Call Home configuration is distributed except the device priority and the sysContact names.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **callhome**	Enters Call Home configuration mode.
Step 3	switch(config-callhome)# **distribute**	Enables CFS to distribute Smart Call Home configuration updates.
Step 4	(Optional) switch(config-callhome)# **show** *application-name* **status**	For the specified application, displays the CFS distribution status.
Step 5	(Optional) switch(config-callhome)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

```
switch# configure terminal
switch(config)# callhome
switch(config-callhome)# distribute
switch(config-callhome)# show callhome status
Distribution : Enabled
switch(config-callhome)# copy running-config startup-config
```

Enabling CFS to Distribute Device Alias Configurations

You can enable CFS to distribute device alias configurations in order to consistently administer and maintain the device alias database across all Cisco NX-OS devices in the fabric.

Before you begin

Make sure that you are in the storage VDC. To change to the storage VDC, use the **switchto vdc fcoe** command.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **device-alias distribute**	Enables CFS to distribute device alias configuration updates.
Step 3	(Optional) switch(config)# **show cfs application**	Displays the CFS distribution status.
Step 4	(Optional) switch(config)# **copy running-config startup config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to enable CFS to distribute device alias configurations:

```
switch(config)# device-alias distribute
switch(config)# show cfs application
----------------------------------------------
Application Enabled Scope
----------------------------------------------
device-alias Yes Physical-fc
switch(config)# copy running-config startup-config
[########################################] 100%
```

Enabling CFS to Distribute DPVM Configurations

You can enable CFS to distribute dynamic port VSAN membership (DPVM) configurations in order to consistently administer and maintain the DPVM database across all Cisco NX-OS devices in the fabric.

Before you begin

Make sure that you are in the storage VDC. To change to the storage VDC, use the **switchto vdc fcoe** command.

Make sure that you enable the DPVM feature. To do so, use the **feature dpvm** command.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **dpvm distribute**	Enables CFS to distribute DPVM configuration updates.
Step 3	(Optional) switch(config)# **show** *application-name* **status**	For the specified application, displays the CFS distribution status.
Step 4	(Optional) switch(config)# **copy running-config startup config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to enable CFS to distribute DPVM configurations:

```
switch(config)# dpvm distribute
switch(config)# show dpvm status
Distribution is enabled.
switch(config)# copy running-config startup-config
[########################################] 100%
```

Enabling CFS to Distribute FC Domain Configurations

You can enable CFS to distribute Fibre Channel (FC) domain configurations in order to synchronize the configuration across the fabric from the console of a single Cisco NX-OS device and to ensure consistency in the allowed domain ID lists on all devices in the VSAN.

Before you begin

Make sure that you are in the storage VDC. To change to the storage VDC, use the **switchto vdc fcoe** command.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **fcdomain distribute**	Enables CFS to distribute FC domain configuration updates.
Step 3	(Optional) switch(config)# **show** *application-name* **status**	For the specified application, displays the CFS distribution status.
Step 4	(Optional) switch(config)# **copy running-config startup config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to enable CFS to distribute FC domain configurations:

```
switch(config)# fcdomain distribute
switch(config)# show fcdomain status
fcdomain distribution is enabled
switch(config)# copy running-config startup-config
[########################################] 100%
```

Enabling CFS to Distribute FC Port Security Configurations

You can enable CFS to distribute Fibre Channel (FC) port security configurations in order to provide a single point of configuration for the entire fabric in the VSAN and to enforce the port security policies throughout the fabric.

Before you begin

Make sure that you are in the storage VDC. To change to the storage VDC, use the **switchto vdc fcoe** command.

Make sure that you enable the FC port security feature. To do so, use the **feature fc-port-security** command.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **fc-port-security distribute**	Enables CFS to distribute FC port security configuration updates.
Step 3	(Optional) switch(config)# **show cfs application**	Displays the CFS distribution status.
Step 4	(Optional) switch(config)# **copy running-config startup config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to enable CFS to distribute FC port security configurations:

```
switch(config)# fc-port-security distribute
switch(config)# show cfs application
----------------------------------------------
Application Enabled Scope
----------------------------------------------
fc-port-securi Yes Logical
switch(config)# copy running-config startup-config
[########################################] 100%
```

Enabling CFS to Distribute FC Timer Configurations

You can enable CFS to distribute Fibre Channel (FC) timer configurations for all Cisco NX-OS devices in the fabric.

Before you begin

Make sure that you are in the storage VDC. To change to the storage VDC, use the **switchto vdc fcoe** command.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **fctimer distribute**	Enables CFS to distribute FC timer configuration updates.
Step 3	(Optional) switch(config)# **show** *application-name* **status**	For the specified application, displays the CFS distribution status.
Step 4	(Optional) switch(config)# **copy running-config startup config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to enable CFS to distribute FC timer configurations:

```
switch(config)# fctimer distribute
switch(config)# show fctimer status
Distribution : Enabled
switch(config)# copy running-config startup-config
[########################################] 100%
```

Enabling CFS to Distribute IVR Configurations

You can enable CFS to distribute inter-VSAN routing (IVR) configurations in order to enable efficient IVR configuration management and to provide a single point of configuration for the entire fabric in the VSAN.

Before you begin

Make sure that you are in the storage VDC. To change to the storage VDC, use the **switchto vdc fcoe** command.

Make sure that you install the Advanced SAN Services license.

Make sure that you enable the IVR feature. To do so, use the **feature ivr** command.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **ivr distribute**	Enables CFS to distribute IVR configuration updates. **Note** You must enable IVR distribution on all IVR-enabled switches in the fabric.
Step 3	(Optional) switch(config)# **show cfs application**	Displays the CFS distribution status.
Step 4	(Optional) switch(config)# **copy running-config startup config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to enable CFS to distribute IVR configurations:

```
switch(config)# ivr distribute
switch(config)# show cfs application
----------------------------------------------
Application Enabled Scope
----------------------------------------------
ivr Yes Physical-fc
switch(config)# copy running-config startup-config
[########################################] 100%
```

Enabling CFS to Distribute NTP Configurations

You can enable CFS to distribute NTP configurations to all Cisco NX-OS devices in the network.

Before you begin

Make sure that you enable the NTP feature (using the **feature ntp** command).

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **ntp distribute**	Enables CFS to distribute NTP configuration updates.
Step 3	(Optional) switch(config)# **show** *application-name* **status**	For the specified application, displays the CFS distribution status.

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

Example

```
switch# configure terminal
switch(config)# ntp distribute
switch(config)# show ntp status
Distribution : Enabled
switch(config)# copy running-config startup-config
```

Enabling CFS to Distribute RADIUS Configurations

You can enable CFS to distribute RADIUS configurations to all Cisco NX-OS devices in the network.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **radius distribute**	Enables CFS to distribute RADIUS configuration updates.
Step 3	(Optional) switch(config)# **show** *application-name* **status**	For the specified application, displays the CFS distribution status.
Step 4	(Optional) switch(config)# **copy running-config startup config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

```
switch# configure terminal
switch(config)# radius distribute
switch(config)# show radius status
Distribution : Enabled
switch(config)# copy running-config startup-config
```

Enabling CFS to Distribute RSCN Configurations

You can enable CFS to distribute registered state change notification (RSCN) configurations to all Cisco NX-OS devices in the fabric.

Before you begin

Make sure that you are in the storage VDC. To change to the storage VDC, use the **switchto vdc fcoe** command.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **rscn distribute**	Enables CFS to distribute RSCN configuration updates.
Step 3	(Optional) switch(config)# **show cfs application**	Displays the CFS distribution status.
Step 4	(Optional) switch(config)# **copy running-config startup config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to enable CFS to distribute RSCN configurations:

```
switch(config)# rscn distribute
switch(config)# show cfs application
----------------------------------------------
Application Enabled Scope
----------------------------------------------
rscn Yes Logical
switch(config)# copy running-config startup-config
[########################################] 100%
```

Enabling CFS to Distribute TACACS+ Configurations

You can enable CFS to distribute TACACS+ configurations to all Cisco NX-OS devices in the network.

Before you begin

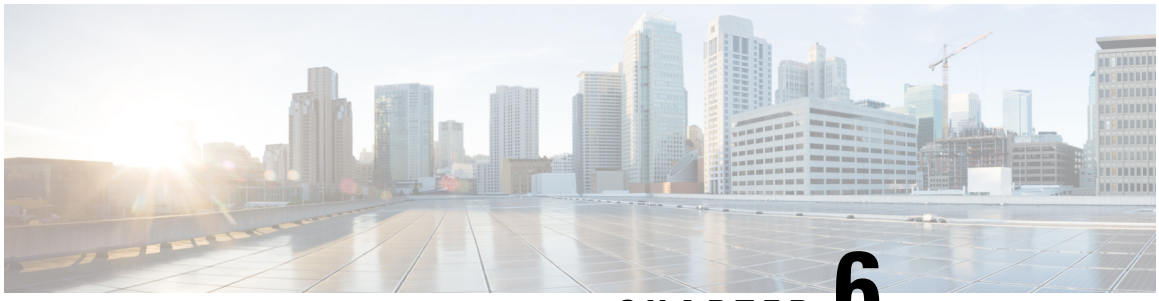
Make sure that you enable the TACACS+ feature (using the **feature tacacs+** command).

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **tacacs+ distribute**	Enables CFS to distribute TACACS+ configuration updates.
Step 3	(Optional) switch(config)# **show** *application-name* **status**	For the specified application, displays the CFS distribution status.
Step 4	(Optional) switch(config)# **copy running-config startup config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

```
switch# configure terminal
switch(config)# tacacs+ distribute
switch(config)# show tacacs+ status
Distribution : Enabled
switch(config)# copy running-config startup-config
```



CHAPTER 6

Configuring PTP

This chapter contains 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.

Starting from Cisco NXOS Release 6.0(2)A8(3), PTP supports configuring multiple PTP clocking domains, PTP grandmaster capability, PTP cost on interfaces for slave and passive election, and clock identity.

All the switches in a multi-domain environment, belong to one domain. The switches that are the part of boundary clock, must have multi-domain feature enabled on them. Each domain has user configurable parameters such as domain priority, clock class threshold and clock accuracy threshold. The clocks in each domain remain synchronized with the master clock in that domain. If the GPS in a domain fails, the master clock in the domain synchronizes time and data sets associated with the announce messages from the master clock in the domain where the GPS is active. If the master clock from the highest priority domain does not meet the clock quality attributes, a clock in the subsequent domain that match the criteria is selected. The Best

Master Clock Algorithm (BMCA) is used to select the master clock if none of the domains has the desired clock quality attributes. If all the domains have equal priority and the threshold values less than master clock attributes or if the threshold values are greater than the master clock attributes, BMCA is used to select the master clock.

Grandmaster capability feature controls the switch's ability of propagating its clock to other devices that it is connected to. When the switch receives announce messages on an interface, it checks the clock class threshold and clock accuracy threshold values. If the values of these parameters are within the predefined limits, then the switch acts as per PTP standards specified in IEEE 1588v2. If the switch does not receive announce messages from external sources or if the parameters of the announce messages received are not within the predefined limits, the port state will be changed to listening mode. On a switch with no slave ports, the state of all the PTP enabled ports is rendered as listening and on a switch with one slave port, the BMCA is used to determine states on all PTP enabled ports. Convergence time prevents timing loops at the PTP level when grandmaster capability is disabled on a switch. If the slave port is not selected on the switch, all the ports on the switch will be in listening state for a minimum interval specified in the convergence time. The convergence time range is from 3 to 2600 seconds and the default value is 3 seconds.

The interface cost applies to each PTP enabled port if the switch has more than one path to grandmaster clock. The port with the least cost value is elected as slave and the rest of the ports will remain as passive ports.

The clock identity is a unique 8-octet array presented in the form of a character array based on the switch MAC address. The clock identity is determined from MAC according to the IEEE1588v2-2008 specifications. The clock ID is a combination of bytes in a VLAN MAC address as defined in IEEE1588v2.

Only Cisco Nexus 3000 Series switches support PTP. Cisco Nexus 3100 Series switches do not support this feature.

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 that are 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. We recommend that you deploy a Grand Master Clock (10 MHz) upstream. The servers contain clocks that require synchronization and are 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 that it was received. For every synchronization message, there is a follow-up message. The number of sync messages should be equal to the number of follow-up messages.
- 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 number of delay request messages should be equal to the number of delay response messages.
- The slave uses these timestamps to adjust its clock to the time of its master.

Clock Management

By default, Cisco NX-OS uses NTP to update the system clock. However, if the **clock protocol** property is configured to **PTP**, PTP is allowed to update the system clock.

If PTP is enabled, NTP does not update the system time.

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 is not supported on interfaces which reside on GEMs (Generic Expansion modules) and can only be configured on interfaces which are fixed to the chassis (non-modular)
- PTP is only configurable on switch ports. Configuring PTP on FEX ports 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.
- PTP is only supported on physical Ethernet-based ports.
- In VPC environments, PTP must be individually configured on each member port.
- PTP over FabricPath is not supported.
- In case of a nondisruptive ISSU from a release earlier than Cisco NX-OS release 7.1(1)N1(1) to the latest release, perform reload before enabling the PTP feature.

Default Settings for PTP

The following table lists the default settings for PTP parameters.

Table 2: Default PTP Parameters

Parameters	Default
PTP	Disabled
PTP version	2
PTP domain	0. PTP multi domain is disabled by default.
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	1 log second
PTP announce timeout	3 announce intervals
PTP minimum delay request interval	1 log second
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 format.
Step 4	(Optional) switch(config) # [**no**] **ptp domain** *number*	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	(Optional) switch(config) # [**no**] **ptp priority1** *value*	Configures the priority1 value to use when advertising this clock. This value overrides the default criteria (clock quality, clock class, and so on) for the best master clock selection. Lower values take precedence. The range for the *value* is from 0 to 255.
Step 6	(Optional) switch(config) # [**no**] **ptp priority2** *value*	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	(Optional) switch(config) # **show ptp brief**	Displays the PTP status.
Step 8	(Optional) switch(config) # **show ptp clock**	Displays the properties of the local clock.
Step 9	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

The following 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 interface configuration mode. **Note** If this is a QSFP+ GEM or a breakout port, the *port* syntax is *QSFP-module*/*port*.
Step 3	switch(config-if) # [**no**] **feature ptp**	Enables or disables PTP on an interface.
Step 4	(Optional) switch(config-if) # [**no**] **ptp announce** {**interval** *log seconds* | **timeout** *count*}	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	(Optional) switch(config-if) # [**no**] **ptp delay request minimum interval** *log seconds*	Configures the minimum interval allowed between PTP delay-request messages when the port is in the master state. The range is from log(-6) to log(1) seconds. Where, log(-2) = 2 frames per second.
Step 6	(Optional) switch(config-if) # [**no**] **ptp sync interval** *log seconds*	Configures the interval between PTP synchronization messages on an interface. The range for the PTP synchronization interval is from -3 log second to 1 log second
Step 7	(Optional) switch(config-if) # [**no**] **ptp vlan** *vlan-id*	Specifies the VLAN for the interface where PTP is being enabled. You can only enable PTP on one VLAN on an interface.

	Command or Action	Purpose
		The range is from 1 to 4094.
Step 8	(Optional) switch(config-if) # **show ptp brief**	Displays the PTP status.
Step 9	(Optional) switch(config-if) # **show ptp port interface** *interface slot/port*	Displays the status of the PTP port. **Note** If this is a QSFP+ GEM or a breakout port, the *port* syntax is *QSFP-module*/*port*.
Step 10	(Optional) switch(config-if)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

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 1/1
PTP Port Dataset: Eth1/1
Port identity: clock identity: f4:4e:05:ff:fe:84:7e:7c
Port identity: port number: 0
PTP version: 2
Port state: Slave
VLAN info: 1
Delay request interval(log mean): 0
Announce receipt time out: 3
Peer mean path delay: 0
Announce interval(log mean): 1
Sync interval(log mean): 1
Delay Mechanism: End to End
Cost: 255
Domain: 5
switch(config-if)#
```

Verifying the PTP Configuration

Use one of the following commands to verify the configuration:

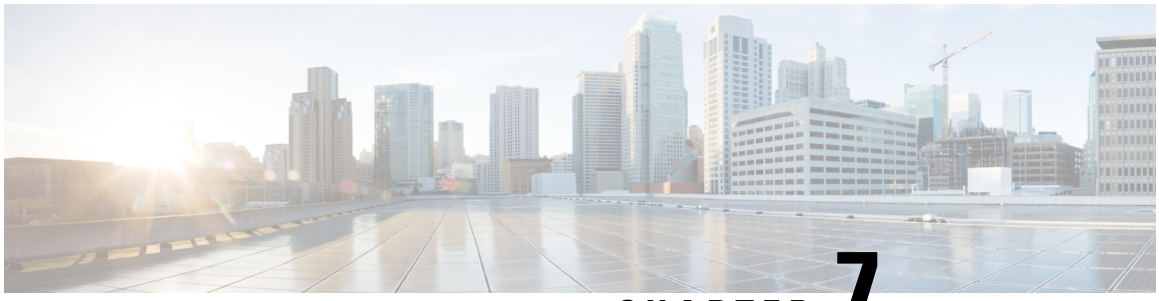
Table 3: PTP Show Commands

Command	Purpose
show ptp brief	Displays the PTP status.
show ptp clock	Displays the properties of the local clock, including the clock identity.
show ptp clock 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. **Note** If this is a QSFP+ GEM or a breakout port, the *port* syntax is *QSFP-module*/*port*.
show ptp domain data	Displays multiple domain data, domain priority, clock threshold and information about grandmaster capabilities.
show ptp interface domain	Displays information about the interface to domain association.
show ptp cost	Displays PTP port to cost association.

Feature History for PTP

This table lists the release history for this feature.

Feature Name	Release	Information
PTP	7.1(1)N1(1)	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).



CHAPTER 7

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. However, the network-operator role cannot run the **show running-config** and **show startup-config** commands.

san-admin

Complete read and write access to Fibre Channel and FCoE administrative tasks using SNMP or CLI.

san-admin

Complete read and write access to FCoE administrative tasks using SNMP or CLI.

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.

Note Only network-admin user can perform a Checkpoint or Rollback in the RBAC roles. Though other users have these commands as a permit rule in their role, the user access is denied when you try to execute these commands.

Predefined SAN Admin User Role

The SAN admin user role is a noneditable, predefined user role that is designed to provide separation between LAN and SAN administrative tasks. Users that have been assigned the SAN admin user role have read-only access to all Ethernet configuration tasks. Write access for Ethernet features is not granted to SAN admin users unless it is assigned to them through another user role.

The following capabilities are permitted to SAN admin users:

- Interface configuration
- VSAN configuration, including database and membership
- Mapping of preconfigured VLANs for FCoE to VSANs
- Zoning configuration
- Configuration of SNMP-related parameters, except SNMP community and SNMP users
- Read-only access to all other configurations
- Configuration and management of SAN features such as the following:
  - FC-SP
  - FC-PORT-SECURITY
  - FCoE
  - FCoE-NPV
  - FPORT-CHANNEL-TRUNK
  - PORT-TRACK
  - FABRIC-BINDING
- Configuration and management for the following of EXEC mode commands:

- DEBUG
- FCDOMAIN
- FCPING
- SAN-PORT-CHANNEL
- SHOW
- ZONE
- ZONESET

Note The SAN Admin role permits configuration on all interface types. The predefined SAN Admin user role was designed to allow access to all interfaces—including Ethernet interfaces—so it would not interfere with SNMP operations.

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 device. 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.

SAN Admin Role-Feature Rule Mapping

The SAN admin role is not editable. The following role-features are part of preconfigured role. The preconfigured role comes complete read access and the following rules:

Table 4: Role-Feature Rules for SAN Admin User Role

Feature	Permissions
copy	Read and write permissions for copy-related commands
fabric-binding	Read and write permissions for fabric binding-related commands
fcoe	Read and write permissions for Fibre Channel over Ethernet-related commands
fdmi	Read and write permissions for Fabric Device Management Interface (FDMI)-related commands
fspf	Read and write permissions for Fabric Shortest Path First (FSPF)-related commands
interface	Read and write permissions for interface-related commands, which
port-track	Read and write permissions for port track-related commands
port-security	Read and write permissions for port security-related commands
rdl	Read and write permissions for Remote Domain Loopback (RDL)-related commands
rmon	Read and write permissions for RMON-related commands
rscn	Read and write permissions for Registered State Change Notification (RSCN)-related commands
snmp	Read and write permissions for SNMP-related commands
snmpTargetAddrEntry	Read and write permissions for SNMP trap target-related commands
snmpTargetParamsEntry	Read and write permissions for SNMP trap target parameter-related commands
span	Read and write permissions for SPAN-related commands
trapRegEntry	Read and write permissions for SNMP trap registry-related commands
vsan	Read and write permissions for VSAN-related commands

Feature	Permissions
vsanIfvsan	Read and write permissions for FCoE VLAN-VSAN mapping command-related commands
wwnm	Read and write permissions for World Wide Name (WWN)-related commands
zone	Read and write permissions for zoning commands

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

- san-admin
- shutdown
- sync
- sys
- uucp
- xfs

Caution The Cisco Nexus 5000 and 6000 Series switch does not support all numeric usernames, even if those usernames were created in TACACS+ or RADIUS. If an all numeric username exists on an AAA server and is entered during login, the switch rejects the login request.

Usernames must begin with an alphanumeric character and can contain only these special characters: (+ = . _ \ -). The # and ! symbols are not supported. If the username contains characters that are not allowed, the specified user is unable to log in. Effective from Cisco NX-OS release 7.3(0)N1(1), usernames starting with _(underscore) are supported.

User Password Requirements

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

Note Starting from Cisco NX-OS Release 7.2(0)N1(1), special characters, such as the dollar sign ($) or the percent sign (%), can be used in Cisco Nexus device passwords.

If a password is trivial (such as a short, easy-to-decipher password), the Cisco Nexus device 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

User accounts have 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	(Optional) switch(config)# **show role**	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.

	Command or Action	Purpose
		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	(Optional) switch# **show user-account**	Displays the role configuration.
Step 6	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

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 SAN Admin Users

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **username** *user-id* **role san-admin password** *password*	Configures SAN admin user role access for the specified user.
Step 3	(Optional) switch(config) # **show user-account**	Displays the role configuration.
Step 4	(Optional) switch(config) # **show snmp-user**	Displays the SNMP user configuration.
Step 5	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

The following example shows how to configure a SAN admin user and display the user account and SNMP user configuration:

```
switch# configure terminal
switch(config)# username user1 role san-admin password xyz123
switch(config)# show user-account
```

```
user:admin
     this user account has no expiry date
     roles:network-admin
user:user1
     this user account has no expiry date
     roles:san-admin
switch(config) # show snmp user
_____________________________________________________________________________
     SNMP USERS
_____________________________________________________________________________

User     Auth  Priv(enforce)  Groups
____     ____  _____________  ______
admin    md5   des(no)        network-admin
user1    md5   des(no)        san-admin

_____________________________________________________________________________
NOTIFICATION TARGET USES (configured for sending V3 Inform)
_____________________________________________________________________________

User     Auth   Priv
____     ____   ____

switch(config) #
```

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.

Note Regardless of the read-write rule configured for a user role, some commands can be executed only through the predefined network-admin and vdc-admin roles.

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.

	Command or Action	Purpose
		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	(Optional) switch(config-role)# **description** *text*	Configures the role description. You can include spaces in the description.
Step 8	switch(config-role)# **end**	Exits role configuration mode.
Step 9	(Optional) switch# **show role**	Displays the user role configuration.
Step 10	(Optional) switch# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

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	(Optional) switch# **show role feature-group**	Displays the role feature group configuration.
Step 5	(Optional) switch# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

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.
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.

	Command or Action	Purpose
		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	(Optional) switch(config-role) # **show role**	Displays the role configuration.
Step 7	(Optional) switch(config-role) # **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

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.
Step 5	switch(config-role-vlan) # **exit**	Exits role VLAN policy configuration mode.
Step 6	(Optional) switch# **show role**	Displays the role configuration.

	Command or Action	Purpose
Step 7	(Optional) switch# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Changing User Role VSAN Policies

You can change a user role VSAN policy to limit the VSANs that the user can access.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config-role) # **role name** *role-name*	Specifies a user role and enters role configuration mode.
Step 3	switch(config-role) # **vsan policy deny**	Enters role VSAN policy configuration mode.
Step 4	switch(config-role-vsan) # **permit vsan** *vsan-list*	Specifies a range of VSANs that the role can access. Repeat this command for as many VSANs as needed.
Step 5	switch(config-role-vsan) # **exit**	Exits role VSAN policy configuration mode.
Step 6	(Optional) switch# **show role**	Displays the role configuration.
Step 7	(Optional) switch# **copy running-config startup-config**	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.

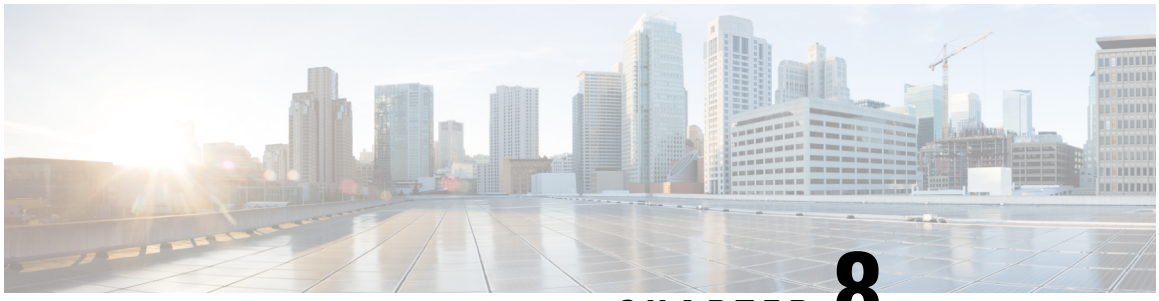
Command	Purpose
show user-account	Displays user account information.

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.
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 8

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 access control list (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	(Optional) switch(config-s)# **show configuration session** [*name*]	Displays the contents of the session.
Step 3	(Optional) switch(config-s)# **save** *location*	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	(Optional) switch(config-s-acl)# **permit** *protocol source destination*	Adds a permit statement to the ACL.
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	(Optional) switch# **show configuration session** [*name*]	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

The following example shows how to create a configuration session for ACLs:

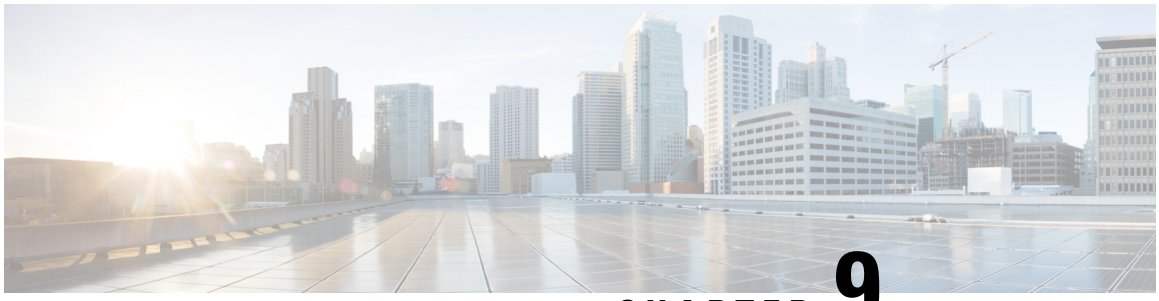
```
switch# configure session name test2
switch(config-s)# ip access-list acl2
switch(config-s-acl)# permit tcp any any
switch(config-s-acl)# exit
switch(config-s)# interface Ethernet 1/4
switch(config-s-ip)# ip port access-group acl2 in
switch(config-s-ip)# exit
switch(config-s)# verify
switch(config-s)# 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 9

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 6: 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.
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 device 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 7: 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.

Note When the switch reaches the intake temperature threshold and does not go within the limits in 60 seconds, the switch will power off and the power supplies will have to be re-seated to recover the switch

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

Table 8: 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.
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.

Note When the switch exceeds the intake temperature threshold of 40 degrees Celsius and does not decrease to within the threshold limits in 60 seconds, the switch powers off and the power supplies must be re-seated to recover the switch.

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 9: 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 10: 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 global 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	(Optional) switch# **show diagnostic bootup level**	Displays the bootup diagnostic level (bypass or complete) that is currently in place on the switch.

Example

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

Use the following commands to verify online diagnostics configuration information:

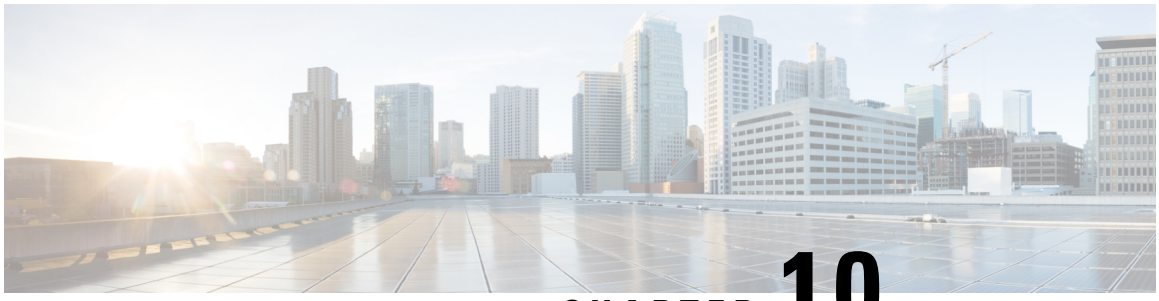
Command	Purpose
show diagnostic bootup level	Displays the bootup diagnostics level.
show diagnostic result module *slot*	Displays the results of the diagnostics tests.

Default Settings for Online Diagnostics

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

Table 11: Default Online Diagnostics Parameters

Parameters	Default
Bootup diagnostics level	complete



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 device 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 12: System Message Severity Levels

Level	Description
0 – emergency	System unusable
1 – alert	Immediate action needed
2 – critical	Critical condition
3 – error	Error condition
4 – warning	Warning condition

Level	Description
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 13: 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 global 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: • 0 – emergency • 1 – alert • 2 – critical • 3 – error

	Command or Action	Purpose
		• 4 – warning • 5 – notification • 6 – informational • 7 – debugging If the severity level is not specified, the default of 2 is used.
Step 4	(Optional) switch(config)# **no logging console** [*severity-level*]	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	(Optional) switch(config)# **no logging monitor** [*severity-level*]	Disables logging messages to Telnet and SSH sessions.
Step 7	(Optional) switch# **show logging console**	Displays the console logging configuration.
Step 8	(Optional) switch# **show logging monitor**	Displays the monitor logging configuration.
Step 9	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

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 global 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. When you configure a new logfile without specifying the size, the existing/previously specified logfile size is assigned and the default file size is not considered.

	Command or Action	Purpose
		Severity levels range from 0 to 7: • 0 – emergency • 1 – alert • 2 – critical • 3 – error • 4 – warning • 5 – notification • 6 – informational • 7 – debugging The file size is from 4096 to 10485760 bytes.
Step 3	(Optional) switch(config)# **no logging logfile** [*logfile-name severity-level* [**size** *bytes*]]	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	(Optional) switch# **show logging info**	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	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

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

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 global 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

	Command or Action	Purpose
		• 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. **Note** If the default severity and current session severity of a component is the same, then the logging level for the component will not be displayed in the running configuration.
Step 4	(Optional) switch(config)# **no logging module** [*severity-level*]	Disables module log messages.
Step 5	(Optional) switch(config)# **no logging level** [*facility severity-level*]	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	(Optional) switch# **show logging module**	Displays the module logging configuration.
Step 7	(Optional) switch# **show logging level** [*facility*]	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	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

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 global 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	(Optional) switch(config)# **no logging timestamp** {**microseconds** | **milliseconds** | **seconds**}	Resets the logging time-stamp units to the default of seconds.
Step 4	(Optional) switch# **show logging timestamp**	Displays the logging time-stamp units configured.
Step 5	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

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 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

	Command or Action	Purpose
		from 0 to 7. See Table 12: System Message Severity Levels , on page 95. • 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 Cisco Fabric Services (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. **Note** Debugging is a CLI facility but the debug syslogs are not sent to the server.
Step 3	(Optional) **no logging server** *host* **Example:** `switch(config)# no logging server 172.28.254.254 5`	Removes the logging server for the specified host.
Step 4	(Optional) **show logging server** **Example:** `switch# show logging server`	Displays the syslog server configuration.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

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 14: 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 global 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	(Optional) switch(config)# **no logging distribute**	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	(Optional) switch# **show logging pending**	Displays the pending changes to the syslog server configuration.

	Command or Action	Purpose
Step 7	(Optional) switch# **show logging pending-diff**	Displays the differences from the current syslog server configuration to the pending changes of the syslog server configuration.
Step 8	(Optional) switch# **show logging internal info**	Displays information about the current state of the syslog server distribution and the last action taken.
Step 9	(Optional) switch# **copy running-config startup-config**	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.

Example

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

Use these commands to verify system message logging configuration information:

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.
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.

Configuring ACL Logging

Information About ACL Logging

The Access Control List (ACL) logging feature allows the logging of the packets which hit the IPv4 ACLs. The log message is displayed on a flow basis. The flow is identified using the combination of IP source address, destination address, Layer 4 protocol, and the Layer 4 source/destination ports on an interface. The log message is generated based on the following conditions:

- When a new flow is created (INFO message)
- When the flow's packet threshold is reached (WARNING message)
- At the end of a periodic interval (default five minutes) with the information about how many packets hit the flow (INFO message - configurable)

Along with the above, when the number of flows exceeds a threshold in a given interval, a warning message is logged and the flow is not added to the logging cache.

The following table describes the limitation in the Cisco Nexus device.

Table 15: ACL Logging Support Table

Feature	**Cisco Nexus Device**	
	Logging support	
PACL	Yes	Drop only
Ingress RACL	Yes	Drop only
Egress RACL	Yes	Drop only
Ingress VACL	Yes	Drop only
Egress VACL	Yes	Drop only
RBACL	N/A	
VTY ACL In/Out	Yes	Permit/Drop
Ingress RACL on mgmt0	Yes	Permit/Drop
SNMP ACL		
NTP ACL		

Table 16: ACL Logging Support Table

Feature	Cisco Nexus Device	
	Logging support	
PACL	Yes	Drop only
Egress RACL	Yes	Drop only
Ingress RACL	No	
Egress VACL	Yes	Drop only
Ingress VACL	Yes	
RBACL	N/A	
VTY ACL In/Out	Yes	Drop Log
Ingress RACL on mgmt0	Yes	Permit/Drop
SNMP ACL		
NTP ACL		
CTS	Yes	
Software-Based RACL	Yes	

Except for the VTY ACL, all other ACLs support ACL logging for only the "deny" ACE entries. However, since the same ACL can be applied for both vty ACL and other features, "permit <> log" CLI cannot be blocked. However, applying such an ACL to any of the interfaces/vlans can be prevented. Mgmt0 supports permit logging.

In the Cisco Nexus device, CTS is not supported, therefore RBACL is not supported.

ACL logging is not supported for IPv6 and MAC ACLs. It is supported on all interfaces where PACL, RACL, VACL and VTY can be applied, including FEX HIF interfaces.

The ACL logging is rate-limited. All the packets that hit the ACL are not sent to the sup. The rate limiter function is per switch and is applied across all ASIC and TCAM regions. The following CLIs will be provided to configure the rate.

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.

	Command or Action	Purpose
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)# **logging ip access-list include sgt**	Includes the source group tag information in the syslogs.
Step 6	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

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

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.

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

Example

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 101 and Configuring System Message Logging to a File, on page 99.
Step 3	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Configuring Rate Limiter for ACL Logging

You can limit the number of logged packets that are sent to the supervisor (CPU) to be logged to the cache.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	**hardware rate-limiter access-list-log packets** *num-packets*	*num-packets*—Value in packets per second. Valid range is 50 to 600000. The default is 100 packets per second.

Example

This example shows how to set the rate limiter to 1000 packets per second.

```
switch# configure terminal
switch(config)# hardware rate-limiter access-list-log packets 1000
```

Clearing ACL Logs

You can clear the ACL logs.

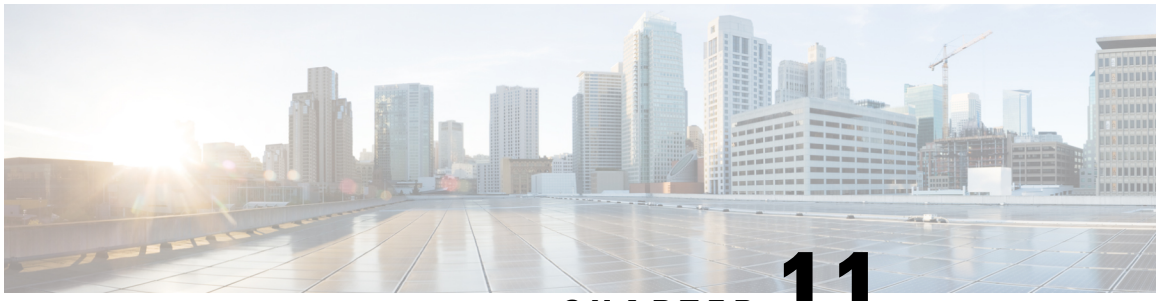
Procedure

	Command or Action	Purpose
Step 1	switch# **clear logging ip access-list cache**	Clears the IP configuration access list cache.

Verifying ACL Logging

Use one of the following commands to verify the configuration:

Command	Purpose
show logging ip access-list status	Displays the ACLLOG status.
show logging ip access-list cache [**detail**]	Displays the entries in cache and optionally additional details.



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 (TAC).

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 the Cisco 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—Text that is suitable for pagers or printed reports.
  - Full Text—Fully formatted message information that is 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 recipients for the Smart Call Home messages that are 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 devices. 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 17: 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**
Configuration	Periodic events related to configuration.	**show version** **show module** **show running-config all** **show startup-config**
System	Events generated by a 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**

Alert Group	Description	Executed Commands
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 **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 18: 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.

Smart Call Home Level	Keyword	Syslog Level	Description
4	Minor	Error (3)	Minor conditions.
3	Warning	Warning (4)	Warning conditions.
2	Notification	Notice (5)	Basic notification and informational messages.
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 19: 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 20: 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

Data Item (Plain Text and XML)	Description (Plain Text and XML)	XML Tag (XML Only)
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
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.		
The following fields may be repeated if multiple CLI commands are executed for this alert group.		

Data Item (Plain Text and XML)	Description (Plain Text and XML)	XML Tag (XML Only)
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 21: 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 22: 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
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

Data Item (Plain Text and XML)	Description (Plain Text and XML)	XML Tag (XML Only)
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 23: 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

- You must have e-mail server connectivity.
- You must have access to contact name (SNMP server contact), phone, and street address information.
- You must have IP connectivity between the switch and the e-mail server.
- You must have an active service contract for the device that you are configuring.

Default Call Home Settings

Table 24: 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

Parameters	Default
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

- Know the sMARTnet contract number for your switch
- Know your e-mail address
- Know 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 an 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	(Optional) switch(config-callhome)# **contract-id** *contract-number*	Configures the contract number for this switch from the service agreement. The *contract-number* can be up to 255 alphanumeric characters.
Step 8	(Optional) switch(config-callhome)# **customer-id** *customer-number*	Configures the customer number for this switch from the service agreement. The *customer-number* can be up to 255 alphanumeric characters.
Step 9	(Optional) switch(config-callhome)# **site-id** *site-number*	Configures the site number for this switch. The *site-number* can be up to 255 alphanumeric characters in free format.

	Command or Action	Purpose
Step 10	(Optional) switch(config-callhome)# **switch-priority** *number*	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. **Note** Switch priority is used by the operations personnel or TAC support personnel to decide which Call Home message should be responded to first. You can prioritize Call Home alerts of the same severity from each switch.
Step 11	(Optional) switch# **show callhome**	Displays a summary of the Smart Call Home configuration.
Step 12	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

The following 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.
Step 2	switch(config)# **callhome**	Enters Smart Call Home configuration mode.

	Command or Action	Purpose
Step 3	switch(config-callhome)# **destination-profile** {**ciscoTAC-1** {**alert-group** *group* | **email-addr** *address* | **http** *URL* | **transport-method** {**email** | **http**}} | *profilename* {**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 your platform.
Step 4	(Optional) switch# **show callhome destination-profile** [**profile** *name*]	Displays information about one or more destination profiles.
Step 5	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

The following 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 Smart Call Home message severity level for this destination profile. The switch sends only alerts that have a matching or higher Smart 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	(Optional) switch# **show callhome destination-profile** [**profile** *name*]	Displays information about one or more destination profiles.
Step 7	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

The following example shows how to modify a destination profile for Smart 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	(Optional) switch# **show callhome destination-profile** [**profile** *name*]	Displays information about one or more destination profiles.
Step 5	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

The following 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, you can 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 **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.
Step 3	switch(config-callhome)# **alert-group** {**Configuration** | **Diagnostic** | **Environmental** | **Inventory** | **License** | **Linecard-Hardware** |	Adds the **show** command output to any Call Home messages sent for this alert group. Only valid **show** commands are accepted.

	Command or Action	Purpose
	Supervisor-Hardware | **Syslog-group-port** | **System** | **Test**} **user-def-cmd** *show-cmd*	**Note** You cannot add user-defined **show** commands to the CiscoTAC-1 destination profile.
Step 4	(Optional) switch# **show callhome user-def-cmds**	Displays information about all user-defined **show** commands added to alert groups.
Step 5	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

The following example shows how to add the **show ip routing** command to 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 Smart 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 *number* range is from1 to 65535. The default port number is 25. Optionally, you can configure the VRF instance to use when communicating with this SMTP server.
Step 4	(Optional) switch(config-callhome)# **transport email from** *email-address*	Configures the e-mail from field for Smart Call Home messages.

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

Example

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

Example

The following 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	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

The following 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	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

The following 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.
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.

	Command or Action	Purpose
Step 4	switch(config-callhome) # **callhome test**	Sends a test message to all configured destinations.
Step 5	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

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

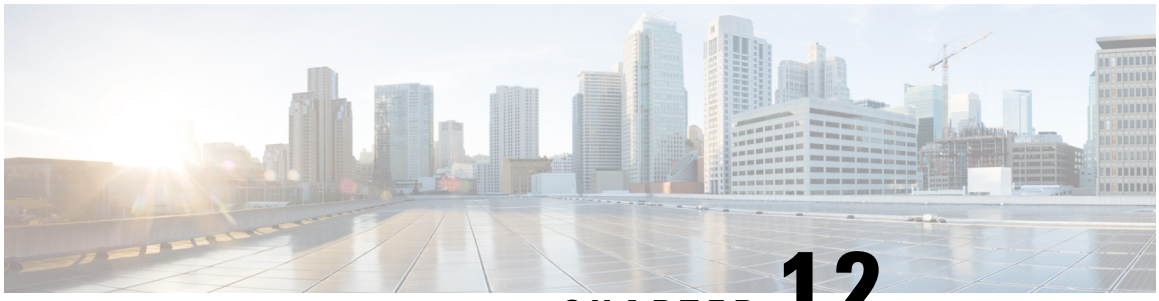
This sample shows the XML format for a syslog port alert-group notification:

```
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 Rollback

This chapter contains the following sections:

Information About Rollbacks

The rollback feature allows you to take a snapshot, or user checkpoint, of the Cisco NX-OS configuration and then reapply that configuration to your 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.

You can create a checkpoint copy of the current running configuration at any time. Cisco NX-OS saves this checkpoint as an ASCII file which you can use to roll back the running configuration to the checkpoint configuration at a future time. You can create multiple checkpoints to save different versions of your running configuration.

When you roll back the running configuration, you can trigger an atomic rollback. An atomic rollback implements a rollback only if no errors occur.

Guidelines and Limitations for Rollback

Rollback has the following configuration guidelines and limitations:

- You can create up to ten checkpoint copies.
- You cannot apply the checkpoint file of one switch into another switch.
- Your checkpoint file names must be 75 characters or less.
- You cannot start a checkpoint filename with the word system.
- You can start a checkpoint filename with the word auto.
- You can name a checkpoint file summary or any abbreviation of the word summary.

- Only one user can perform a checkpoint, rollback, or copy the running configuration to the startup configuration at the same time.
- After you enter the **write erase** and **reload** command, checkpoints are deleted. You can use the clear checkpoint database command to clear out all checkpoint files.
- When checkpoints are created on bootflash, differences with the running-system configuration cannot be performed before performing the rollback, and the system reports “No Changes.”
- Checkpoints are local to a switch.
- Checkpoints that are created using the **checkpoint** and **checkpoint** *checkpoint_name* commands are present upon a switchover for all switches.
- A rollback to files on bootflash is supported only on files that are created using the **checkpoint** *checkpoint_name* command and not on any other type of ASCII file.
- Checkpoint names must be unique. You cannot overwrite previously saved checkpoints with the same name.
- Checkpoints are not supported post upgrade or downgrade.
- The Cisco NX-OS commands may differ from the Cisco IOS commands.

Creating a Checkpoint

You can create up to ten checkpoints of your configuration per switch.

Procedure

	Command or Action	Purpose
Step 1	switch# **checkpoint** { [*cp-name*] [**description** *descr*] |**file** *file-name* **Example:** `switch# checkpoint stable`	Creates a checkpoint of the running configuration to either a user checkpoint name or a file. The checkpoint name can be any alphanumeric string up to 80 characters but cannot contain spaces. If you do not provide a name, Cisco NX-OS sets the checkpoint name to user-checkpoint-<number> where number is from 1 to 10. The description can contain up to 80 alphanumeric characters, including spaces.
Step 2	(Optional) switch# **no checkpoint***cp-name* **Example:** `switch# no checkpoint stable`	You can use the **no** form of the **checkpoint** command to remove a checkpoint name. Use the **delete** command to remove a checkpoint file.
Step 3	(Optional) switch# **show checkpoint***cp-name* **Example:** [**all**]	Displays the contents of the checkpoint name.

	Command or Action	Purpose
	`switch# show checkpoint stable`	

Implementing a Rollback

You can implement a rollback to a checkpoint name or file. Before you implement a rollback, you can view the differences between source and destination checkpoints that reference current or saved configurations.

Note If you make a configuration change during an atomic rollback, the rollback will fail.

Procedure

	Command or Action	Purpose
Step 1	**show diff rollback-patch** {**checkpoint** *src-cp-name* | **running-config** | **startup-config** | **file** *source-file*} {**checkpoint** *dest-cp-name* | **running-config** | **startup-config** | **file** *dest-file*} **Example:** `switch# show diff rollback-patch checkpoint stable running-config`	Displays the differences between the source and destination checkpoint selections.
Step 2	**rollback running-config** {**checkpoint** *cp-name* | **file** *cp-file*} **atomic** **Example:** `switch# rollback running-config checkpoint stable`	Creates an atomic rollback to the specified checkpoint name or file if no errors occur.

Example

The following example shows how to create a checkpoint file and then implement an atomic rollback to a user checkpoint name:

```
switch# checkpoint stable
switch# rollback running-config checkpoint stable atomic
```

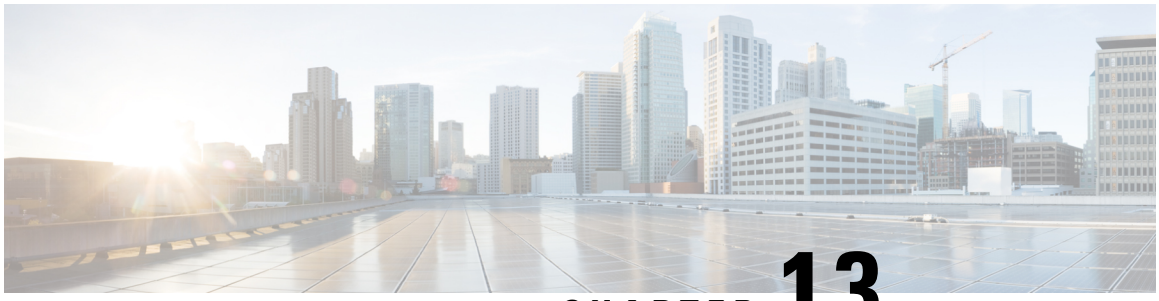
Verifying the Rollback Configuration

Use the following commands to verify the rollback configuration:

Command	Purpose
show checkpoint *name* [**all**]	Displays the contents of the checkpoint name.

Command	Purpose
show checkpoint all [**user** | **system**]	Displays the contents of all checkpoints in the current switch. You can limit the displayed checkpoints to user or system-generated checkpoints.
show checkpoint summary [**user** | **system**]	Displays a list of all checkpoints in the current switch. You can limit the displayed checkpoints to user or system-generated checkpoints.
show diff rollback-patch {**checkpoint** *src-cp-name* | **running-config** | **startup-config** | **file** *source-file*} {**checkpoint** *dest-cp-name* | **running-config** | **startup-config** | **file** *dest-file*}	Displays the differences between the source and destination checkpoint selections.
show rollback log [**exec** | **verify**]	Displays the contents of the rollback log.

Note Use the **clear checkpoint database** command to delete all checkpoint files.



CHAPTER 13

Configuring DNS

This chapter contains the following sections:

Information About DNS Client

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 hostnames 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 hostname 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 the 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 hostname 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 hostnames, 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 hostname.

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 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 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 for DNS Clients

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

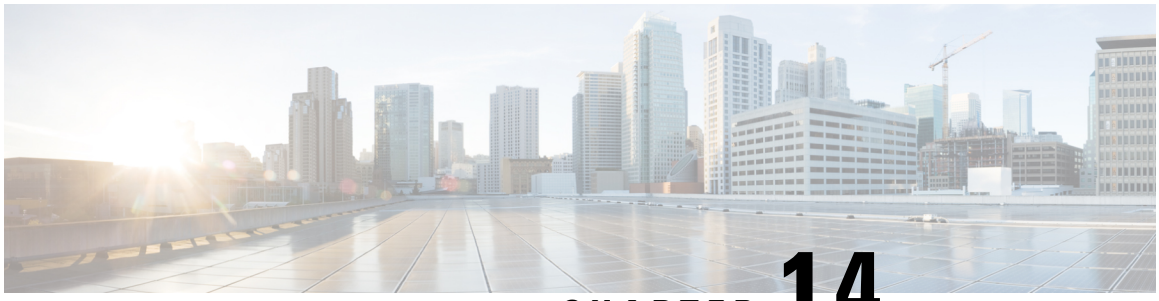
	Command or Action	Purpose
Step 1	switch# **configuration terminal**	Enters global configuration mode.
Step 2	switch(config)# vrf context managment	Specifies a configurable virtual and routing (VRF) name.
Step 3	switch(config)# **ip host** *name address1 [address2... address6]*	Defines up to six static hostname-to-address mappings in the host name cache.
Step 4	(Optional) switch(config)# **ip domain name** *name* [**use-vrf** *vrf-name*]	Defines the default domain name server that Cisco NX-OS uses to complete unqualified hostnames. 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	(Optional) switch(config)# **ip domain-list** *name* [**use-vrf** *vrf-name*]	Defines additional domain name servers that Cisco NX-OS can use to complete unqualified hostnames. 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 uses each entry in the domain list to append that domain name to any hostname 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	(Optional) switch(config)# **ip name-server** *server-address1 [server-address2... server-address6]* **[use-vrf** *vrf-name*]	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	(Optional) switch(config)# **ip domain-lookup**	Enables DNS-based address translation. This feature is enabled by default.
Step 8	(Optional) switch(config)# **show hosts**	Displays information about DNS.

	Command or Action	Purpose
Step 9	switch(config)# **exit**	Exits configuration mode and returns to EXEC mode.
Step 10	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

The following 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 14

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 device 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 device 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 device 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, and 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. This level is not supported for SNMPv3.
- 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 25: 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	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 nonmaliciously.
- Message origin authentication—Confirms that the claimed identity of the user who received the data was originated.
- 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 and 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 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 a 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 username 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

Cisco NX-OS does not support the SNMPv3 noAuthNoPriv security level.

Default SNMP Settings

Table 26: Default SNMP Parameters

Parameters	Default
license notifications	Enabled
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	(Optional) switch# **show snmp user** **Example:** `switch(config) # show snmp user`	Displays information about one or more SNMP users.
Step 4	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

The following example shows how to configure 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 that uses a 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 who belong 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.

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 about creating ACLs, see the NX-OS security configuration guide for the Cisco Nexus Series software that you are using.

Use the following command in global configuration mode to assign an IPv4 or IPv6 ACL to an SNMPv3 community to filter SNMP requests:

Command	Purpose
switch(config)# **snmp-server community** *name* [**use-ipv4acl** *ipv4acl-name*] [**use-ipv6acl** *ipv6acl-name*] `switch(config)# snmp-server community public use-ipv4acl myacl`	Assigns an IPv4 or IPv6 ACL to an SNMPv3 community to filter SNMP requests.

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 device 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.

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

Example

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

Example

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 a Source Interface for Sending Out All SNMP Notifications

You can configure SNMP to use the IP address of an interface as the source IP address for notifications. When a notification is generated, its source IP address is based on the IP address of this configured interface.

Complete the following steps to configure a source interface for sending out all SNMP notifications:

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 source-interface** {**traps** | **informs**} *if-type if-number* **Example:** `switch(config) # snmp-server source-interface traps ethernet 2/1`	Configures a source interface for sending out SNMPv2c traps or informs. Use **?** to determine the supported interface types.

Example

This example shows how to configure a source interface to sending out SNMPv2c traps:

```
switch# configure terminal
switch(config) # snmp-server source-interface traps ethernet 2/1
```

What to do next

To display information about configured source interfaces, enter the **show snmp source-interface** command.

Configuring a Host Receiver for SNMP Notifications

Note This configuration overrides the global source interface configuration.

Complete the following steps to configure a host receiver on a source interface responsible for receiving all SNMP 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	switch(config) # **snmp-server host** *ip-address* **source-interface** *if-type if-number* [**udp_port** *number*] **Example:** `switch(config) # snmp-server host 192.0.2.1 source-interface traps ethernet 2/1`	Configures a host receiver for SNMPv2c traps or informs. The *ip-address* can be an IPv4 or IPv6 address. Use **?** to determine the supported interface types.

Example

To the following example configures a source interface responsible for receiving all SNMP notifications:

```
switch# config t
switch(config) # snmp-server host 192.0.2.1 source-interface ethernet 2/1
```

What to do next

To display information about configured source interface, enter the **show snmp source-interface** command.

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 that 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. **Note** By default, SNMP sends the traps using the management VRF. If you do not want to use the management VRF, you must use this command to specify the desired VRF.

	Command or Action	Purpose
Step 3	switch(config)# **snmp-server community** *community-name* **group** *group-name*	Maps an SNMPv2c community to an SNMP context and identifies the group to which 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.

Example

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 27: Enabling SNMP Notifications

MIB	Related Commands
All notifications	**snmp-server enable traps**

MIB	Related Commands
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**
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**

MIB	Related Commands
CISCO-CONFIG-MAN-MIB **Note** Supports no MIB objects except the following notification: ccmCLIRunningConfigChanged	**snmp-server enable traps config**

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.
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:

- cieLinkDown—Enables the Cisco extended link state down notification.
- cieLinkUp—Enables the Cisco extended link state up notification.
- cisco-xcvr-mon-status-chg—Enables the Cisco interface transceiver monitor status change notification.
- delayed-link-state-change—Enables the delayed link state change.
- extended-linkUp—Enables the Internet Engineering Task Force (IETF) extended link state up notification.
- extended-linkDown—Enables the IETF extended link state down notification.
- linkDown—Enables the IETF Link state down notification.
- linkUp—Enables the IETF Link state up notification.

Procedure

	Command or Action	Purpose

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

Step 2	**snmp-server enable traps link** [**cieLinkDown** | **cieLinkUp** | **cisco-xcvr-mon-status-chg** | **delayed-link-state-change**] | **extended-linkUp** | **extended-linkDown** | **linkDown** | **linkUp**] **Example:** `switch(config)# snmp-server enable traps link cieLinkDown`	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 these limit notifications on a flapping interface (an interface that transitions between up and down repeatedly).

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **interface** *type slot*/*port*	Specifies the interface to be changed. **Note** If this is a 10G breakout port, the *slot*/*port* syntax is *QSFP-module*/*port*. **Note** If this is a QSFP+ GEM or a breakout port, the *port* syntax is *QSFP-module*/*port*.
Step 3	switch(config -if)# **no snmp trap link-status**	Disables SNMP link-state traps for the interface. This feature is 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. This feature is disabled by default.

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 global 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	(Optional) switch# **show snmp**	Displays information about one or more destination profiles.
Step 5	(Optional) switch# **copy running-config startup-config**	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 global 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	(Optional) switch(config)# **no snmp-server context** *context-name* [**instance** *instance-name*] [**vrf** *vrf-name*] [**topology** *topology-name*]	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.

Modifying the AAA Synchronization Time

You can modify how long Cisco NX-OS holds the synchronized user configuration.

Procedure

	Command or Action	Purpose

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

Step 2	**snmp-server aaa-user cache-timeout** *seconds* **Example:** `switch(config)# snmp-server aaa-user cache-timeout 1200`	Configures how long the AAA synchronized user configuration stays in the local cache. The range is from 1 to 86400 seconds. The default value is 3600 seconds.
Step 3	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

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 the SNMP Configuration

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

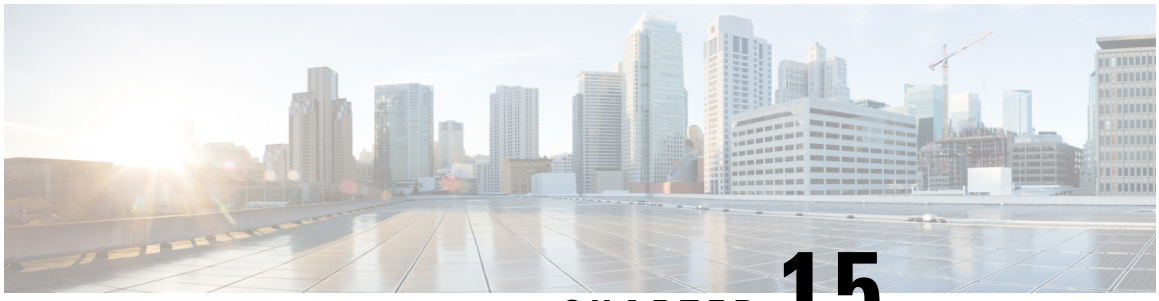
Command	Purpose
show snmp	Displays the SNMP status.

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

Additional References

MIBs

MIBs	MIBs Link
MIBs related to SNMP	To locate and download supported MIBs, go to the follow https://cisco.github.io/cisco-mibs/supportlists/nexus5000/Nexus5000MIBSupportList.html



CHAPTER 15

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 device.

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 devices. 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 device uses to collect a sample value of the MIB object.
- 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 device triggers a rising alarm or resets a falling alarm.
- Falling threshold—The value at which theCisco Nexus device triggers a falling alarm or resets a rising alarm.
- Events—The action that the Cisco Nexus device 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 does 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 and a 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 eventnumber 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.

Before you begin

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

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global 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	(Optional) switch# **show rmon** {**alarms** | **hcalarms**}	Displays information about RMON alarms or high-capacity alarms.
Step 5	(Optional) switch# **copy running-config startup-config**	Saves this configuration change.

Example

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.

Before you begin

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

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global 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	(Optional) switch(config)# **show rmon** {**alarms** | **hcalarms**}	Displays information about RMON alarms or high-capacity alarms.
Step 4	(Optional) switch# **copy running-config startup-config**	Saves this configuration change.

Verifying the RMON Configuration

Use the following commands to verify the RMON configuration information:

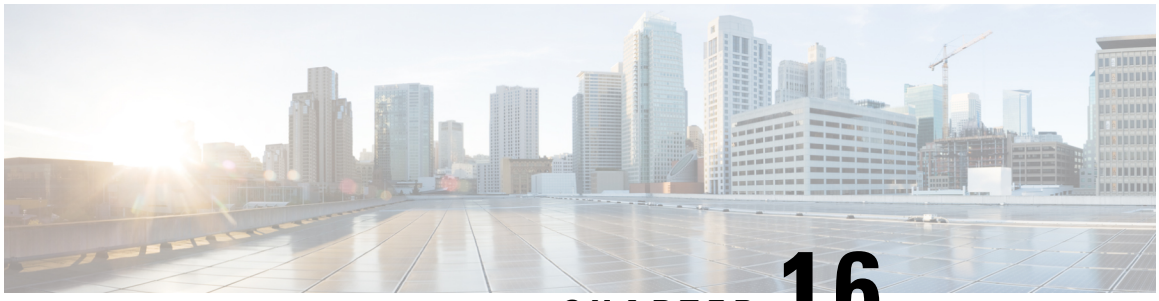
Command	Purpose
show rmon alarms	Displays information about RMON alarms.
show rmon events	Displays information about RMON events.
show rmon hcalarms	Displays information about RMON hcalarms.
show rmon logs	Displays information about RMON logs.

Default RMON Settings

The following table lists the default settings for RMON parameters.

Table 28: Default RMON Parameters

Parameters	Default
Alarms	None configured.
Events	None configured.



CHAPTER 16

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, a Fibre Channel Analyzer, or other Remote Monitoring (RMON) probes.

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 device supports Ethernet, port channels, SAN port channels, VSANs and VLANs as SPAN sources. With VLANs or VSANs, all supported interfaces in the specified VLAN or VSAN are included as SPAN sources. You can choose the SPAN traffic in the ingress direction, the egress direction, or both directions for Ethernet and virtual Fibre Channel 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.

Note Fibre Channel ports and VSAN ports cannot be configured as ingress source ports in a SPAN session.

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 and Fibre Channel interfaces as SPAN destinations.

Starting with Cisco NX-OS Release 7.2(0)N1(1), HIF and virtual ethernet (Veth) ports as SPAN destination is supported.

Source SPAN	Dest SPAN
Ethernet	Ethernet

Source SPAN	Dest SPAN
Fibre Channel	Fibre Channel
Fibre Channel	Ethernet (FCoE)
Virtual Fibre Channel	Fibre Channel
Virtual Fibre Channel	Ethernet (FCoE)

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, FCoE, and Fibre Channel ports cannot be destination ports.
- Can be any physical port. Source Ethernet and FCoE 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.
- The FEX interface cannot be a span destination.
- The same destination interface cannot be used for multiple SPAN sessions. However, an interface can act as a destination for a SPAN and an ERSPAN session.

SPAN with ACL

The SPAN with ACL filtering feature allows you to filter SPAN traffic so that you can reduce bandwidth congestion. To configure SPAN with ACL filtering, you use ACL's for the session to filter out traffic that you do not want to span. An ACL is a list of permissions associated to any entity in the system; in the context of a monitoring session, an ACL is a list of rules which results in spanning only the traffic that matches the ACL criteria, saving bandwidth for more meaningful data. The filter can apply to all sources in the session.

Guidelines and Limitations for SPAN

- The **switchport monitor rate-limit interface** command is not applicable on the Nexus 5500 device. The rate limit for SPAN traffic takes place at the SPAN source port on a Nexus 5500 device. Also, to avoid impacting monitored production traffic:

- SPAN is rate-limited to 5 Gbps for every 8 ports (one ASIC).
- RX-SPAN is rate-limited to 0.71 Gbps per port when the RX-traffic on the port exceeds 5 Gbps.

- The switch supports four active SPAN sessions. When you configure more than two SPAN sessions, the first two sessions are active. During startup, the order of active sessions is reversed; the last two sessions are active. For example, if you configured ten sessions 1 to 10 where 1 and 2 are active, after a reboot, sessions 9 and 10 will be active. To enable deterministic behavior, explicitly suspend sessions 3 to 10 with the **monitor session** *session-number* **shut** command.
- Starting from Cisco NX-OS Release 7.3(0)N1(1), a host interface (HIF) port can be a destination port for local SPAN sessions. However, a HIF port cannot be a destination port for SPAN-on-Latency, SPAN-on-Drop and ERSPAN sessions.
- An interface cannot be added as a source interface in the same direction in more than one SPAN session.
- Connecting SPAN destination ports to a switch device is not supported.
- SPAN is not supported on a management interface.
- Some protocols such as LLDP, DCBX, LACP, CDP are offloaded to FEX CPU. Hence the parent switch never sees native frames for these protocols and uses MTS messaging to inform the parent CPU.

  Moreover, since SPAN is done on the parent fabric interface, native packets for the protocols that are handled by FEX CPU are not seen in the SPAN.

The following limitations apply to SPAN (local SPAN) session Access Control Lists (ACL) configurations:

- Due to system limitations, the extent to which an ACL associated to SPAN session can scale depends on the how the SPAN source is configured. The following table shows different scenarios and the corresponding maximum ACL size supported.

Note These calculations assume that each ACE in the ACL results in one final TCAM entry.

Scenario	Maximum ACL Size
SPAN has single Switch Port as source with both Tx and Rx.	Current Available TCAM Entries/2
SPAN has multiple Switch Ports as source with both Tx and Rx.	Current Available TCAM Entries/3
SPAN has Port Channel (with one or more member switch ports) as source with both Tx and Rx.	Current Available TCAM Entries/3
SPAN has single HIF Ports as source with both Tx and Rx.	Current Available TCAM Entries/3
SPAN has multiple HIF Ports as source with both Tx and Rx.	Current Available TCAM Entries/4
SPAN has HIF Port Channel (with one or more member HIF ports) as source with both Tx and Rx.	Current Available TCAM Entries/4

- The following scenarios are unaffected by any system limitations for ACL and SPAN session scaling:
  - SPAN has single Switch Port as source with Tx only.
  - SPAN has multiple Switch Ports as source with Tx only.
  - SPAN has a Port Channel (with one or more member switch ports) as source with Tx only.
  - SPAN has a single Host Interface (HIF) Port as source with Tx only.
  - SPAN has multiple HIF Ports as source with Tx only.
  - SPAN has a single Port HIF Channel (with one or more member HIF ports) as source with Tx only.
  - SPAN has a single Switch Port as source with Rx only.
  - SPAN has multiple Switch Ports as source with Rx only.
  - SPAN has a Port Channel (with one or more member switch ports) as source with Rx only.
  - SPAN has a single HIF Ports as source with with Rx only.
  - SPAN has multiple HIF Ports as source with Rx only.
  - SPAN has a HIF Port Channel (with one or more member HIF ports) as source with Rx only
- The following guidelines apply when configuring local SPAN sessions with ACLs:
  - When you associate an ACL with a SPAN session, you must ensure that its size is not greater than the calculations given in the table above. Otherwise the SPAN session fails and generate a "TCAM resource unavailable" error. If the ACL has Layer 4 Operations and TCAM resource expansion is enabled, you need to know the expected expanded size and you need to use the expanded size to calculate the maximum ACL size.
  - If you change the ACL that is attached to a SPAN session, the ACL size can exceed the maximum ACL size allowed. In this scenario, the SPAN session continues to work with the modified ACL. However, you should undo the ACEs added to the ACL to limit the size to maximum allowed ACL size.
  - If you add a SPAN session when one already exists, then to modify the first span session there should be free TCAM entries of size equal to number of ACEs in the associated ACL (Assuming that each ACE requires one TCAM entry. If it gets expanded, the expanded size should be considered). Therefore, TCAM entries consumed by the second SPAN session should be released.
  - To replace a large ACL with another large ACL (which could cause the SPAN session to enter a generic error state), you must first remove the existing filter access group (using the **no filter access-group** *current acl name* command), and then configure the new filter access group (using the **filter access-group** *new acl name* command).
- Local SPAN/SPAN on Drop/SPAN on Latency is not aware of VPC.
- The following is the limitation for HIF and Virtual Ethernet (Veth) as SPAN destination:
  - Multi-destination SPAN is not supported. If HIF/VETH port is a destination, the monitor session must have single destination.

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.

Example

The following 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. **Note** If this is a 10G breakout port, the *slot*/*port* syntax is *QSFP-module*/*port*. **Note** If this is a QSFP+ GEM or a breakout port, the *port* syntax is *QSFP-module*/*port*.

	Command or Action	Purpose
		Note To enable the **switchport monitor** command on virtual ethernet ports, you can use the **interface vethernet** *slot*/*port* command.
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. **Note** If this is a 10G breakout port, the *slot*/*port* syntax is *QSFP-module*/*port*. **Note** If this is a QSFP+ GEM or a breakout port, the *port* syntax is *QSFP-module*/*port*. **Note** To enable the virtual ethernet port as destination interface in the monitor configuration, you can use the **destination interface vethernet** *slot*/*port* command.

Example

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

```
switch# configure terminal
switch(config)# interface ethernet100/1/24
switch(config-if)# switchport monitor
switch(config-if)# exit
switch(config)# monitor session 1
switch(config-monitor)# destination interface ethernet100/1/24
switch(config-monitor)#
```

The following example shows how to configure a virtual ethernet (VETH) SPAN destination port:

```
switch# configure terminal
switch(config)# interface vethernet10
switch(config-if)# switchport monitor
switch(config-if)# exit
switch(config)# monitor session 2
switch(config-monitor)# destination interface vethernet10
switch(config-monitor)#
```

Configuring MTU Truncation for Each SPAN Session

To reduce the SPAN traffic bandwidth, you can configure the maximum bytes allowed for each replicated packet in a SPAN session. This value is called the maximum transmission unit (MTU) truncation size. Any SPAN packet larger than the configured size is truncated to the configured size.

Note MTU Truncation is not supported for SPAN-on-Drop sessions.

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 and specifies the SPAN session for which the MTU truncation size is to be configured.
Step 3	switch(config-monitor) # [**no**] **mtu**	Configures the MTU truncation size for packets in the specified SPAN session. The range is from 64 to 1518 bytes.
Step 4	(Optional) switch(config-monitor) # **show monitor session** *session-number*	Displays the status of SPAN sessions, including the configuration status of MTU truncation, the maximum bytes allowed for each packet per session, and the modules on which MTU truncation is and is not supported.
Step 5	(Optional) switch(config-monitor) # **copy running-config** startup-config	Copies the running configuration to the startup configuration.

Example

This example shows how to configure MTU truncation for a SPAN session:

```
switch# configure terminal
switch(config) # monitor session 3
switch(config-monitor) # mtu
switch(config-monitor) # copy running-config startup-config
switch(config-monitor) #
```

Configuring Fibre Channel Destination Port

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

You can configure a Fibre Channel port as a SPAN destination port.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **interface fc** *slot*/*port*	Enters interface configuration mode for the specified Fibre Channel interface selected by the slot and port values. **Note** If this is a 10G breakout port, the *slot*/*port* syntax is *QSFP-module*/*port*. **Note** If this is a QSFP+ GEM or a breakout port, the *port* syntax is *QSFP-module*/*port*.
Step 3	switch(config-if)# **switchport mode SD**	Sets the interface to SPAN destination (SD) mode.
Step 4	switch(config-if)# **switchport speed 1000**	Sets the interface speed to 1000. The auto speed option is not allowed.
Step 5	switch(config-if)# **exit**	Reverts to global configuration mode.
Step 6	switch(config)# **monitor session** *session-number*	Enters the monitor configuration mode.
Step 7	switch(config-monitor)# **destination interface fc** *slot*/*port*	Configures the Fibre Channel destination port. **Note** If this is a 10G breakout port, the *slot*/*port* syntax is *QSFP-module*/*port*. **Note** If this is a QSFP+ GEM or a breakout port, the *port* syntax is *QSFP-module*/*port*.

Example

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

```
switch# configure terminal
switch(config)# interface fc 2/4
switch(config-if)# switchport mode SD
switch(config-if)# switchport speed 1000
switch(config-if)# exit
switch(config)# monitor session 2
switch(config-monitor)# destination interface fc 2/4
```

Configuring Source Ports

A source port can be an Ethernet port, port channel, Fiber Channel ports, SAN port channel, VLAN, or a VSAN port. It cannot be a destination port.

Source ports can only be Ethernet ports.

A source port can be an Ethernet port, port channel, Fiber Channel port, SAN port channel, VLAN, or a VSAN port. It cannot be a destination port.

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**]	Adds an Ethernet SPAN source port and specifies the traffic direction in which to duplicate packets. You can enter a range of Ethernet ports. You can specify the traffic direction to duplicate as ingress (Rx), egress (Tx), or both. By default, the direction is both. **Note** If this is a 10G breakout port, the *slot*/*port* syntax is *QSFP-module*/*port*. **Note** If this is a QSFP+ GEM or a breakout port, the *port* syntax is *QSFP-module*/*port*.

Example

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)#
```

The following example shows how to configure a Fibre Channel SPAN source port:

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

Example

The following 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)#
```

The following 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 a descriptive name for the SPAN session.

Example

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) #
```

Configuring an ACL Filter for a SPAN Session

To selectively monitor traffic in a SPAN session, you can configure an access-control list (ACL) to filter packets. The SPAN session ignores any permit or deny actions specified in the access-list, and spans only the packets that match the access-list filter criteria.

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 and specifies the SPAN session for which the ACL filter is to be configured.
Step 3	switch(config-monitor) # [**no**] **filter access-group** *acl_filter*	Configures the ACL filter for packets in the specified SPAN session. The ACL filter can be a MAC or an IP access-list.
Step 4	(Optional) switch(config-monitor) # **show monitor session** *session-number*	Displays the status of SPAN sessions, including the configuration status of ACL filter.
Step 5	(Optional) switch(config-monitor) # **copy running-config** startup-config	Copies the running configuration to the startup configuration.

Example

This example shows how to configure an ACL filter for a SPAN session:

```
switch# configure terminal
switch(config) # monitor session 3
switch(config-monitor) # filter access-group acl_span_ses_3
switch(config) # copy running-config startup-config
switch(config) #
```

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.

Example

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

Example

The following example shows how to suspend a SPAN session:

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

Troubleshooting SPAN session issues

If a SPAN session is down, do the following:

- Check if one of the destination port is operational by performing the following:
  - Use the **show running interface** *interface* command and check if the switchport monitor is configured.
  - Use the **show interface** *interface* command and check if the destination interface shows the status as "admin up".

- Use the **show interface** *interface* command to check if one of the source port is operational and if the source interface shows the status as "admin up".
- If ACL filter is applied, check if the filter definition exists. Use the **show access-lists** *listname* command to check the configured access list with entries

Troubleshooting SPAN session with large number of source ports issues

Table 29: Troubleshooting SPAN session with large number of source ports

Problem Description	Solution	Recommendation
When a SPAN session is configured with maximum supported range of 128 source ports at one go, the configuration session may encounter "Service not responding" message.	Remove the ports and configure them in smaller ranges (example, 1 to 48) and then use the **shutdown** and **no shutdown** command on the session.	Configure the individual ports in small ranges (example, 1 to 48).
After using the **shutdown** and then **no shutdown** on a range of SPAN session configured with maximum of ports (example, 128), some sessions do not come up.	Remove some ports from the specific SPAN session. Add the removed ports back to the same SPAN session and then use the **no shutdown** command.	Use the **shutdown** command on each port.
After creating a SPAN session with 128 source ports, the **no shutdown** command displays a "Service not responding" message.	Use the **no shutdown** command repeatedly to bring up the SPAN session.	

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.

Example

The following 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
```

The following example shows how to display SPAN session details:

```
switch# show monitor session 2
   session 2
---------------
type              : local
state             : up
acl-name          : acl1
source intf       :
    rx            : fc3/1
    tx            : fc3/1
    both          : fc3/1
source VLANs      :
    rx            :

destination ports : Eth3/1
```

This example shows details for a SPAN session with multiple destination ports:

```
switch(config-monitor)# show monitor session 5
   session 5
---------------
type              : local
state             : up
source intf       :
    rx            : Eth1/1
    tx            : Eth1/1
    both          : Eth1/1
source VLANs      :
    rx            :
source VSANs      :
    rx            :
destination ports : Eth1/8, Eth1/9
```

This example shows details for a SPAN-on-Drop session:

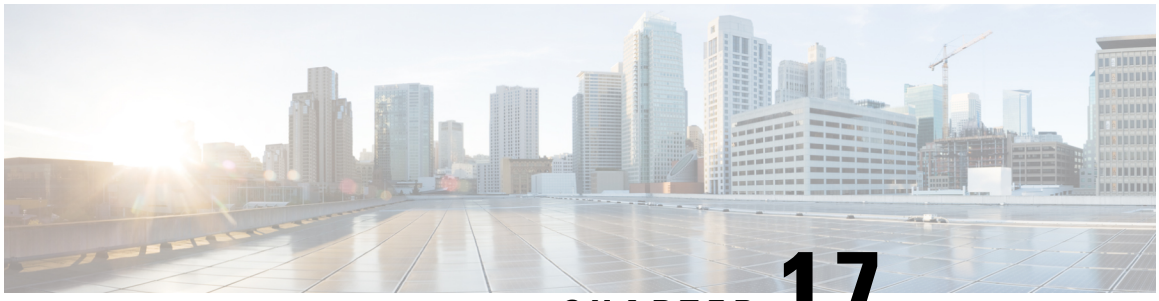
```
switch(config-monitor)# show monitor session 48
  session 48
---------------
description       : span-on-drop-session
```

```
type              : span-on-drop
state             : up
mtu               : 0
source ports      : Eth1/2
destination ports : Eth1/3
```

Configuration Example for a SPAN ACL

This example shows how to configure a SPAN ACL:

```
switch# configure terminal
switch(config)# ip access-list match_11_pkts
switch(config-acl)# permit ip 11.0.0.0 0.255.255.255 any
switch(config-acl)# exit
switch(config)# monitor session 1
switch(config-erspan-src)# filter access-group match_11_pkts
```



CHAPTER 17

Configuring ERSPAN

This chapter contains the following sections:

Information About ERSPAN

The Cisco NX-OS system supports the Encapsulated Remote Switching Port Analyzer (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 can separately configure ERSPAN source sessions and destination sessions on different switches.

ERSPAN Source Sessions

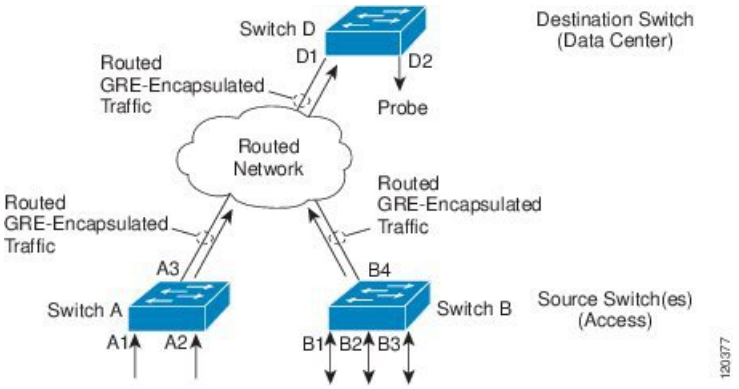
An ERSPAN source session is defined by the following:

- A session ID.
- A list of source ports, source VLANs, or source VSANs to be monitored by the session.
- An ERSPAN flow ID.
- Optional attributes related to the GRE envelope such as IP TOS and TTL.
- Destination IP address.
- Virtual Routing and Forwarding tables.

ERSPAN source sessions do not copy ERSPAN GRE-encapsulated traffic from source ports. Each ERSPAN source session can have ports, VLANs, or VSANs as sources. However, there are some limitations. For more information, see Guidelines and Limitations for ERSPAN.

The following figure shows an example ERSPAN configuration.

Figure 1: ERSPAN Configuration



Monitored Traffic

By default, ERSPAN monitors all traffic, including multicast and bridge protocol data unit (BPDU) frames.

The direction of the traffic that ERSPAN monitors depends on the source, as follows:

- For a source port, the ERSPAN can monitor ingress, egress, or both ingress and egress traffic.
- For a source VLAN or source VSAN, the ERSPAN can monitor only ingress traffic.

ERSPAN Types

Cisco NX-OS Release 6.1 and later releases support ERSPAN Type II (default) and Type III. All previous Cisco NX-OS releases support only ERSPAN Type II.

ERSPAN Type III supports all of the ERSPAN Type II features and functionality and adds these enhancements:

- Provides timestamp information in the ERSPAN Type III header that can be used to calculate packet latency among edge, aggregate, and core switches.
- Identifies possible traffic sources using the ERSPAN Type III header fields.
- Provides the ability to configure timestamp granularity to determine how the clock manager synchronizes the ERSPAN timers.
- Beginning with Cisco NX-OS Release 7.1(1)N1(1), ERSPAN Type III provides configurable switch IDs that can be used to identify traffic flows across multiple switches.

Table 30: Differences between ERSPAN Type II and ERSPAN Type III

Attribute	Type II	Type III
Timestamp	NA	Timestamp provided.

Attribute	Type II	Type III
Platform-specific info	NA	Platform-specific info is required for Nexus 5500, Nexus 5600 and Nexus 6000 platforms.
Source Port Identification at Termination Switch	Limited identification.	Detailed identification. Provision of switch IDs.

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.

Truncated ERSPAN

Truncated ERSPAN can be used to reduce the amount of fabric or network bandwidth used in sending ERSPAN packets.

The default is no truncation so switches or routers receiving large ERSPAN packets might drop these oversized packets.

Note Do not enable the truncated ERSPAN feature if the destination ERSPAN router is a Cisco Nexus 6001 or Cisco Nexus 6004 switch because the Cisco Nexus 6000 Series switch drops these truncated packets.

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/licensing/guide/nx-os_lic.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. For more information, see the Interfaces configuration guide for your platform.

Guidelines and Limitations for ERSPAN

ERSPAN has the following guidelines and limitations:

- A maximum of 16 active sessions are supported. They can be ERSPAN sessions only or a mixture of ERSPAN and SPAN sessions.
- The maximum number of ports for each ERSPAN session is 128.
- The maximum number of VLANs per session is 32.
- You can have source ports, source VLANs, and source VSANs in one ERSPAN session.
- ERSPAN can monitor ingress, egress, or both ingress and egress traffic on a source port and only ingress traffic on source VLANs or source VSANs as long as the VLAN is not mapped to a VSAN. ERSPAN cannot monitor egress traffic on source VLANs and VSANs.
- To bring up an ERSPAN monitor session, you must first configure a global origin address using the **monitor erspan origin ip-address** *ip-address* **global** command.
- Source ports and source VLANs can be in the same ERSPAN session.
- ERSPAN traffic can exit the switch through a Layer 2 interface, Layer 3 interface, port channel, or FabricPath core port.
- A destination IP address of a remote switch cannot be reached through a virtual Ethernet port or FEX port. This functionality is not supported.
- Configuring a FEX interface as a destination port for ERSPAN is not supported.
- ERSPAN traffic is not load balanced if the reachability to a destination IP address is a Layer 3 ECMP or a port channel. In the case of ECMP, the ERSPAN traffic is sent to only one next-hop router or one member of the port channel.

- ERSPAN supports Fast Ethernet, Gigabit Ethernet, TenGigabit Ethernet, and port channel interfaces as source ports for a source session.
- When a session is configured through the ERSPAN configuration commands, the session ID and the session type cannot be changed. In order to change them, you must first use the no version of the configuration command to remove the session and then reconfigure the session.
- ERSPAN traffic might compete with regular data traffic.
- ERSPAN traffic is assigned to the QoS class-default system class (qos-group 0).
- To ensure that data traffic is prioritized over ERSPAN traffic, you can create a QoS system class with prioritization above the class-default system class on the ERSPAN destination port.

  On Layer 3 networks, ERSPAN traffic can be marked with a the desired Differentiated Services Code Point (DSCP) value using the ip dscp command. By default, ERSPAN traffic is marked with a DSCP value of 0.
- The **rate limit** command is not supported.
- ERSPAN is not supported on a management interface.
- You cannot use the same source interface in multiple SPAN or ERSPAN sessions.

The following limitations apply to ERSPAN source sessions Access Control Lists (ACL) configurations:

- The SPAN session ignores any permit or deny actions specified in the access-list, and spans only the packets that match the access-list filter criteria.
- ACLs are supported on ERSPAN source sessions only. ACLs are not supported on ERSPAN destination sessions.
- Due to system limitations, the extent to which an ACL associated to ERSPAN session can scale depends on the how the SPAN source is configured. The following table shows different scenarios and the corresponding maximum ACL size supported.

Note These calculations assume that each ACE in the ACL results in one final TCAM entry.

Scenario	Maximum ACL Size
ERSPAN has single Switch Port as source with both Tx and Rx.	Current Available TCAM Entries/2
ERSPAN has multiple Switch Ports as source with both Tx and Rx.	Current Available TCAM Entries/3
ERSPAN has Port Channel (with one or more member switch ports) as source with both Tx and Rx.	Current Available TCAM Entries/3
ERSPAN has single HIF Ports as source with both Tx and Rx.	Current Available TCAM Entries/3

Scenario	Maximum ACL Size
ERSPAN has multiple HIF Ports as source with both Tx and Rx.	Current Available TCAM Entries/4
ERSPAN has HIF Port Channel (with one or more member HIF ports) as source with both Tx and Rx.	Current Available TCAM Entries/4

- Due to system limitations, the extent to which an ACL associated to ERSPAN session can scale depends on the how the SPAN source is configured. The following table shows different scenarios and the corresponding maximum ACL size supported.

Note These calculations assume that each ACE in the ACL results in one final TCAM entry.

Scenario	Maximum ACL Size
ERSPAN has single Switch Port as source with both Tx and Rx.	Current Available TCAM Entries/2
ERSPAN has multiple Switch Ports as source with both Tx and Rx.	Current Available TCAM Entries/3
ERSPAN has Port Channel (with one or more member switch ports) as source with both Tx and Rx.	Current Available TCAM Entries/3
ERSPAN has single HIF Ports as source with both Tx and Rx.	Current Available TCAM Entries/3
ERSPAN has multiple HIF Ports as source with both Tx and Rx.	Current Available TCAM Entries/4
ERSPAN has HIF Port Channel (with one or more member HIF ports) as source with both Tx and Rx.	Current Available TCAM Entries/4

- The following scenarios are unaffected by any system limitations for ACL and SPAN session scaling:
  - ERSPAN has single Switch Port as source with Tx only.
  - ERSPAN has multiple Switch Ports as source with Tx only.
  - ERSPAN has a Port Channel (with one or more member switch ports) as source with Tx only.
  - ERSPAN has a single Host Interface (HIF) Port as source with Tx only.
  - ERSPAN has multiple HIF Ports as source with Tx only.
  - ERSPAN has a single Port HIF Channel (with one or more member HIF ports) as source with Tx only.
  - ERSPAN has a single Switch Port as source with Rx only.
  - ERSPAN has multiple Switch Ports as source with Rx only.

- ERSPAN has a Port Channel (with one or more member switch ports) as source with Rx only.
- ERSPAN has a single HIF Port as source with with Rx only.
- ERSPAN has multiple HIF Ports as source with Rx only.
- ERSPAN has a HIF Port Channel (with one or more member HIF ports) as source with Rx only

- The following guidelines apply when configuring ERSPAN source sessions with ACLs:
  - When you associate an ACL with an ERSPAN session, you must ensure that its size is not greater than the calculations given in the table above. Otherwise the ERSPAN session fails and generate a "TCAM resource unavailable" error. If the ACL has Layer 4 Operations and TCAM resource expansion is enabled, you need to know the expected expanded size and you need to use the expanded size to calculate the maximum ACL size.
  - If you change the ACL that is attached to a ERSPAN session, the ACL size can exceed the maximum ACL size allowed. In this scenario, the SPAN session continues to work with the modified ACL. However, you should undo the ACEs added to the ACL to limit the size to maximum allowed ACL size.
  - If you add a ERSPAN session when one already exists, then to modify the first span session there should be free TCAM entries of size equal to number of ACEs in the associated ACL (Assuming that each ACE requires one TCAM entry. If it gets expanded, the expanded size should be considered). Therefore, TCAM entries consumed by the second ERSPAN session should be released.
  - To replace a large ACL with another large ACL (which could cause the ERSPAN session to enter a generic error state), you must first remove the existing filter access group (using the **no filter access-group** *current acl name* command), and then configure the new filter access group (using the **filter access-group** *new acl name* command).

Guidelines and Limitations for ERSPAN Type III

ERSPAN Type III has the following guidelines and limitations:

- Only IPv4 networks are supported by ERSPAN Type III. IPv6 networks are not supported by ERSPAN Type III but IPv6 packets can be captured by ERSPAN.
- To calculate packet latency across ports, ERSPAN timestamp should be taken from the Precision Time Protocol (PTP) clock and the PTP feature must be enabled on the switch.

Default Settings for ERSPAN

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

The ERSPAN source session defines the session configuration parameters and the ports or VLANs to be monitored. This section describes how to configure an ERSPAN source session.

Procedure

	Command or Action	Purpose

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

Step 2	**monitor session** *span-session-number* **type** {**erspan-source** | **local**}
	Example:
	`switch(config)# monitor session 1 type erspan-source`
	`switch(config-erspan-src)#`
		Defines an ERSPAN source session using the session ID and the session type, and places the command in ERSPAN monitor source session configuration mode.
		The *span-session-number* argument range is from 1 to 1024. The same session number cannot be used more than once.
		The session IDs for source sessions are in the same global ID space, so each session ID is globally unique.
		The session ID (configured by the *span-session-number* argument) and the session type (configured by the **erspan-source** keyword) cannot be changed once entered. To change session ID or session type, use the **no** version of the command to remove the session and then recreate the session through the command with a new session ID or a new session type.

Step 3	(Optional) **description** *erspan_session_description* **Example:** `switch(config-erspan-src)# description source1`	Describes the ERSPAN source session. The *erspan_session_description* argument can be up to 32 characters and cannot contain special characters or spaces.
Step 4	**source interface** { **ethernet** *slot/chassis number* | **portchannel** *number* } **Example:** `switch(config-erspan-src)# source interface eth 1/1`	Associates the ERSPAN source session number with the source ports (1-255).

	Command or Action	Purpose
Step 5	**source vlan** *number* **Example:** `switch(config-erspan-src)# source vlan 1`	Associates the ERSPAN source session number with the VLANs (1-4096).
Step 6	**source vsan** *number* **Example:** `switch(config-erspan-src)# source vsan 1`	Specifies the VSAN ID number. The range is 1 to 4093.
Step 7	**destination ip** *ip-address* **Example:** `switch(config-erspan-src)# destination ip 192.0.2.2`	Configures the destination IP address in the ERSPAN session. Only one destination IP address is supported per ERSPAN source session.
Step 8	**erspan-id** *flow-id* **Example:** `switch(config-erspan-src)# erspan-id 5`	Configures the flow ID to identify the ERSPAN flow. The range is from 1 to 1023.
Step 9	**vrf** {*vrf-name* | **default** } **Example:** `switch(config-erspan-src)# vrf default`	Configures the VRF to use instead of the global routing table. You can use a VRF that you have specifically configured or the default VRF.
Step 10	[**no**] **filter access-group** *acl_filter* **Example:** `switch(config-erspan-src)# filter access-group erspan_acl_filter`	Configures the ACL filter for packets in this ERSPAN session. The ACL filter can be a MAC or an IP access-list.
Step 11	(Optional) **ip ttl** *ttl-number* **Example:** `switch(config-erspan-src)# ip ttl 5`	Configures the IP time-to-live (TTL) value of the packets in the ERSPAN traffic. Valid values are from 1 to 255. The default value is 255.
Step 12	(Optional) **ip dscp** *dscp_value* **Example:** `switch(config-erspan-src)# ip dscp 42`	Configures the IP Differentiated Services Code Point (DSCP) value of the packets in the ERSPAN traffic. Valid values are from 0 to 63. The default value is 0.
Step 13	**no shut** **Example:** `switch(config-erspan-src)# no shut`	Enables the ERSPAN source session. By default, the session is created in the shut state.

Step 14	**exit**
	Example:
	`switch(config-erspan-src)# exit`
	`switch(config)# exit`
		Updates the configuration and exits ERSPAN source session configuration mode.

	Command or Action	Purpose
Step 15	(Optional) **copy running-config startup-config** **Example:** `switch(config-erspan-src)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring an ERSPAN Type III Source Session

Procedure

	Command or Action	Purpose

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

Step 2	**monitor erspan switch-id** *switch-id* **Example:** `switch(config)# monitor erspan switch-id 1009`	Configures the ERSPAN global switch ID. The switch ID is applicable for all ERSPAN Type III sessions. Default value is 0.
Step 3	**monitor erspan granularity 1588** **Example:** `switch(config)# monitor erspan granularity 1588`	Specifies granularity for all ERSPAN Type III sessions. 1588 (in seconds or nanoseconds) is the only option available and it is the default value.

Step 4	**monitor session** *span-session-number* **type** {**erspan-source** | **local**}
	Example:
	`switch(config)# monitor session 1 type erspan-source`
	`switch(config-erspan-src)#`
		Defines an ERSPAN source session using the session ID and the session type, and places the command in ERSPAN monitor source session configuration mode.
		The *span-session-number* argument range is from 1 to 1024. The same session number cannot be used more than once.
		The session IDs for source sessions are in the same global ID space, so each session ID is globally unique for both session types.
		The session ID (configured by the *span-session-number* argument) and the session type (configured by the **erspan-source** keyword) cannot be changed once entered. To change session ID or session type, use the **no** version of the command to remove the session and then recreate the session through the

	Command or Action	Purpose
		command with a new session ID or a new session type.
Step 5	Required: **header-type** *version* **Example:** `switch(config-erspan-src)# header-type 3`	Changes the ERSPAN source session from Type II to Type III. **Note** You can use the **no** form of this command to change an ERSPAN source session from Type III to Type II.
Step 6	(Optional) **description** *erspan_session_description* **Example:** `switch(config-erspan-src)# description source1`	Describes the ERSPAN source session. The *erspan_session_description* argument can be up to 240 characters and cannot contain special characters or spaces.
Step 7	**source interface** { **ethernet** *slot/chassis number* | **portchannel** *number* } **Example:** `switch(config-erspan-src)# source interface eth 1/1`	Associates the ERSPAN source session number with the source ports (1-255).
Step 8	**source vlan** *number* **Example:** `switch(config-erspan-src)# source vlan 1`	Associates the ERSPAN source session number with the VLANs (1-4096).
Step 9	**source vsan** *number* **Example:** `switch(config-erspan-src)# source vsan 1`	On Cisco Nexus 5000 Series switches, specifies the VSAN ID number. The range is 1 to 4093. On Cisco Nexus 5500 Series switches, you cannot configure source VSANs.
Step 10	**destination ip** *ip-address* **Example:** `switch(config-erspan-src)# destination ip 192.0.2.2`	Configures the destination IP address in the ERSPAN session. Only one destination IP address is supported per ERSPAN source session.
Step 11	**erspan-id** *flow-id* **Example:** `switch(config-erspan-src)# erspan-id 5`	Configures the flow ID to identify the ERSPAN flow. The range is from 1 to 1023.
Step 12	**vrf** {*vrf-name* | **default** } **Example:** `switch(config-erspan-src)# vrf default`	Configures the VRF to use instead of the global routing table. You can use a VRF that you have specifically configured or the default VRF.

	Command or Action	Purpose
Step 13	[**no**] **filter access-group** *acl_filter* **Example:** `switch(config-erspan-src)# filter access-group erspan_acl_filter`	Configures the ACL filter for packets in this ERSPAN session. The ACL filter can be a MAC or an IP access-list.
Step 14	(Optional) **ip ttl** *ttl-number* **Example:** `switch(config-erspan-src)# ip ttl 5`	Configures the IP time-to-live (TTL) value of the packets in the ERSPAN traffic. Valid values are from 1 to 255. The default value is 255.
Step 15	(Optional) **ip dscp** *dscp_value* **Example:** `switch(config-erspan-src)# ip dscp 42`	Configures the IP Differentiated Services Code Point (DSCP) value of the packets in the ERSPAN traffic. Valid values are from 0 to 63. The default value is 0.
Step 16	**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** On Cisco Nexus 5000 Series switches, only two ERSPAN source sessions can be running simultaneously. On Cisco Nexus 5500 Series switches, up to four source sessions can be running simultaneously.

Step 17	**exit**
	Example:
	`switch(config-erspan-src)# exit`
	`switch(config)# exit`
		Updates the configuration and exits ERSPAN source session configuration mode.

Step 18	(Optional) **copy running-config startup-config** **Example:** `switch(config-erspan-src)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring Truncated ERSPAN

You can configure an MTU size for the ERSPAN traffic to reduce the amount of fabric or network bandwidth used in sending ERSPAN packets.

Procedure

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

	Command or Action	Purpose
	`switch> enable`	

Step 2	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 3	**monitor session** *erspan_session_number* **type** {**erspan-source** | **local**}
	Example:
	`switch(config)# monitor session 1 type erspan-source`
	`switch(config-erspan-src)#`
		Defines an ERSPAN source session using the session ID and the session type, and places the command in ERSPAN monitor source session configuration mode.
		The span-session-number argument range is from 1 to 1024. The same session number cannot be used more than once.
		The session IDs for source sessions are in the same global ID space, so each session ID is globally unique for both session types.
		The session ID (configured by the span-session number argument) and the session type (configured by the erspan-source keyword) cannot be changed once entered. To change session ID or session type, use the no version of the command to remove the session and then re-create the session through the command with a new session ID or a new session type.

Step 4	**mtu** *mtu-value* **Example:** `switch(config-erspan-src)# mtu 64`	Defines the maximum transmission unit (MTU) truncation size for ERSPAN packets. Valid values are from 64 to 1518. The default is no truncation enabled.
Step 5	**exit** **Example:** `switch(config-mon-erspan-src)# exit`	Updates the configuration and exits ERSPAN source session configuration mode.
Step 6	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Saves the change persistently through reboots and restarts by copying 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 a specific number of ERSPAN sessions can be running simultaneously, you can shut down a session 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 . By default, sessions are created in the shut state. **Note** • In Cisco Nexus 5000 and 5500 platforms, two sessions can run simultaneously. • In Cisco Nexus 5600 and 6000 platforms, 16 sessions can run simultaneously.
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 . By default, sessions are created in the shut state. . **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:**	Enters the monitor configuration mode for the ERSPAN destination type.

	Command or Action	Purpose
	`switch(config-erspan-src)# monitor session 3 type erspan-destination`	
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	(Optional) **show monitor session all** **Example:** `switch(config-erspan-src)# show monitor session all`	Displays the status of ERSPAN sessions.
Step 9	(Optional) **show running-config monitor** **Example:** `switch(config-erspan-src)# show running-config monitor`	Displays the running ERSPAN configuration.
Step 10	(Optional) **show startup-config monitor** **Example:** `switch(config-erspan-src)# show startup-config monitor`	Displays the ERSPAN startup configuration.
Step 11	(Optional) **copy running-config startup-config** **Example:** `switch(config-erspan-src)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Verifying the ERSPAN Configuration

Use the following command to verify the ERSPAN configuration information:

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

The following 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 erspan granularity 100_ns
switch(config-erspan-src)# header-type 3
switch(config)# monitor session 1 type erspan-source
switch(config-erspan-src)# source interface e14/30
switch(config-erspan-src)# erspan-id 1
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
```

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# monitor session 1 type erspan-source
switch(config-erspan-src)# description source1
switch(config-erspan-src)# source interface ethernet 1/1
switch(config-erspan-src)# source vlan 1
switch(config-erspan-src)# source vsan 1
switch(config-erspan-src)# destination ip 192.0.2.2
switch(config-erspan-src)# erspan-id 1
switch(config-erspan-src)# vrf default
switch(config-erspan-src)# ip ttl 5
switch(config-erspan-src)# ip dscp 5
switch(config-erspan-src)# no shut
switch(config-erspan-src)# exit
switch(config)# copy running-config startup config
```

Configuration Example for an ERSPAN Type III Source Session

The following example shows how to configure an ERSPAN Type III source session:

```
switch# config t
switch(config)# monitor erspan origin ip-address 10.0.0.1 global
switch(config)# monitor session 1 type erspan-source
switch(config-erspan-src)# header-type 3
switch(config-erspan-src)# erspan-id 1
switch(config-erspan-src)# vrf default
switch(config-erspan-src)# destination ip 10.0.0.1
switch(config-erspan-src)# source interface ethernet 1/22 both
switch(config-erspan-src)# mtu 100
switch(config-erspan-src)# no shut
switch(config-erspan-src)# exit
```

```
switch(config)# exit
switch# show monitor session all
```

Configuration Example for an IP Address as the Source for an ERSPAN Session

This example shows how to configure an IP address as the source for an ERSPAN session:

```
switch# configure terminal
switch(config)# monitor erspan origin ip-address 192.0.2.1
switch(config)#  exit
switch(config)# copy running-config startup config
```

Configuration Example for Truncated ERSPAN

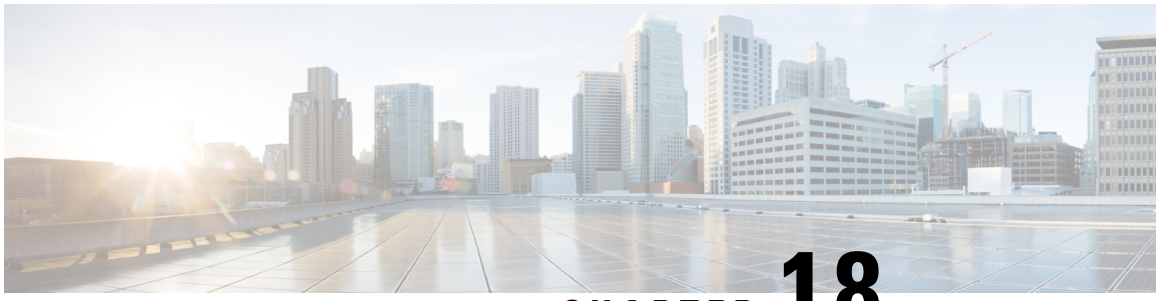
This example shows how to configure truncated ERSPAN:

```
switch# configure terminal
switch(config)# monitor session 1 type erspan-source
switch(config-erspan-src)# mtu 64
switch(config-mon-erspan-src)# exit
switch(config)# copy running-config startup config
```

Additional References

Related Documents

Related Topic	Document Title
ERSPAN commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco Nexus NX-OS System Management Command Reference* for your platform.



CHAPTER 18

Configuring NTP

This chapter contains the following sections:

Information About NTP

Information About the NTP Server

The Network Time Protocol (NTP) synchronizes the time of day among a set of distributed time servers and clients so that you can correlate events when you receive system logs and other time-specific events from multiple network devices. NTP uses the User Datagram Protocol (UDP) as its transport protocol.

All NTP communications use Coordinated Universal Time (UTC).

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

NTP uses a stratum to describe the distance between a network device and an authoritative time source:

- A stratum 1 time server is directly attached to an authoritative time source (such as a radio or atomic clock or a GPS time source).
- A stratum 2 NTP server receives its time through NTP from a stratum 1 time server.

Before synchronizing, NTP compares the time reported by several network devices and does not synchronize with one that is significantly different, even if it is a stratum 1. Because Cisco NX-OS cannot connect to a radio or atomic clock and act as a stratum 1 server, we recommend that you use the public NTP servers

available on the Internet. If the network is isolated from the Internet, Cisco NX-OS allows you to configure the time as though it were synchronized through NTP, even though it was not.

Note You can create NTP peer relationships to designate the time-serving hosts that you want your network device to consider synchronizing with and to keep accurate time if a server failure occurs.

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

NTP as Time Server

The Cisco Nexus device can use NTP to distribute time.

Other devices can configure it as a time server. You can also configure the device to act as an authoritative NTP server, enabling it to distribute time even when it is not synchronized to an outside time source.

Distributing NTP Using CFS

Cisco Fabric Services (CFS) distributes the local NTP configuration to all Cisco devices in the network.

After enabling CFS on your device, a network-wide lock is applied to NTP whenever an NTP configuration is started. After making the NTP configuration changes, you can discard or commit them.

In either case, the CFS lock is then released from the NTP application.

Clock Manager

Clocks are resources that need to be shared across different processes.

Multiple time synchronization protocols, such as NTP and Precision Time Protocol (PTP), might be running in the system.

The clock manager allows you to specify the protocol and a VDC running that protocol to control the various clocks in the system. Once you specify the protocol and VDC, the system clock starts updating.

High Availability

Stateless restarts are supported for NTP. After a reboot or a supervisor switchover, the running configuration is applied.

You can configure NTP peers to provide redundancy in case an NTP server fails.

Licensing Requirements

Product	License Requirement
Cisco NX-OS	NTP 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.

Prerequisites for NTP

NTP has the following prerequisites:

- To configure NTP, you must have connectivity to at least one server that is running NTP.

Guidelines and Limitations for NTP

NTP has the following configuration guidelines and limitations:

- The Cisco NX-OS software supports NTP version 4 (NTPv4).
- NTP server functionality is supported.
- You should have a peer association with another device only when you are sure that your clock is reliable (which means that you are a client of a reliable NTP server).
- A peer configured alone takes on the role of a server and should be used as a backup. If you have two servers, you can configure several devices to point to one server and the remaining devices to point to the other server. You can then configure a peer association between these two servers to create a more reliable NTP configuration.
- If you have only one server, you should configure all the devices as clients to that server.
- You can configure up to 64 NTP entities (servers and peers).
- If CFS is disabled for NTP, NTP does not distribute any configuration and does not accept a distribution from other devices in the network.
- After CFS distribution is enabled for NTP, the entry of an NTP configuration command locks the network for NTP configuration until a **commit** command is entered. During the lock, no changes can be made to the NTP configuration by any other device in the network except the device that initiated the lock.
- If you use CFS to distribute NTP, all devices in the network should have the same VRFs configured as you use for NTP.
- If you configure NTP in a VRF, ensure that the NTP server and peers can reach each other through the configured VRFs.

- You must manually distribute NTP authentication keys on the NTP server and Cisco NX-OS devices across the network.
- Use NTP broadcast or multicast associations when time accuracy and reliability requirements are modest, your network is localized, and the network has more than 20 clients. We recommend that you use NTP broadcast or multicast associations in networks that have limited bandwidth, system memory, or CPU resources.

Note Time accuracy is marginally reduced in NTP broadcast associations because information flows only one way.

Default Settings for NTP

The following table lists the default settings for NTP parameters:

Table 32: Default NTP Parameters

Parameters	Default
NTP	Enabled
NTP authentication	Disabled
NTP access	Enabled
NTP logging	Disabled

Configuring NTP

Enabling or Disabling NTP

Before you begin

Make sure that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# [**no**] **feature ntp**	Enables or disables NTP in VDC. NTP is enabled by default. **Note** NTP is enabled or disabled using the [**no**] **ntp enable** command.

	Command or Action	Purpose
Step 3	(Optional) switch(config)# **show ntp status**	Displays the status of the NTP application.
Step 4	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to disable NTP:

```
switch# configure terminal
switch(config)# no feature ntp
```

Configuring the Device as an Authoritative NTP Server

You can configure the device to act as an authoritative NTP server, enabling it to distribute time even when it is not synchronized to an existing time server.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	[**no**] **ntp master** [*stratum*]	Configures the device as an authoritative NTP server. You can specify a different stratum level from which NTP clients get their time synchronized. The range is from 1 to 15.
Step 3	(Optional) **show running-config ntp**	Displays the NTP configuration.
Step 4	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to configure the Cisco NX-OS device as an authoritative NTP server with a different stratum level:

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# ntp master 5
```

Configuring an NTP Server and Peer

You can configure an NTP server and peer.

Before you begin

Make sure that you know the IP address or DNS names of your NTP server and its peers.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# [**no**] **ntp server** {*ip-address* | *ipv6-address* | *dns-name*} [**key** *key-id*] [**maxpoll** *max-poll*] [**minpoll** *min-poll*] [**prefer**] [**use-vrf** *vrf-name*]	Forms an association with a server. Use the **key** keyword to configure a key to be used while communicating with the NTP server. The range for the *key-id* argument is from 1 to 65535. Use the **maxpoll** and **minpoll** keywords to configure the maximum and minimum intervals in which to poll a peer. The range for the *max-poll* and *min-poll* arguments is from 4 to 16 seconds, and the default values are 6 and 4, respectively. Use the **prefer keyword** to make this the preferred NTP server for the device. Use the **use-vrf** keyword to configure the NTP server to communicate over the specified VRF. The *vrf-name* argument can be default, management, or any case-sensitive alphanumeric string up to 32 characters. **Note** If you configure a key to be used while communicating with the NTP server, make sure that the key exists as a trusted key on the device.
Step 3	switch(config)# [**no**] **ntp peer** {*ip-address* | *ipv6-address* | *dns-name*} [**key** *key-id*] [**maxpoll** *max-poll*] [**minpoll** *min-poll*] [**prefer**] [**use-vrf** *vrf-name*]	Forms an association with a peer. You can specify multiple peer associations. Use the **key** keyword to configure a key to be used while communicating with the NTP peer. The range for the *key-id* argument is from 1 to 65535. Use the **maxpoll** and **minpoll** keywords to configure the maximum and minimum intervals in which to poll a peer. The range for the

	Command or Action	Purpose
		max-poll and *min-poll* arguments is from 4 to 17 seconds, and the default values are 6 and 4, respectively. Use the **prefer** keyword to make this the preferred NTP peer for the device. Use the **use-vrf** keyword to configure the NTP peer to communicate over the specified VRF. The *vrf-name* argument can be **default**, **management**, or any case-sensitive alphanumeric string up to 32 characters.
Step 4	(Optional) switch(config)# **show ntp peers**	Displays the configured server and peers. **Note** A domain name is resolved only when you have a DNS server configured.
Step 5	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# ntp server 2400::1 use-vrf default
switch(config)# show ntp peers
-------------------------------------------------
Peer IP Address Serv/Peer
-------------------------------------------------
2400::1   Server(configured)
switch(config)# copy running-config startup-config
[########################################] 100%

switch(config)#show ntp peer-status
Total Peers: 1
* - selected for sync, + - peer mode(active),
--peer mode(passive), = -polled in client mode
remote    local  st  poll  reach  delay    vrf
-------------------------------------------------
*2400::1  ::     9   16    37     0.00122  default
```

Configuring NTP Authentication

You can configure the device to authenticate the time sources to which the local clock is synchronized. When you enable NTP authentication, the device synchronizes to a time source only if the source carries one of the authentication keys specified by the **ntp trusted-key** command. The device drops any packets that fail the authentication check and prevents them from updating the local clock. NTP authentication is disabled by default.

Before you begin

Authentication for NTP servers and NTP peers is configured on a per-association basis using the key keyword on each **ntp server** and **ntp peer** command. Make sure that you configured all NTP server and peer associations with the authentication keys that you plan to specify in this procedure. Any **ntp server** or **ntp peer** commands that do not specify the key keyword will continue to operate without authentication.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# [**no**] **ntp authentication-key** *number* **md5** *md5-string*	Defines the authentication keys. The device does not synchronize to a time source unless the source has one of these authentication keys and the key number is specified by the **ntp trusted-key** *number* command. Cisco NX-OS supports up to 15 alphanumeric characters for the MD5 string.
Step 3	(Optional) switch(config)# **show ntp authentication-keys**	Displays the configured NTP authentication keys.
Step 4	switch(config)# [**no**] **ntp trusted-key** *number*	Specifies one or more keys (defined in Step 2) that a time source must provide in its NTP packets in order for the device to synchronize to it. The range for trusted keys is from 1 to 65535. This command provides protection against accidentally synchronizing the device to a time source that is not trusted.
Step 5	(Optional) switch(config)# **show ntp trusted-keys**	Displays the configured NTP trusted keys.
Step 6	switch(config)# [**no**] **ntp authenticate**	Enables or disables the NTP authentication feature. NTP authentication is disabled by default.
Step 7	(Optional) switch(config)# **show ntp authentication-status**	Displays the status of NTP authentication.
Step 8	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to configure the device to synchronize only to time sources that provide authentication key 42 in their NTP packets:

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# ntp authentication-key 42 md5 aNiceKey
switch(config)# ntp server 10.1.1.1 key 42
switch(config)# ntp trusted-key 42
switch(config)# ntp authenticate
switch(config)# copy running-config startup-config
[########################################] 100%
switch(config)#
```

Configuring NTP Access Restrictions

You can control access to NTP services by using access groups. Specifically, you can specify the types of requests that the device allows and the servers from which it accepts responses.

If you do not configure any access groups, NTP access is granted to all devices. If you configure any access groups, NTP access is granted only to the remote device whose source IP address passes the access list criteria.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# [**no**] **ntp access-group** {**match-all** | {{**peer** | **serve** | **serve-only** | **query-only** } *access-list-name*}}	Creates or removes an access group to control NTP access and applies a basic IP access list. The access group options are scanned in the following order, from least restrictive to most restrictive. However, if NTP matches a deny ACL rule in a configured peer, ACL processing stops and does not continue to the next access group option. • The **peer** keyword enables the device to receive time requests and NTP control queries and to synchronize itself to the servers specified in the access list. • The **serve** keyword enables the device to receive time requests and NTP control queries from the servers specified in the access list but not to synchronize itself to the specified servers. • The **serve-only** keyword enables the device to receive only time requests from servers specified in the access list. • The **query-only** keyword enables the device to receive only NTP control queries from the servers specified in the access list.

	Command or Action	Purpose
		• The **match-all** keyword enables the access group options to be scanned in the following order: peer, serve, serve-only, query-only.
Step 3	switch(config)# **show ntp access-groups**	(Optional) Displays the NTP access group configuration.
Step 4	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to configure the device to allow it to synchronize to a peer from access group "accesslist1":

```
switch# configure terminal
switch(config)# ntp access-group peer accesslist1
switch(config)# show ntp access-groups
Access List Type
----------------------------
accesslist1 Peer
switch(config)# copy running-config startup-config
[########################################] 100%
switch(config)#
```

Configuring the NTP Source IP Address

NTP sets the source IP address for all NTP packets based on the address of the interface through which the NTP packets are sent. You can configure NTP to use a specific source IP address.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	[**no**] **ntp source** *ip-address*	Configures the source IP address for all NTP packets. The *ip-address* can be in IPv4 or IPv6 format.

Example

This example shows how to configure an NTP source IP address of 192.0.2.2.

```
switch# configure terminal
switch(config)# ntp source 192.0.2.2
```

Configuring the NTP Source Interface

You can configure NTP to use a specific interface.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	[**no**] **ntp source-interface** *interface*	Configures the source interface for all NTP packets. The following list contains the valid values for *interface*. • ethernet • loopback • mgmt • port-channel • vlan

Example

This example shows how to configure the NTP source interface:

```
switch# configure terminal
switch(config)# ntp source-interface ethernet
```

Configuring NTP Logging

You can configure NTP logging in order to generate system logs with significant NTP events. NTP logging is disabled by default.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# [**no**] **ntp logging**	Enables or disables system logs to be generated with significant NTP events. NTP logging is disabled by default.
Step 3	(Optional) switch(config)# **show ntp logging-status**	Displays the NTP logging configuration status.
Step 4	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

The following example shows how to enable NTP logging in order to generate system logs with significant NTP events:

```
switch# configure terminal
switch(config)# ntp logging
switch(config)# copy running-config startup-config
[########################################] 100%
switch(config)#
```

Enabling CFS Distribution for NTP

You can enable CFS distribution for NTP in order to distribute the NTP configuration to other CFS-enabled devices.

Before you begin

Make sure that you have enabled CFS distribution for the device.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# [**no**] **ntp distribute**	Enables or disables the device to receive NTP configuration updates that are distributed through CFS.
Step 3	(Optional) switch(config)# **show ntp status**	Displays the NTP CFS distribution status.
Step 4	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to enable the device to receive NTP configuration updates through CFS:

```
switch# configure terminal
switch(config)# ntp distribute
switch(config)# copy running-config startup-config
```

Committing NTP Configuration Changes

When you commit the NTP configuration changes, the effective database is overwritten by the configuration changes in the pending database and all the devices in the network receive the same configuration.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **ntp commit**	Distributes the NTP configuration changes to all Cisco NX-OS devices in the network and releases the CFS lock. This command overwrites the effective database with the changes made to the pending database.

Discarding NTP Configuration Changes

After making the configuration changes, you can choose to discard the changes instead of committing them. If you discard the changes, Cisco NX-OS removes the pending database changes and releases the CFS lock.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **ntp abort**	Discards the NTP configuration changes in the pending database and releases the CFS lock. Use this command on the device where you started the NTP configuration.

Releasing the CFS Session Lock

If you have performed an NTP configuration and have forgotten to release the lock by either committing or discarding the changes, you or another administrator can release the lock from any device in the network. This action also discards pending database changes.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **clear ntp session**	Discards the NTP configuration changes in the pending database and releases the CFS lock.

Verifying the NTP Configuration

Command	Purpose
show ntp access-groups	Displays the NTP access group configuration.

Command	Purpose
show ntp authentication-keys	Displays the configured NTP authentication keys.
show ntp authentication-status	Displays the status of NTP authentication.
show ntp internal	Displays internal NTP information.
show ntp logging-status	Displays the NTP logging status.
show ntp peer-status	Displays the status for all NTP servers and peers.
show ntp peer	Displays all the NTP peers.
show ntp pending	Displays the temporary CFS database for NTP.
show ntp pending-diff	Displays the difference between the pending CFS database and the current NTP configuration.
show ntp rts-update	Displays the RTS update status.
show ntp session status	Displays the NTP CFS distribution session information.
show ntp source	Displays the configured NTP source IP address.
show ntp source-interface	Displays the configured NTP source interface.
show ntp statistics {**io** | **local** | **memory** | **peer** {**ipaddr** {*ipv4-addr*} | **name** *peer-name*}}	Displays the NTP statistics.
show ntp status	Displays the NTP CFS distribution status.
show ntp trusted-keys	Displays the configured NTP trusted keys.
show running-config ntp	Displays NTP information.

Configuration Examples for NTP

Configuration Examples for NTP

This example shows how to configure an NTP server and peer, enable NTP authentication, enable NTP logging, and then save the startup configuration so that it is saved across reboots and restarts:

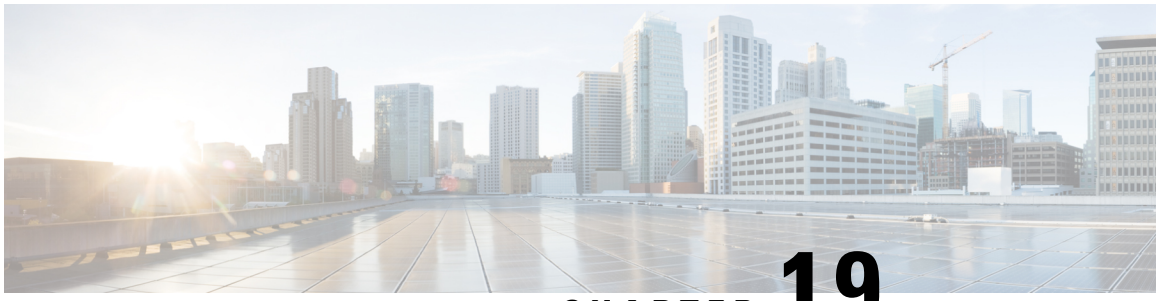
```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# ntp server 192.0.2.105 key 42
switch(config)# ntp peer 192.0.2.105
switch(config)# show ntp peers
-------------------------------------------------
Peer IP Address Serv/Peer
-------------------------------------------------
192.0.2.100 Peer (configured)
192.0.2.105 Server (configured)
switch(config)# ntp authentication-key 42 md5 aNiceKey
```

```
switch(config)# show ntp authentication-keys
----------------------------
Auth key MD5 String
----------------------------
42 aNicekey
switch(config)# ntp trusted-key 42
switch(config)# show ntp trusted-keys
Trusted Keys:
42
switch(config)# ntp authenticate
switch(config)# show ntp authentication-status
Authentication enabled.
switch(config)# ntp logging
switch(config)# show ntp logging
NTP logging enabled.
switch(config)# copy running-config startup-config
[########################################] 100%
switch(config)#
```

This example shows an NTP access group configuration with the following restrictions:

- Peer restrictions are applied to IP addresses that pass the criteria of the access list named “peer-acl.”
- Serve restrictions are applied to IP addresses that pass the criteria of the access list named “serve-acl.”
- Serve-only restrictions are applied to IP addresses that pass the criteria of the access list named “serve-only-acl.”
- Query-only restrictions are applied to IP addresses that pass the criteria of the access list named “query-only-acl.”

```
switch# configure terminal
switch(config)# ntp peer 10.1.1.1
switch(config)# ntp peer 10.2.2.2
switch(config)# ntp peer 10.3.3.3
switch(config)# ntp peer 10.4.4.4
switch(config)# ntp peer 10.5.5.5
switch(config)# ntp peer 10.6.6.6
switch(config)# ntp peer 10.7.7.7
switch(config)# ntp peer 10.8.8.8
switch(config)# ntp access-group peer peer-acl
switch(config)# ntp access-group serve serve-acl
switch(config)# ntp access-group serve-only serve-only-acl
switch(config)# ntp access-group query-only query-only-acl
switch(config)# ip access-list peer-acl
switch(config-acl)# 10 permit ip host 10.1.1.1 any
switch(config-acl)# 20 permit ip host 10.8.8.8 any
switch(config)# ip access-list serve-acl
switch(config-acl)# 10 permit ip host 10.4.4.4 any
switch(config-acl)# 20 permit ip host 10.5.5.5 any
switch(config)# ip access-list serve-only-acl
switch(config-acl)# 10 permit ip host 10.6.6.6 any
switch(config-acl)# 20 permit ip host 10.7.7.7 any
switch(config)# ip access-list query-only-acl
switch(config-acl)# 10 permit ip host 10.2.2.2 any
switch(config-acl)# 20 permit ip host 10.3.3.3 any
```



CHAPTER 19

Configuring EEM

This chapter contains the following sections:

Information About Embedded Event Manager

The Embedded Event Manager (EEM) monitors events that occur on your device and takes action to recover or troubleshoot these events, based on your configuration.

EEM consists of three major components:

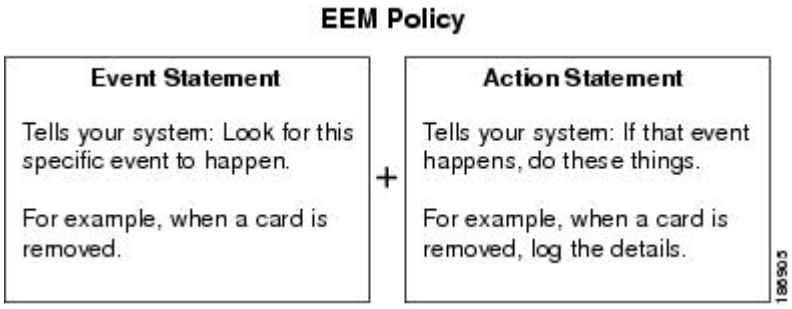
- Event statements—Events to monitor from another Cisco NX-OS component that might require some action, workaround, or notification.
- Action statements—An action that EEM can take, such as sending an e-mail, or disabling an interface, to recover from an event.
- Policies—An event paired with one or more actions to troubleshoot or recover from the event.

EEM 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.

The following figure shows the two basic statements in an EEM policy.

Figure 2: EEM Policy Statement



You can configure EEM policies by using the CLI or a VSH script.

EEM gives you a device-wide view of policy management. You configure EEM policies on the supervisor, and EEM pushes the policy to the correct module based on the event type. EEM takes any actions for a triggered event either locally on the module or on the supervisor (the default option).

EEM maintains event logs on the supervisor.

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 (__).

You can create user policies to suit your network. If you create a user policy, any actions in your policy occur after EEM triggers any system policy actions that are related to the same event as your policy. To configure a user policy, see Defining a User Policy Using the CLI, on page 229.

You can also override some system policies. The overrides that you configure take the place of the system policy. You can override the event or the actions.

Use the **show event manager system-policy** command to view the preconfigured system policies and determine which policies that you can override.

To configure an overriding policy, see Overriding a Policy, on page 235.

Note You should use the **show running-config eem** command to check the configuration of each policy. An override policy that consists of an event statement and no action statement triggers no action and no notification of failures.

Your override policy should always include an event statement. An override policy without an event statement overrides all possible events in the system policy.

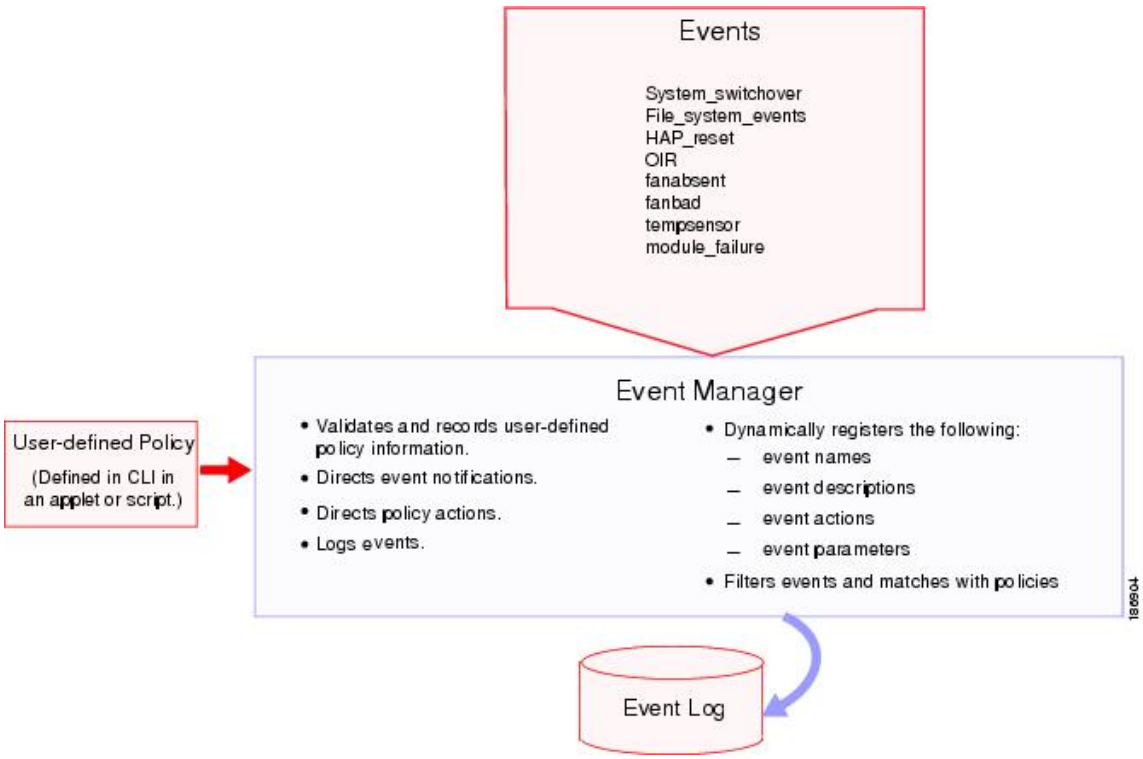
EEM Event Statement

An event is any device activity for which some action, such as a workaround or a notification, should be taken. In many cases, these events are related to faults in the device such as when an interface or a fan malfunctions.

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

The following figure shows events that are handled by EEM.

Figure 3: EEM Overview



Event statements specify the event that triggers a policy to run. You can configure multiple event triggers. For more information on configuring multiple events, see EEM Event Correlation, on page 226.

EEM schedules and runs policies on the basis of event statements. EEM examines the event and action commands and runs them as defined.

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

EEM Action Statements

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

EEM supports the following actions in action statements:

- Execute any CLI commands.
- Update a counter.
- Log an exception.
- Reload the device.
- Generate a syslog message.
- Generate an SNMP notification.
- Use the default action for the system policy.

Note If you want to allow the triggered event to process any 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 CLI command to execute.

Verify that your action statements within your user policy or overriding policy do not negate each other or adversely affect the associated system policy.

VSH Script Policies

You can also write policies in a VSH script, using a text editor. These policies have an event statement and action statement(s) just as other policies, and these policies can either augment or override system policies. After you write your VSH script policy, copy it to the device and activate it. To configure a policy in a VSH script, see Defining a Policy Using a VSH Script, on page 234.

EEM Event Correlation

You can trigger an EEM policy based on a combination of events. First, you use the **tag** keyword to create and differentiate multiple events in the EEM policy. Then, using a set of Boolean operators (and, or, andnot), with the count and time, you can define a combination of these events to trigger a custom action.

Note For information about configuring EEM event correlation, see Defining a User Policy Using the CLI, on page 229.

EEM Virtualization Support

You configure EEM in the virtual device context (VDC) that you are logged into. By default, Cisco NX-OS places you in the default VDC. You must be in this VDC to configure policies for module-based events.

Not all actions or events are visible in all VDCs. You must have network-admin or vdc-admin privileges to configure policies.

EEM Licensing Requirements

Product	License Requirement
Cisco NX-OS	EEM 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.

Prerequisites for EEM

EEM has the following prerequisite:

- You must have network-admin or vdc-admin user privileges to configure EEM.

Guidelines and Limitations for EEM

EEM has the following configuration guidelines and limitations:

- 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 CLI command in a match statement, you must add the event-default action statement to the EEM policy or EEM does not allow the CLI 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 * symbol can be used for argument replacement.
- 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, module, module-failure, oir, 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.

Default Settings for EEM

The following table lists the default setting for EEM parameters:

Table 33: Default EEM Parameters

Parameters	Default
System Policies	Active

Configuring EEM

Defining an Environment Variable

You can define a variable to serve as a parameter in an EEM policy.

Before you begin

Make sure that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **event manager environment** *variable-name variable-value*	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 alphanumeric string up to 39 characters.
Step 3	(Optional) switch(config)# **show event manager environment** {*variable-name* | **all**}	Displays information about the configured environment variables. Enclose the string in quotation marks.
Step 4	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to define an environment variable:

```
switch# configure terminal
switch(config)# event manager environment emailto "admin@anyplace.com"
switch(config)# show event manager environment all
switch(config)# copy running-config startup-config
```

Defining a User Policy Using the CLI

You can define a user policy using the CLI.

Before you begin

Make sure that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **event manager applet** *applet-name*	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	(Optional) switch(config-applet)# **description** *policy-description*	Configures a descriptive string for the policy. The string can be any alphanumeric string up to 80 characters. Enclose the string in quotation marks.
Step 4	switch(config-applet)# **event** *event-statement*	Configures the event statement for the policy. See Event Statement Configuration, on page 230. Repeat Step 4 for multiple event statements.
Step 5	(Optional) switch(config-applet)# **tag** *tag* {**and** | **andnot** | **or**} *tag* [**and** | **andnot** | or {*tag*}] {**happens** *occurs* **in** *seconds*}	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	switch(config-applet)# **action** *number*[.*number2*] *action-statement*	Configures an action statement for the policy. See Action Statement Configuration, on page 232. Repeat Step 6 for multiple action statements.
Step 7	switch(config-applet)# **show event manager policy-state** *name* [**module** *module-id*]	Displays information about the status of the configured policy.

	Command or Action	Purpose
Step 8	switch(config-applet)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to define a user policy by using the CLI:

```
switch# configure terminal
switch(config)# event manager applet monitorShutdown
switch(config-applet)# description "Monitors interface shutdown."
switch(config-applet)# event cli match "shutdown"
switch(config-applet)# tag one or two happens 1 in 10000
switch(config-applet)# action 1.0 cli local show interface e 3/1
switch(config-applet)# show event manager policy-state monitorShutdown
switch(config-applet)# copy running-config startup-config
```

Event Statement Configuration

Use one of the following commands in EEM configuration mode to configure an event statement:

Command	Purpose
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* argument range is from 1 to 65000. The *seconds* argument range is from 0 to 4294967295, where 0 indicates no time limit. Enclose the string in quotation marks.
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.

Command	Purpose
event fanabsent [**fan** *number*] **time** *seconds* **Example**: switch(config-applet)# **event fanabsent time 3000**	Triggers an event if a fan is removed from the device for more than the configured time, in seconds. The *number* range is module dependent. The *seconds* range is from 10 to 64000.
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.
event oir [**tag** *tag*] {**fan** | **module** | **powersupply**} {**anyoir** | **insert** | **remove**} [*number*] **Example:** switch(config-applet)# **event oir fan remove 4**	Triggers an event if the configured device element (fan, module, or power supply) is inserted or removed from the device. The **tag** *tag* keyword-argument pair identifies this specific event when multiple events are included in the policy. You can optionally configure a specific fan, module, or power supply number. The *number* range is as follows: • Fan number—Module dependent. • Module number—Device dependent. • Power supply number—The range is from 1 to 3.
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.
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. 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 1 to 2147483647.

Command	Purpose
event storm-control **Example:** switch(config-applet)# **event storm-control**	Triggers an event if the traffic on a port exceeds the configured storm control threshold.
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 range for the percentage is from 1 to 99.
event temperature [**module** *slot*] [**sensor** *number*] **threshold** {**any** | **major** | **minor**} **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.
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.

Action Statement Configuration

Use the following commands in EEM configuration mode to configure action statements:

Command	Purpose
action *number*[*.number2*] **cli** *command1* [*command2...*] [**local**] **Example:** switch(config-applet)# **action 1.0 cli "show interface e 3/1"**	Runs the configured CLI commands. You can optionally run the commands on the module where the event occurred. The action label is in the format number1.number2. *number* can be any number up to 16 digits. The range for *number2* is from 0 to 9. Enclose the string in quotation marks.

Command	Purpose
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. *number* can be any number up to 16 digits. The range for *number2* is from 0 to 9. The counter name 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.
action *number*[.*number2*] **event-default** **Example:** switch(config-applet)# **action 1.0 event-default**	Executes the default action for the associated event. The action label is in the format number1.number2. *number* can be any number up to 16 digits. The *range* for number2 is from 0 to 9.
action *number*[.*number2*] **policy-default** **Example:** switch(config-applet)# **action 1.0 policy-default**	Executes the default action for the policy that you are overriding. The action label is in the format number1.number2. *number* can be any number up to 16 digits. The range for *number2* is from 0 to 9.
action *number*[.*number2*] **reload** [**module** *slot* [-*slot*]] **Example:** switch(config-applet)# **action 1.0 reload module 3-5**	Forces one or more modules or the entire system to reload. *number* can be any number up to 16 digits. The range for *number2* is from 0 to 9.
action *number*[.*number2*] **snmp-trap** {[**intdata1** *data* [**intdata2** *data*] [*strdata string*]} **Example:** switch(config-applet)# **action 1.0 snmp-trap strdata "temperature problem"**	Sends an SNMP trap with the configured data. *number* can be any number up to 16 digits. The range for *number2* is from 0 to 9. The *data* arguments can by any number up to 80 digits. The *string* can be any alphanumeric string up to 80 characters. Enclose the string in quotation marks.
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. *number* can be any number up 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.

Note If you want to allow the triggered event to process any 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 CLI command to execute. You can use the **terminal event-manager bypass** command to allow all EEM policies with CLI matches to execute the CLI command.

Defining a Policy Using a VSH Script

You can define a policy using a VSH script.

Before you begin

Make sure that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

Ensure that you are logged in with administrator privileges.

Ensure that your script name is the same name as the script filename.

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
```

Registering and Activating a VSH Script Policy

You can register and activate a policy defined in a VSH script.

Before you begin

Make sure that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **event manager policy** *policy-script*	Registers and activates an EEM script policy. The *policy-script* can be any case-sensitive, alphanumeric string up to 29 characters.
Step 3	(Optional) switch(config)# **show event manager policy internal** *name*	Displays information about the configured policy.

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

Example

This example shows how to register and activate a VSH script policy:

```
switch# configure terminal
switch(config)# event manager policy moduleScript
switch(config)# show event manager policy internal moduleScript
switch(config)# copy running-config startup-config
```

Overriding a Policy

You can override a system policy.

Before you begin

Make sure that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **show event manager policy-state** *system-policy*	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	switch(config)# **event manager applet** *applet-name* **override** *system-policy*	Overrides a system policy and enters applet configuration mode. The *applet-name* can be any case-sensitive, alphanumeric string up to 29 characters. The *system-policy* must be one of the existing system policies.
Step 4	(Optional) switch(config-applet)# **description** *policy-description*	Configures a descriptive string for the policy. The string can be any alphanumeric string up to 80 characters. Enclose the string in quotation marks.
Step 5	switch(config-applet)# **event** *event-statement*	Configures the event statement for the policy. See the "Configuring Event Statements" section on page 14-10.
Step 6	switch(config-applet)# **action** *number action-statement*	Configures an action statement for the policy. See the "Configuring Action Statements" section on page 14-13.

	Command or Action	Purpose
		Repeat Step 6 for multiple action statements.
Step 7	(Optional) switch(config-applet)# **show event manager policy-state** *name*	Displays information about the configured policy.
Step 8	switch(config-applet)# **copy running-config startup-config**	Saves this configuration change.

Example

This example shows how to override a policy:

```
switch# configure terminal
switch(config)# 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
switch(config)# event manager applet ethport override _ethpm_link_flap
switch(config-applet)# description "Overrides link flap policy"
switch(config-applet)# event policy-default count 2 time 1000
switch(config-applet)# action 1.0 syslog priority warnings msg "Link is flapping."
switch(config-applet)# show event manager policy-state ethport
switch(config-applet)# copy running-config startup-config
```

Configuring the Syslog as an EEM Publisher

You can monitor syslog messages from the switch.

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

Before you begin

Make sure that the EEM is available for registration by syslog.

Configure and execute the syslog daemon.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **event manager applet** *applet-name*	Registers an applet with EEM and enters applet configuration mode.
Step 3	switch(config)# **event syslog** [**tag** *tag*] {**occurs** *number* | **period** *seconds* | **pattern** *msg-text* | **priority** *priority*}	Monitors syslog messages and invokes the policy based on the search string in the policy.

	Command or Action	Purpose
		• The **tag** *tag* keyword-argument pair identifies this specific event when multiple events are included in the policy. • The **occurs** *number* keyword-argument pair specifies the number of occurrences. The range is from 1 to 65000. • The **period** *seconds* keyword-argument pair specifies the interval during which the event occurs. The range is from 1 to 4294967295. • The **pattern** *msg-text* keyword-argument pair specifies the matching regular expression. The pattern can contain character text, an environment variable, or a combination of the two. If the string contains embedded blanks, it is enclosed in quotation marks. • The **priority** *priority* keyword-argument pair specifies the priority of the syslog messages. If this keyword is not selected, all syslog messages are set at the informational priority level.
Step 4	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to configure the syslog as an EEM publisher:

```
switch# configure terminal
switch(config)# event manager applet abc
switch(config-applet)# event syslog occurs 10
switch(config-applet)# copy running-config startup-config
```

Defining a User Policy Using the CLI to Trigger a Tcl Script

Before you begin

Copy the Tcl script which is triggered through an EEM policy to the bootflash of the switch.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **event manager applet** *applet-name*	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	(Optional) switch(config-applet)# **description** *policy-description*	Configures a descriptive string for the policy. The string can be any alphanumeric string up to 80 characters. Enclose the string in quotation marks.
Step 4	switch(config-applet)# **event** *event-statement*	Configures the event statement for the policy. See Event Statement Configuration, on page 230. Repeat Step 4 for multiple event statements.
Step 5	(Optional) switch(config-applet)# **tag** *tag* {**and** | **andnot** | **or**} *tag* [**and** | **andnot** | or {*tag*}] {**happens** *occurs* **in** *seconds*}	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	switch(config-applet)# **action** *number*[.*number2*] *action-statementtcl-filename*	Configures an action statement for the policy. See Action Statement Configuration, on page 232. Repeat Step 6 for multiple action statements.
Step 7	switch(config-applet)# **show event manager policy-state** *name* [**module** *module-id*]	Displays information about the status of the configured policy.
Step 8	switch(config-applet)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

Sample Tcl file (Vlan.tcl). Copy this file to the bootflash. Running the file creates 99 VLANs and names them.

```
set i 1
while {$i<100} {
cli configure terminal
cli vlan $i
cli name VLAN$i
cli no shutdown
cli exit
incr i
}
```

This example shows how to define a user policy by using the CLI and invoking a Tcl script by using the action statement once the event is triggered:

```
switch# configure terminal
switch(config)# event manager applet TCL
switch(config-applet)# description "Triggers TCL Script"
switch(config-applet)# event cli match "shutdown"
switch(config-applet)# action 1.0 cli local tclsh VLAN.tcl
switch(config-applet)# copy running-config startup-config
```

Defining a User Policy Using the CLI to Trigger a Python Script

Before you begin

Copy the Python script which is triggered through an EEM policy to the bootflash of the switch.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **event manager applet** *applet-name*	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	(Optional) switch(config-applet)# **description** *policy-description*	Configures a descriptive string for the policy. The string can be any alphanumeric string up to 80 characters. Enclose the string in quotation marks.
Step 4	switch(config-applet)# **event** *event-statement*	Configures the event statement for the policy. See Event Statement Configuration, on page 230. Repeat Step 4 for multiple event statements.
Step 5	(Optional) switch(config-applet)# **tag** *tag* {**and** | **andnot** | **or**} *tag* [**and** | **andnot** | or {*tag*}] {**happens** *occurs* **in** *seconds*}	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	switch(config-applet)# **action** *number*[.*number2*] *action-statementpython-filename*	Configures an action statement for the policy. See Action Statement Configuration, on page 232. Repeat Step 6 for multiple action statements.
Step 7	switch(config-applet)# **show event manager policy-state** *name* [**module** *module-id*]	Displays information about the status of the configured policy.

	Command or Action	Purpose
Step 8	switch(config-applet)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

Sample Python file (Python.py). Copy this file to the bootflash:scripts folder

```
import re
import cisco
cisco.cli ("show interface eth 1/1-32 transceiver detail >> bootflash:link_flap.txt")
```

This example shows how to define a user policy by using the CLI and invoking a Python script by using the action statement once the event is triggered:

```
switch# configure terminal
switch(config)# event manager applet PYTHON
switch(config-applet)# description "Triggers PYTHON Script"
switch(config-applet)# event cli match "shutdown"
switch(config-applet)# action 1.0 cli source Python.py
switch(config-applet)# copy running-config startup-config
```

Verifying the EEM Configuration

To display EEM configuration information, enter one of the following commands:

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

Command	Purpose
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 EEM

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

```
switch# configure terminal
switch(config)# event manager applet ethport override __ethpm_link_flap
switch(config-applet)# event policy-default count 2 time 1000
switch(config-applet)# action 1 cli conf t
switch(config-applet)# action 2 cli int et1/1
switch(config-applet)# action 3 cli no shut
```

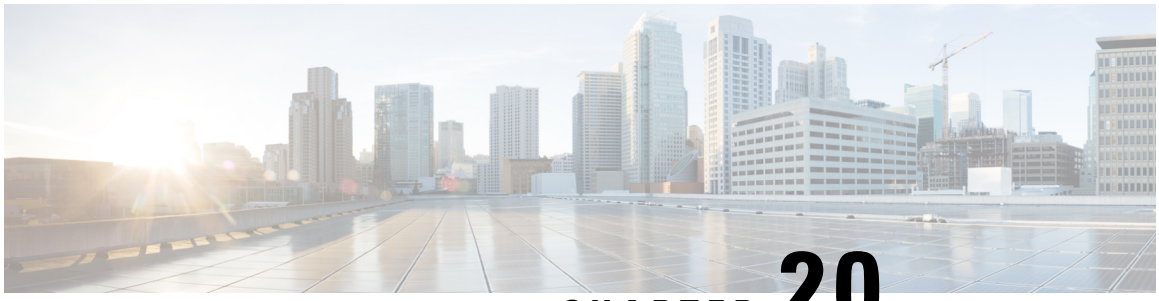
This example shows how to create an EEM policy that allows the CLI command to execute but trigger an SNMP notification when a user enters configuration mode on the device:

```
switch# configure terminal
switch(config)# event manager applet TEST
switch(config-applet)# event cli match "conf t"
switch(config-applet)# action 1.0 snmp-trap strdata "Configuration change"
switch(config-applet)# action 2.0 event-default
```

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

This 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:

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



CHAPTER 20

Configuring OpenFlow

This chapter contains the following sections:

Information About OpenFlow

OpenFlow is a specification from the Open Networking Foundation (ONF) that defines a flow-based forwarding infrastructure (L2-L4 Ethernet switch model) and a standardized application programmatic interface (protocol definition) to learn capabilities, add and remove flow control entries and request statistics. OpenFlow allows a controller to direct the forwarding functions of a switch through a secure channel.

Cisco ONE Platform Kit provides the ability to host Cisco internal or external third party applications on or adjacent to Cisco's networking infrastructure, and enables programmatic access to networking services in a controlled and consistent manner. When hosting applications on Cisco routers or switches, the applications will run within a virtual-machine or container.

OpenFlow Limitations

The Cisco Nexus 5500 and Cisco Nexus 6000 switches do not support the OpenFlow action to rewrite the layer-2 destination MAC address. Therefore, the XNC controller use cases such as Topology Independent

Forwarding and Latency Optimized Forwarding may not be work correctly on the Cisco Nexus 5500 and Cisco Nexus 6000 switches.

Supported Interface Types

The following is a list of supported interface types:

- Regular Layer 2 physical ports (switchport)
- Layer 2 port channels

Unsupported Interface Types

The following is a list of unsupported interface types:

- Layer 3 ports (no switchport)
- Fabric extender ports
- Virtual Port-Channel (VPC) ports
- Layer 3 Port-Channel

Supported Interface Modes

The following is a list of supported interface modes:

- Access port
- Trunk port

Supported Match Fields

The following are lists of supported match fields:

- Layer 2 header
  - Ethertype
  - VLAN ID
  - VLAN priority (PCP)
  - Source MAC address
  - Destination MAC address
- Layer 3 header
  - Source IP address

    - Destination IP address
    - Layer 4 protocol
    - Differentiated services Code Point (DSCP)
- Layer 4 header
    - Source port
    - Destination port
- Ingress Interface

Supported Actions

The following is a list of supported actions:

- Redirect the packet to one output port
- Divert the datapath packet to the OpenFlow controller
- Drop the packet

- Redirect the packet to one output port
- Redirect the packet to multiple output ports
- Set the VLAN tag (vlan rewrite) on egress
- Strip the VLAN tag on egress
- Divert the datapath packet to the OpenFlow controller
- Drop the packet

- Redirect the packet to one output port
- Redirect the packet to multiple output ports
- Set the VLAN tag (vlan rewrite) on egress
- Strip the VLAN tag on egress
- Divert the datapath packet to the OpenFlow controller
- Drop the packet

Scale Flow Numbers

- The Cisco Nexus device supports up to 1200 ACL-table flows and 32K MAC-table flows.
- The Cisco Nexus device supports a maximum of 65535 flows in total. The device supports a combination of up to 3500 ACL-table flows and 62K MAC-table flows.

- The Cisco Nexus device supports up to 14K MAC flows in the ACL table.
- The Cisco Nexus device supports up to 64 flows when the action is punt-to-controller.

Pipeline Support

OpenFlow policies can be applied to the ACL-table and the MAC-table. OpenFlow relates tables by means of the 'pipeline' concept. The Cisco Nexus device supports two pipelines, 201 and 202. You can toggle the pipeline between 201and 202 by entering the **pipeline** *id* command in the openflow-agent logical switch configuration.

- Pipeline 201
  - All the flows are added to the ACL-table. For example, ACL TCAM.
  - ACL-table flows with the action as redirect or drop gets installed in the IFACL region of the ACL-TCAM.
  - ACL-table flows with the action punt-to-controller are installed in the SUP region of the ACL-TCAM.
  - Source and destination MAC address match are supported as actions.
- Pipeline 202
  - Flows can be added to both the ACL-table(ACL TCAM) and the MAC-table(STM table).
  - Flows with only L2-dest-mac and VLAN as the match criteria are installed in the MAC-table. The remaining flows are installed in ACL-table
  - Supported actions for the MAC-table are redirect-to-port, normal and drop.
  - The MAC-table supports a higher scale number than the ACL-table.
  - Supported action for default rule in the MAC-table is punt-to-controller.

Prerequisites for OpenFlow

The OpenFlow agent requires the Cisco Nexus device to be configured with OpenFlow specific commands in order to support topology discovery and the installation of flows. The Cisco Nexus device works in a hybrid mode so that the default commands from the startup-config file are executed upon boot up. This might create an undesirable effect and therefore must be changed.

Note If you change or negate these required commands, it can lead to unpredictable system behavior.

VLAN Creation

The following command is used to create the necessary VLANs in an OpenFlow-controller switch. This command creates the OpenFlow specific VLANs in the VLAN database.

```
vlan x[-y]
```

Even with the hybrid-Ships-In-Night mode of operation, we recommend that you segregate the VLANs among the OpenFlow-controlled ports and the regular ports. You should take caution in ensuring that the VLANs are not shared among the OpenFlow and non-OpenFlow ports in order to prevent traffic leaks.

Interface Level Configurations

To make the interfaces connected to other switches receive spanned traffic, the interface is connected to the analyzer and configured to support OpenFlow. The **interface ethernet** command changes the parser to the interface submode. Before entering the **mode openflow** command which enables OpenFlow support on the interface, the following commands are required:

- **switchport mode trunk**
- **switchport trunk allowed vlan x-y**

In order for the strip-vlan functionality to work on the Cisco Nexus device, the trunk port must be configured with the native VLAN.

Cisco One controllers can perform topology discovery of OpenFlow enabled ports. To allow topology discovery on trunk ports, the native VLANs must be configured on trunk ports

switchport trunk native vlan z

When an interface is added to the OpenFlow logical switch, the following commands are applied to the interface implicitly:

- **mode openflow**
- **spanning-tree bpdufilter enable**
- **no lldp transmit**

Template Based TCAM Carving for OpenFlow

The Cisco Nexus device supports template-based TCAM carving. To configure OpenFlow on the device, you must make a number of changes to the TCAM carving regions using the template based TCAM carving commands.

To configure OpenFlow on the switch you must increase the default size of the Sup region to 256 using the template based TCAM carving commands, as shown in this example:

```
switch(config)# hardware profile tcam resource template openflow
switch(config-tcam-templ)# sup 256
switch(config)# hardware profile tcam resource service-template openflow
```

To support higher scale numbers for OpenFlow policies, the IFACL-region of the TCAM must be recarved accordingly. To apply TCAM carving for a maximum flow scale, enter the following commands:

```
switch(config)# hardware profile tcam resource template openflow
switch(config-tcam-templ)# vacl 64
switch(config-tcam-templ)# ifacl 3520
switch(config-tcam-templ)# qos 128
switch(config-tcam-templ)# rbacl 64
switch(config-tcam-templ)# span 64
switch(config)# hardware profile tcam resource service-template openflow
```

Enter the following command to verify the TCAM carving: **show hardware profile tcam resource template** *tmplt-name*

Note Configuring TCAM carving requires that the Cisco Nexus device be reloaded.

Setting Up an OpenFlow Virtual Service

The virtual service manager allows you to enable the OpenFlow agent application to run as a virtual service on a container. To setup a virtual service for OpenFlow you must perform the following tasks:

- Download the application OVA package to your system.
- Install the OVA package for a named virtual service. For example:

```
switch#virtual-service install name openflow-agent package file-url
```

- Configure and activate the virtual service. For example:

```
switch(config)#virtual-service openflow-agent
switch(config-virt-serv)#activate
```

To upgrade a software package installed on a virtual service you use the **virtual-service upgrade name** *application-name* **package** *file-url* command.

Note An active virtual service can not be updated.

To remove a software package installed on a virtual service you use the **virtual-service uninstall name** *application-name* command.

Note An active virtual service can not be removed.

Enabling OpenFlow

OpenFlow capability is enabled by entering the **hardware profile openflow** command to allocate the hardware resources required for the OpenFlow agent. Following a switch reload, the **hardware profile** command is used to configure ACL Feature Manager (AFM) and Forwarding Manager (FWM) modules for OpenFlow functionality.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **hardware profile openflow**	Allocates the hardware resources required for the OpenFlow agent.

	Command or Action	Purpose
Step 3	switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts b copying the running configuration to the startup configuration.
Step 4	switch#**reload**	Reloads the operating system on the switch.

Configuring the OpenFlow Switch

You must enable OpenFlow on the switch, for the configuration to take effect.

Procedure

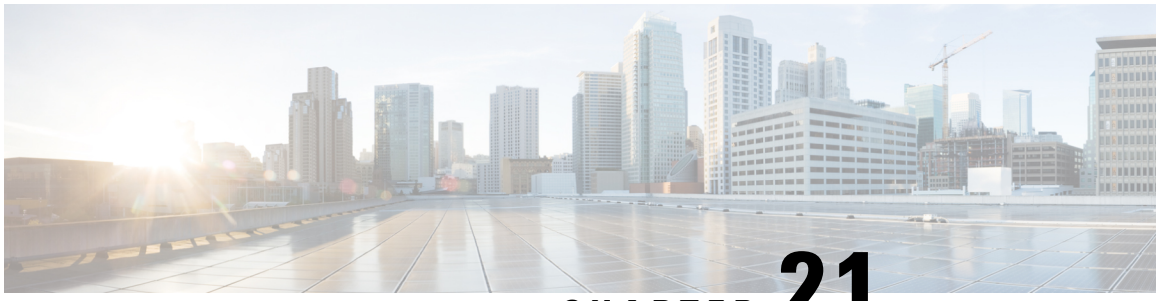
	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	**openflow** **Example:** `switch(config)# openflow`	Enters OpenFlow configuration mode.
Step 3	**switch** *switch-number* **Example:** `switch(config-ofa)# switch 1`	Specifies the OpenFlow logical switch and enters OpenFlow switch configuration mode.
Step 4	**pipeline {201|202}** **Example:** `switch(config-ofa-switch)# pipeline 201`	Specifies the pipeline mode. OpenFlow policies can be applied to the ACL-table and the MAC-table. Cisco Nexus devices support two pipelines, 201 and 202. This command allows you to switch between the supported pipeline modes.
Step 5	**controller ipv4** *ipv4-address* **port** *port-number***vrf** *vrf-name* **security** {**none** | **tls**} **Example:** `switch(config-ofa-switch)# controller ipv4 192.0.2.10 port 6653 vrf mnagement security none`	Establishes the connection with the controller over the specified VRF. You can disable or enable the TLS.
Step 6	**of-port interface** *interface-type slot / port* **Example:** `switch(config-ofa-switch)# interface ethernet 2/5`	Adds the interface to the OpenFlow logical switch.
Step 7	**default-miss cascade** {**drop** | **controller** | **normal**} **Example:**	Enables hybrid-normal mode on the switch. To change the OpenFlow agent from hybrid-normal to punt-to-controller mode use the **default-miss cascade controller**

	Command or Action	Purpose
	`switch(config-ofa-switch)# default-miss cascade normal`	command. To change th OpenFlow agent from hybrid-normal to default-drop mode, use the **default-miss cascade drop** command.
Step 8	**max-backoff** *back-off-time* **Example:** `switch(config-ofa-switch)# max-backoff 7`	Sets the OpenFlow controller maximum backoff timer. The default value is 8 seconds.
Step 9	**probe-internal** *interval-time* **Example:** `switch(config-ofa-switch)# probe-interval 6`	Sets the OpenFlow controller probe interval timer. The default value is 5 seconds.
Step 10	**exit** **Example:** `switch(config-ofa-switch)# exit`	Exits OpenFlow switch configuration mode.
Step 11	**exit** **Example:** `switch(config-ofa)# exit`	Exits OpenFlow configuration mode.

Verifying OpenFlow

Use one of the following commands to verify the configuration:

Command	Purpose
show running-config | section openflow	Displays the OpenFlow running configuration information.
show running-config interface ethernet *slot/port*	Displays the running configuration for a specific ethernet interface.
show openflow *openflow-agent* **switch** *number* **controllers**	Displays information about the OpenFlow agent connectivity to controller
show openflow *openflow-agent* **switch** *number* **flows**	Displays information about the OpenFlow agent flows.
show openflow *openflow-agent* **switch** *number* **ports**	Displays information about the OpenFlow agent port status.



CHAPTER 21

Configuring Secure Erase

Configuring Secure Erase

Beginning with Cisco NX-OS Release 7.3(11)N1(1), the Secure Erase feature is introduced to erase all customer information for switches. Secure Erase is an operation to remove all the identifiable customer information on Cisco NX-OS devices in conditions of product removal due to Return Merchandise Authorization (RMA), or upgrade or replacement, or system end-of-life.

Switches consume storage to conserve system software images, switch configuration, software logs, and operational history. These areas can have customer-specific information such as details regarding network architecture, and design as well as a potential target for data thefts.

Note To remove all to erase the customer data on FEX, ensure that the FEX factory reset action is performed before performing factory-reset on switch. For more information, refer to Configure Secure Erase section.

The Secure Erase process is used in the following two scenarios:

- Return Material Authorization (RMA) for a device - If you must return a device to Cisco for RMA, remove all the customer-specific data before obtaining an RMA certificate for the device.
- Recovering the compromised device - If the key material or credentials that are stored on a device is compromised, reset the device to factory configuration, and then reconfigure the device.

The device reloads to perform a factory reset which results in the switch entering the power-down mode. After a factory reset, the device clears all its environment variables including the MAC_ADDRESS and the SERIAL_NUMBER which are required to locate and load the software.

Prerequisites for Performing Secure Erase

- Ensure that all the software images, configurations, and personal data are backed up before performing the secure erase operation.
- Ensure that the device is not in stacking mode as factory reset is supported only in the standalone mode.
- Ensure that there is an uninterrupted power supply when the process is in progress.

- Ensure that you take a backup before you begin the secure erase process.
- Ensure that neither In-Service Software Upgrade (ISSU) nor In-Service Software Downgrade (ISSD) is in progress before starting the secure erase process.

Guidelines and Limitations for Secure Erase

- Software patches, if installed on the device, will not be restored after the Secure Erase operation.
- If the **factory-reset** command is issued through a session, the session is not restored after the completion of the factory reset process.
- The standby supervisor will be powered down after erasing it.
- If you configure secure erase of fex, the factory reset is initiated and fex configuration will be removed.
- After a successful factory reset, the switch will be powered down.
- The secure erase operation can take from 15 minutes to 2 hours.

Configuring Secure Erase

To delete all necessary data before shipping to RMA, configure secure erase using the below command:

Note If fex is attached to the switch, to erase the customer data on the connected fex perform below operation before performing a factory reset on the switch:

- To erase customer data on a single fex - **factory reset fex <fex-id>**
- To erase customer data on all fex - **factory reset fex all**

Procedure

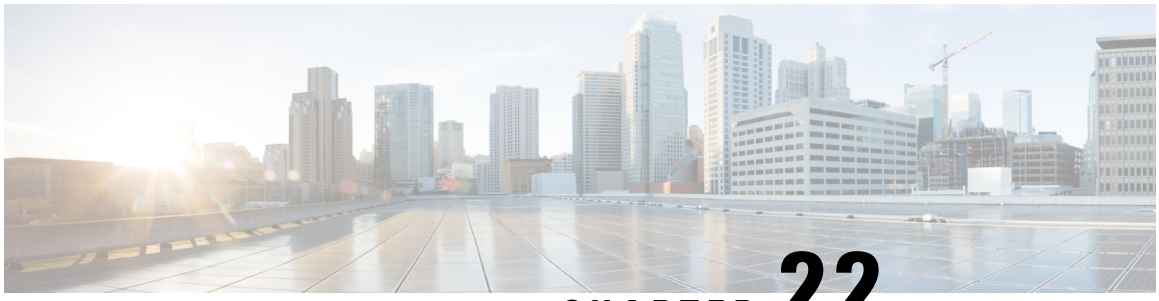
	Command or Action	Purpose
Step 1	**factory-reset** [**fex** [*all*]|[*fex id*]] **Example:** `switch# factory-reset`	Use the command with all options enabled. No system configuration is required to use the factory reset command. To initiate secure erase on switch only, use **factory-reset fex**. To initiate secure erase on fex only, use **factory-reset [fex {all|fex-id}]**.

Example

The following is an example output for secure erase on fex 102:

```
switch# factory-reset fex {all | fex-id}
switch# factory-reset fex 102
!!!! WARNING: This command will perform factory-reset of FEX module 102 !!!!
The factory reset operation will erase ALL persistent storage on the specified FEX module.
This includes configuration, all log data, and the full contents of flash and SSDs.
Special steps are taken in an effort to render data non-recoverable. Please, proceed with
caution and understanding that this operation cannot be undone and will leave the system
in
a fresh-from-factory state.
!!!! WARNING !!!!

Continue? (y/n) [n] y

Initiating factory-reset for the FEX: 102 --- SUCCESS!!
------------------------------------------------------------------------------
FEX: 102 is reloading for the reset operation to proceed.
Factory reset may take time...
Please, wait and do not power off the FEX...

Trying to remove the FEX:102 config !!!
2022 Feb 10 10:57:26 UUT4 %$ VDC-1 %$ %NOHMS-2-NOHMS_ENV_FEX_OFFLINE: FEX-102 Off-line
(Serial Number SSI182005PM)
2022 Feb 10 10:57:26 UUT4 %$ VDC-1 %$ %PFMA-2-FEX_STATUS: Fex 102 is offline
Successfully removed FEX:102 config. !!!
```

Configuring Secure Erase



CHAPTER 22

Soft Reload

This chapter contains the following sections:

Information About Soft Reload

The Soft Reload feature provides a best effort mechanism for the switch to be gracefully brought up with minimal impact to production traffic when a process crash occurs. You can also use the **soft-reload** command to trigger a manual soft reload of the switch. After a successful soft reload, we mandatorily recommend performing a normal switch reload as there may be some mismatch between the hardware and software configurations after the soft reload. The normal switch reload can also be performed during the next maintenance window.

We also recommend not making any changes to the configuration until a normal switch reload is done after a soft reload. During a normal switch reload, the switch is reloaded with the **copy running-config startup-config** command ensuring that all configurations are restored without any mismatch between hardware and software configurations. By default, Soft Reload is disabled.

Soft Reload Debugging

Syslogs are generated during various stages of a soft reload indicating the current health of a switch. The following syslogs can be used for debugging Soft Reload:

Syslog	Severity	Description
SOFTRELOAD_ATTEMPT_TRIGGERED	ALERT	Soft reload is triggered.
SOFTRELOAD_ATTEMPTED_SUCCESSFUL	ALERT	Soft reload triggered successfully.

Syslog	Severity	Description
SOFTRELOAD_ATTEMPTED_FAILURE	ALERT	Attempt for soft-reload failed.
INSTALLER_SOFTRELOAD_BASED_KEXEC	ALERT	Kexec is triggered.
INSTALLER_SOFTRELOAD_BASED_KEXEC_FAILED	ALERT	Kexec trigger failed.
INSTALLER_SOFTRELOAD_INITIATED	ALERT	Soft reload process is initiated.
INSTALLER_SOFTRELOAD_ABORTED	ALERT	Soft reload aborted due to internal failure.
INSTALLER_SOFTRELOAD_BOOTUP_ATTEMPT	ALERT	Soft reload attempting bootup.
INSTALLER_SOFTRELOAD_PROCESSING_BOOTUP	ALERT	Bootup attempt successful post softreload.
INSTALLER_SOFTRELOAD_BOOTUP_INPROGRESS	ALERT	Soft reload still in progress, waiting for login to be enabled.

Licensing Requirements for Soft Reload

The following table shows the licensing requirements for this feature:

Product	License Requirement
Cisco NX-OS	Soft Reload 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.

Guidelines and Limitations for Soft Reload

- A normal switch reload is attempted if a soft reload due to a process crash fails.
- A soft reload is not triggered when the following scenarios occur:
  - If Layer 3 licenses (LAN_BASE_SERVICES_PKG and LAN_ENTERPRISE_SERVICES_PKG) are installed.
  - Kernel panic/crash
  - Sysmgr crash
  - Crashing of the following processes: mmode, provision, xmlma, res, evms, evmc, securityd, aaa, snmpd, callhome, cts, m2rib, stp, ntp, ntpd, bigsurusd, carmelusd, pfma, sensor, pacifica, bootvar, ipqosmgr, vms, sh, libvirtd, init, sysmgr, pfma, vshd, licmgr and sysinfo.
- We recommend performing a manual soft reload during a debugging window. For example, you can initiate a soft reload to debug a periodic crash with minimal traffic disruption.

- If a soft reload that has been triggered by using the **soft-reload** command fails, the switch will not be reloaded. Soft reload can then be attempted again by using the **soft-reload** command after the failures shown have been corrected.
- Any connected FEXs are not reloaded during a soft reload.
- Soft reload is not triggered if an ISSU is in progress.
- ISSU is not attempted if soft reload is in progress.
- After a soft reload, the switch will come up with the last saved configuration along with any configuration changes made since the last saved switch configuration. However, the **show running-config** command will display only the last saved switch configuration.
- Production traffic may be impacted if there are any STP configurations that may lead to network convergences.
- Soft reload does not trigger all the syslogs available for ISSU.
- Soft reload is not triggered if any FEX processes crash.
- Hardware configurations are not modified during a soft reload.
- Follow the usual debug process to debug any process crashes that lead to a soft reload.
- If the switch crashes within 20 minutes after a soft reload, another soft reload is not triggered.
- If the switch crashes more than 20 minutes after a soft reload, another soft reload is triggered.
- We recommend doing a manual switch reload as soon as possible after a soft reload.
- After a soft reload, we recommend not making any configuration changes until a manual switch reload is done.

Default Setting for Soft Reload

Parameter	Default
Soft Reload	Disabled

Configuring Soft Reload

Enabling the Switch to Perform a Soft Reload After a Process Crash

Procedure

Step 1 Enter global configuration mode:

switch# **configure terminal**

Step 2 Enable the switch to perform a soft reload after a process crash:

switch(config)# **system soft-reload enable**

Step 3 (Optional) Display the status of the soft reload:

switch# **show system soft-reload status**

Running Configuration

This example shows a running configuration, followed by a verification command that displays the status of the soft reload.

```
configure terminal
  system soft-reload enable
  .
  .
  .
switch# show system soft-reload status
Soft-reload is enabled
```

Performing a Manual Soft Reload

Procedure

Step 1 (Optional) Display the status of the soft reload:

switch# **show system soft-reload status**

Step 2 Perform a manual soft reload of the switch:

switch(config)# **soft-reload**

Running Configuration

This example shows a verification command that displays the status of the soft reload, followed by the command to initiate a manual soft reload.

```
show system soft-reload status
Soft-reload is disabled

soft-reload
.
.
.
```

Configuration Examples for Soft Reload

This section provides configuration examples for Soft Reload.

Enabling the Switch to Perform a Soft Reload After a Process Crash

```
configure terminal
  system soft-reload enable
```

Performing a Manual Soft Reload

```
soft-reload
```

Verifying the Soft Reload Status

Command	Purpose
show system soft-reload status	Displays whether Soft Reload is enabled or disabled.

Additional References for Soft Reload

This section describes additional information related to implementing Soft Reload.

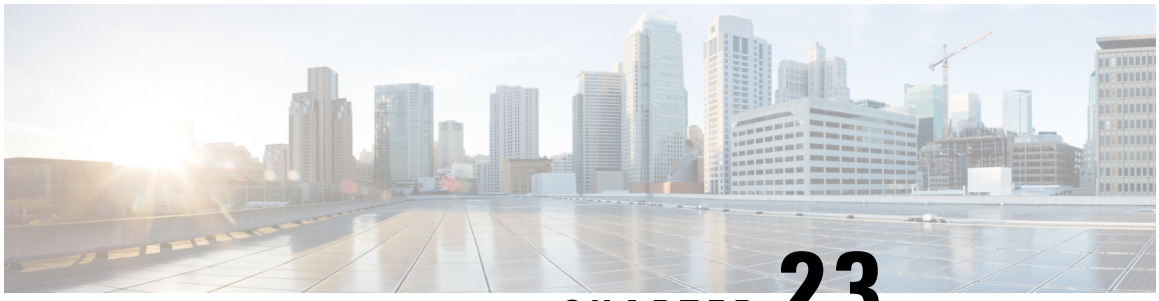
Related Documents

Related Topic	Document Title
Command reference	*Cisco Nexus 5500 Series NX-OS System Management Command Reference* *Cisco Nexus 6000 Series NX-OS System Management Command Reference*

Feature History for Soft Reload

This table lists the release history for this feature.

Feature Name	Release	Information
Soft Reload	7.3(2)N1(1)	The Soft Reload feature provides a best effort mechanism for the switch to be gracefully brought up with minimal impact to production traffic when a process crash occurs. You can also use the **soft-reload** command to trigger a manual soft reload of the switch.



CHAPTER 23

Configuring GIR (Cisco NX-OS Release 7.3(0)N1(1))

This chapter contains the following sections:

Information About GIR

You can use Graceful Insertion and Removal (GIR) to put a switch in maintenance mode in order to perform debugging or an upgrade. When switch maintenance is complete, you can return the switch to normal mode.

When you place the switch in maintenance mode, all protocols are isolated from the network. When normal mode is restored, all the protocols are brought back up.

In Cisco NX-OS Release 7.1(0)N1(1), the default mode for GIR is "**shutdown**". When you place the switch in maintenance mode, all protocols are gracefully brought down and all physical ports are shut down. When normal mode is restored, all the protocols and ports are brought back up. The following protocols are supported:

- Border Gateway Protocol (BGP)
- BGPv6
- Enhanced Interior Gateway Routing Protocol (EIGRP)
- EIGRPv6
- Intermediate System-to-Intermediate System (ISIS)

- ISISv6
- Open Shortest Path First (OSPF)
- OSPFv3
- RIP

Also supported are:

- Virtual port channel (vPC) and vPC+
- Interfaces
- FabricPath

Starting with Cisco NX-OS Release 7.3(0)N1(1), the default mode for GIR is "**isolate**". Use the **system mode maintenance** command to put all the enabled protocols in maintenance mode. The switch will use the **isolate** command to isolate the protocols from the network. The switch will then be isolated from the network but is not shut down. Routing protocols will be running on the switch to maintain neighborship with peer switches when it is isolated from the network. The **isolate** command is applied on the protocol instance and is applicable for the following protocols:

- Border Gateway Protocol (BGP)
- BGPv6
- Enhanced Interior Gateway Routing Protocol (EIGRP)
- EIGRPv6
- Intermediate System-to-Intermediate System (ISIS)
- ISISv6
- Open Shortest Path First (OSPF)
- OSPFv3
- FabricPath (Only applicable for Spine switches)

Note

- You can use the **system mode maintenance shutdown** command to use the "**shutdown**" mode for GIR as in the Cisco NX-OS Release 7.1(0)N1(1).
- When you cold boot a switch that has custom profile configured and is running a Cisco NX-OS Release 7.3(1)N1(1) image to any other Cisco NX-OS Release that does not support maintenance mode, the same configuration file cannot be used after write-erase reload.
- In normal mode, the processing of protocols will happen in an order that is the reverse of the order in which the protocols are processed in maintenance mode. Similarly, in maintenance mode, the processing of protocols will happen in an order that is the reverse of the order in which the protocols are processed in normal mode.

Maintenance Profile

Maintenance profile contains a set of commands that will be applied sequentially during graceful removal or graceful insertion.

By default, the system isolates all enabled protocols during graceful removal and restores them during graceful insertion. The protocols are isolated and restored in a predefined order.

The switch supports the following profiles:

- Maintenance-mode profile—Contains all the commands that will be executed during graceful removal, when the switch enters maintenance mode.
- Normal-mode profile—Contains all the commands that will be executed during graceful insertion, when the switch returns to normal mode.

System-generated Profile

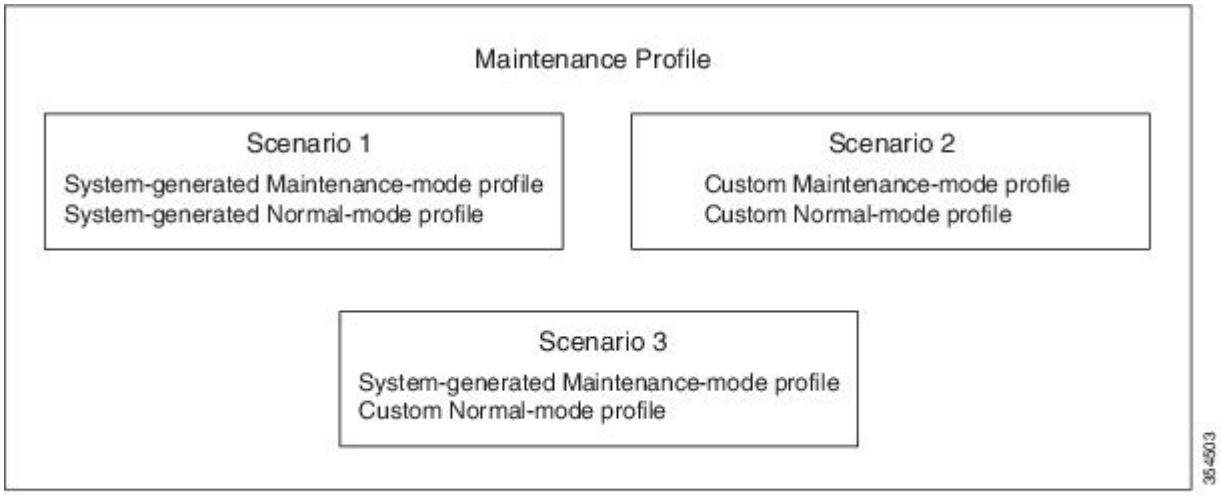
You can allow the system to generate a maintenance-mode or normal-mode profile with specific configuration commands. The system generates a maintenance-mode profile when you use the **system mode maintenance** command or a normal-mode profile when you use the **no system mode maintenance** command.

Custom Profile

You can create a custom maintenance-mode or normal-mode profile with configuration commands that can be applied during graceful removal or graceful insertion to isolate, shut down, or restore the protocols individually (or perform additional configurations). You can use a custom profile when the system-generated profile does not provide the required configuration or if you need to enhance the existing system-generated or custom profile to include additional functionality specific to your deployment. Use the **configure maintenance profile maintenance-mode** command to configure a custom maintenance-mode profile with the required commands or the **configure maintenance profile normal-mode** command to configure a custom normal-mode profile with the required commands.

The system-generated profile will overwrite the custom profile and vice-versa. The system can have either a system-generated maintenance-mode profile or a custom maintenance-mode profile at a time. Similarly, the system can have either a system-generated normal-mode profile or a custom normal-mode profile at a time. The scenarios are as given in the figure below:

Figure 4: Maintenance Profile Scenarios



Note We recommend using Scenario 1 or 2.

Unplanned Maintenance

You can put the switch in unplanned maintenance mode when the switch reloads due to a critical failure. For switches with a single supervisor, configure a reset reason CLI using the **system mode maintenance on-reload reset-reason** command to enable the switch to go into maintenance mode after a switch reloads due to a critical failure. For switches with dual supervisors, SUP switchover occurs when there is a critical failure of the switch and the switch will not go into maintenance mode. The maintenance-mode profile existing in the startup configuration is applied when the switch goes in to unplanned maintenance mode. If no maintenance mode profile exists in the startup configuration, a system-generated maintenance-mode profile is created and applied when the switch goes in to unplanned maintenance mode.

Maintenance Mode Timer

Use the **system mode maintenance timeout** command before entering maintenance mode to keep the switch in maintenance mode for a specified number of minutes. You can also use this command while the switch is in maintenance mode to change the number of minutes for which the switch will be in maintenance mode. The timer will then restart from that instant with the new timer value. Once the configured time elapses, the switch returns to normal mode automatically without using the **no system mode maintenance mode** command. Use the **no system mode maintenance timeout** command to disable the timer.

Snapshot

Use the **snapshot** command to capture the running states of selected features and to store the running states on the persistent storage media.

You can use snapshots to compare the state of a switch before it went into maintenance mode and after it came back to normal mode. The snapshot process consists of three parts:

- Creating a snapshot of the states of a few preselected features on the switch and storing them on the persistent storage media.
- Listing the snapshots taken at various time intervals and managing them.
- Comparing snapshots and showing the summary and details of each feature.

There are two types of snapshots:

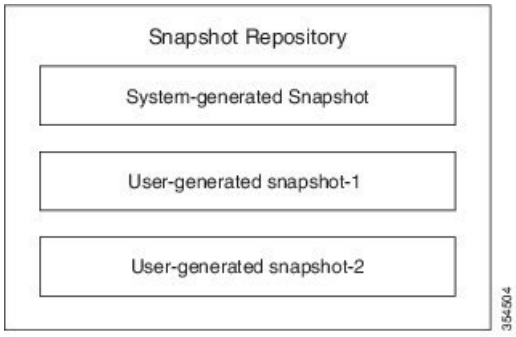
- System-generated snapshot—This is generated by the system when you use the **[no] system mode maintenance** command. The system creates the before_maintenance snapshot just before the system goes into maintenance mode. The system creates the after_maintenance snapshot just before the system goes into normal mode. The system overwrites any old snapshots when you use the **[no] system mode maintenance** command. Use the **snapshot delete** {**all** | *snapshot-name*} command to delete the system-generated snapshots.

  In certain scenarios, the system-generated after_maintenance snapshot may be taken when hardware programming is ongoing. In such cases, we recommend taking a user-generated snapshot after the system has completed hardware programming and is in a stable state. You can then compare the new after_maintenance snapshot with the before_maintenance snapshot.

- User-generated snapshot—Use the **snapshot create** *name description* command to create a user-generated snapshot. Use the**snapshot delete** {**all** | *snapshot-name*} command to delete user-generated snapshots.

The system-generated and user-generated snapshots are stored in the snapshot repository.

Figure 5: Snapshot Repository



The following table lists the snapshot sections with the corresponding show commands:

Name of the Section	Corresponding 'show' command
bgp-sessions	show bgp sessions vrf all
eigrp	show ip eigrp topology summary
eigrpv6	show ipv6 eigrp topology summary
interface	show interface
ospf	show ip ospf vrf all
ospfv3	show ipv6 ospfv3 vrf all
isis	show isis database detail vrf all
rip	show ip rip vrf all
route-summary	show ip route summary vrf all
routev6-summary	show ipv6 route summary vrf all
vpc	show vpc

Suppress FIB Pending

The Suppress Forwarding Information Base (FIB) Pending feature uses the Border Gateway Protocol-Routing Information Base (BGP-RIB) and the Enhanced Interior Gateway Routing Protocol-Routing Information Base (EIGRP-RIB) feedback mechanism to avoid premature route advertisements and subsequent packet loss in a network. This mechanism is enabled by default and ensures that routes are installed locally before they are advertised to a neighbor.

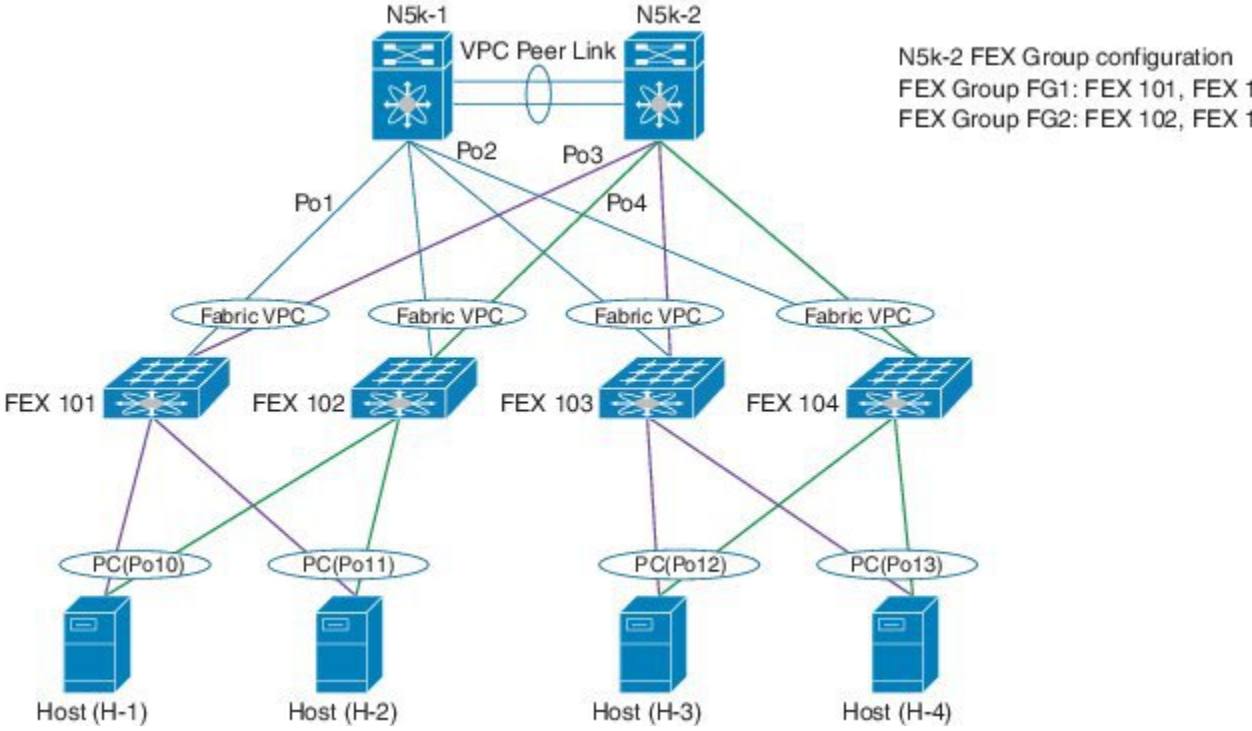
BGP and EIGRP wait for feedback from RIB indicating that the routes that EIGRP or BGP installed in the RIB are installed in the FIB before EIGRP or BGP sends out updates to the neighbors. EIGRP or BGP will send out updates of only those routes that have versions up to the version that FIB has installed. This selective update ensures that EIGRP or BGP does not send out premature updates resulting in attracting traffic even before the data plane is programmed after a switch reload, line card reload, or when the switch moves to normal mode from maintenance mode.

FEX Group GIR Functionality

You can use GIR to perform maintenance and software upgrade of the Cisco Nexus 5000, 5500 and 6000 Series switches and the connected FEXs in a dual homed vPC topology. A FEX group is a logical grouping of FEXs. A FEX group is added to optimize the procedure to bring up or take down the FEX.

Consider a scenario (refer figure below) where there are 2 FEX groups, FG1 and FG2 in a VPC domain. Assuming all hosts are redundantly connected, one leg is connected to one of the FEXs of FG1 and the other leg is connected to one of the FEXs of FG2. Before putting the "secondary" switch in maintenance mode, bring down FG1 to force all FEXs in FG1 to upgrade to newer version of image and establish connection to "primary" switch. After the connection to "primary" switch has been established, bring down FEXs of FG2 and then put the "secondary" switch in maintenance mode.

Figure 6: Sample Topology



Note Use the **fex-group** *name* command to create a FEX group. Use the **fex** *range* command to add or remove a FEX from the FEX-group. Use the **system fex-group** *name* **shutdown** command to shut down a FEX group. Use the**no system fex-group** *name* **shutdown** command to bring up a FEX group.

Guidelines and Limitations for GIR

- Custom maintenance profile has to be used for custom topologies and protocols that are not supported by automatic or system-generated profiles.
- Before starting with maintenance, ensure that the switch is not attracting any data traffic after the switch has been put in maintenance mode. You can use counters and statistics to ensure that there is no data traffic on the switch.
- Use the **system mode maintenance always-use-custom-profile** command when using custom profiles to ensure that the custom profile is not overwritten by the system-generated profile.
- Snapshot information is not copied automatically to the standby supervisor in a dual supervisor system.
- GIR may not provide zero application traffic loss for certain topologies and configurations.
- Starting with Cisco NX-OS Release 7.3(0)N1(1), we recommend not using the **configure profile [maintenance-mode | normal-mode] type admin** command and we strongly recommend using the **configure maintenance profile [maintenance-mode | normal-mode]** command.

- You cannot perform an in-service software upgrade (ISSU) or an in-service software downgrade (ISSD) in maintenance mode.

Configuring Custom Maintenance Mode and Custom Normal Mode Profile

You can create the maintenance-mode profile or normal-mode profile with configuration commands that can be applied during graceful removal or graceful insertion. We recommend using the **system mode maintenance always-use-custom-profile** command after configuring custom maintenance mode and custom normal mode profiles to ensure that custom profiles are always used during maintenance mode operations.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure maintenance profile** [**maintenance-mode** | **normal-mode**]	Enters a configuration session for the maintenance-mode profile or the normal-mode profile. **Note** Depending on which protocols you have configured, enter the appropriate commands to bring down the protocols.
Step 2	Required: switch# **end**	Closes the maintenance mode profile.

Example

This example shows how to create a custom maintenance mode profile:

```
switch# configure maintenance profile maintenance-mode
Please configure 'system mode maintenance always-use-custom-profile' if you want to use
custom profile always for maintenance mode.
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-mm-profile)# router bgp 100
switch(config-mm-profile-router)# isolate
switch(config-mm-profile-router)# exit
switch(config-mm-profile)# sleep instance 1 10
switch(config-mm-profile)# interface ethernet 1/1
switch(config-mm-profile-if-verify)# shutdown
switch(config-mm-profile-if-verify)# end
Exit maintenance profile mode.
```

This example shows how to create a custom normal mode profile:

```
switch# configure maintenance profile normal-mode
Please configure 'system mode maintenance always-use-custom-profile' if you want to use
custom profile always for maintenance mode.
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-mm-profile)# interface ethernet 1/1
```

```
switch(config-mm-profile-if-verify)# no shutdown
switch(config-mm-profile-if-verify)# exit
switch(config-mm-profile)# sleep instance 1 20
switch(config-mm-profile)# router bgp 100
switch(config-mm-profile-router)# no isolate
switch(config-mm-profile-router)# end
Exit maintenance profile mode.
```

```
switch# show maintenance profile
[Normal Mode]
interface Ethernet1/1
no shutdown
sleep instance 1 20
router bgp 100
no isolate
[Maintenance Mode]
router bgp 100
isolate
sleep instance 1 20
interface Ethernet1/1
shutdown
```

Creating a Snapshot

You can create a snapshot of the running states of selected features. When you create a snapshot, a predefined set of show commands are run and the outputs are saved.

Procedure

	Command or Action	Purpose
Step 1	switch# **snapshot create** *name description*	Creates a snapshot. The *name* variable can be 64 characters in length. The *description* variable can be 256 characters in length. Use the **snapshot delete** {**all** | *snapshot-name*} command to delete all snapshots or a specific snapshot.
Step 2	(Optional) switch# **show snapshots**	Displays snapshots present on the switch.
Step 3	(Optional) switch# **show snapshots compare** *snapshot-name-1 snapshot-name-2* [**summary**]	Displays a comparison of two snapshots. The **summary** keyword displays just enough information to see the overall changes between the two snapshots.

Example

This example shows how to create a snapshot:

```
switch# snapshot create before_maint taken before maint
Executing 'show interface'... Done
Executing 'show ip route summary vrf all'... Done
Executing 'show ipv6 route summary vrf all'... Done
Executing 'show bgp sessions vrf all'... Done
```

```
Executing 'show ip eigrp topology summary'... Done
Executing 'show ipv6 eigrp topology summary'... Done
Executing 'show vpc'... Done
Executing 'show ip ospf vrf all'... Done
Feature 'ospfv3' not enabled, skipping...
Executing 'show isis database detail vrf all'... Done
Executing 'show ip rip vrf all'... Done
Executing user-specified 'show ip route detail vrf all'... Done
Snapshot 'before_maint' created
```

This example shows how to display the snapshots present on the switch:

```
switch# show snapshots
Snapshot Name           Time                            Description
--------------------------------------------------------------------------------
before_maint            Wed Oct 14 10:56:50 2015       taken before maint
```

This example displays a comparison between two snapshots:

```
switch# show snapshots compare before_maintenance after_maintenance summary
================================================================================
Feature changed                       before_maintenance after_maintenance
================================================================================
basic summary
   # of interfaces                          50               50
   # of vlans                               0                0
   # of ipv4 routes  vrf default            13               13
   # of ipv4 paths   vrf default            13               13
   # of ipv4 routes  vrf management         14               14
   # of ipv4 paths   vrf management         14               14
   # of ipv6 routes  vrf default            3                3
   # of ipv6 paths   vrf default            3                3

interfaces
   # of eth interfaces                      48               48
   # of eth interfaces up                   1                1
   # of eth interfaces down                 47               47
   # of eth interfaces other                0                0

   # of vlan interfaces                     0                0
   # of vlan interfaces up                  0                0
   # of vlan interfaces down                0                0
   # of vlan interfaces other               0                0
```

This example shows how to delete a snapshot:

```
switch# snapshot delete before_maint
switch# show snapshots
Snapshot Name           Time                            Description
--------------------------------------------------------------------------------
```

Adding Show Commands to Snapshots

You can specify additional **show** commands to be captured in snapshots. These **show** commands are defined in user-specified snapshot sections.

Procedure

	Command or Action	Purpose
Step 1	switch# **snapshot section add** *section "show-command" row-id element-key1* [*element-key2*]	Adds a user-specified section to snapshots. The *section* variable is used to name the **show** command output. You can use any word to name the section. The **show** command must be enclosed in quotation marks. Non-**show** commands will not be accepted. The *row-id* argument specifies the tag of each row entry of the **show** command's XML output. The *element-key1* and *element-key2* arguments specify the tags used to distinguish among row entries. In most cases, only the *element-key1* argument needs to be specified to be able to distinguish among row entries. **Note** To delete a user-specified section from snapshots, use the **snapshot section delete** *section* section command.
Step 2	(Optional) switch# **show snapshots sections**	Displays the user-specified snapshot sections.

Example

The following example shows how to add the **show ip route detail vrf all** command to the snapshot:

```
switch# snapshot section add v4route "show ip route detail vrf all" ROW_prefix ipprefix
switch# show snapshots sections
user-specified snapshot sections
--------------------------------
[v4route]
show command: show ip route detail vrf all
row id: ROW_prefix
key1: ipprefix
key2: -
```

The following example shows how to add the **show ipv6 route detail vrf all** command to the snapshot:

```
switch# snapshot section add routev6 "show ipv6 route detail vrf all" ROW_prefix ipprefix
added section "routev6"

switch# show snapshots sections
user-specified snapshot sections
--------------------------------
[routev6]
show command: show ipv6 route detail vrf all
row id: ROW_prefix
key1: ipprefix
key2: -
```

The following example shows how to delete a user-specified snapshot section:

```
switch# snapshot section delete v4route
deleted section ”v4route"

switch# show snapshots sections
user-specified snapshot sections
--------------------------------
none
```

The following example displays the XML output of the **show ip route detail vrf all** command:

```
switch(config)# show ip route detail vrf all | xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<nf:rpc-reply xmlns:nf="urn:ietf:params:xml:ns:netconf:base:1.0" xmlns="http://w
ww.cisco.com/nxos:7.3.0.N1.1.:urib">
 <nf:data>
  <show>
   <ip>
       <__readonly__>
         <TABLE_vrf>
          <ROW_vrf>
           <vrf-name-out>default</vrf-name-out>
           <TABLE_addrf>
            <ROW_addrf>
             <addrf>ipv4</addrf>
             <TABLE_prefix>
              <ROW_prefix>
               <ipprefix>0.0.0.0/32</ipprefix>
               <ucast-nhops>1</ucast-nhops>
               <mcast-nhops>0</mcast-nhops>
               <attached>false</attached>
              … <snip>
              </ROW_prefix>
```

Dumping Snapshot Sections

Procedure

	Command or Action	Purpose
Step 1	switch# **show snapshots dump** *snapshot-name*	Displays the content of the various sections in a generated snapshot.

Example

The following example shows how to dump content of the various sections in a generated snapshot:

```
switch# show snapshots dump new
File:  interface.xml        Snapshot:  new
================================================================================
<?xml version="1.0" encoding="ISO-8859-1"?>
<nf:rpc-reply xmlns:nf="urn:ietf:params:xml:ns:netconf:base:1.0" xmlns="http://w
ww.cisco.com/nxos:7.3.0.N1.1.:if_manager">
 <nf:data>
  <show>
```

```
<interface>
 <__readonly__>
  <TABLE_interface>
   <ROW_interface>
    <interface>mgmt0</interface>
    <state>up</state>
    <admin_state>up</admin_state>
    <eth_hw_desc>GigabitEthernet</eth_hw_desc>
    <eth_hw_addr>5cfc.666d.3b34</eth_hw_addr>
    <eth_bia_addr>5cfc.666d.3b34</eth_bia_addr>
    <eth_ip_addr>5.24.100.101</eth_ip_addr>
    <eth_ip_mask>16</eth_ip_mask>
    <eth_ip_prefix>5.24.0.0</eth_ip_prefix>
    <eth_mtu>1500</eth_mtu>
 … <snip> …
```

Entering Maintenance Mode

If you are going to create your own profile rather than using the system mode maintenance command to do it for you, see the Configuring Custom Maintenance Mode and Custom Normal Mode Profile section.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **system mode maintenance** [**always-use-custom-profile** |**dont-generate-profile** |**on-reload reset-reason** *reason* |**shutdown** |**timeout** *value*]	Puts all enabled protocols in maintenance mode (using the **isolate** command). Use the **dont-generate-profile** and **shutdown** options to put the switch in maintenance mode. • **dont-generate-profile**—Prevents the dynamic searching of enabled protocols and executes commands configured in a maintenance mode profile. Use this option if you want the system to execute commands in a custom maintenance mode profile. • **shutdown**—Shuts down all protocols and interfaces except the management interface (using the **shutdown** command). This option is disruptive while the default (using the **isolate** command) is not. The **on-reload reset-reason**, **timeout** and **always-use-custom-profile** options are used to configure maintenance mode parameters and will not put the switch in maintenance mode. • **timeout** *value*—Keeps the switch in maintenance mode for a specified number of minutes. The range is from 5 to 65535.

	Command or Action	Purpose
		We recommend setting the timeout value to at least 60 minutes. Once the configured time elapses, the switch returns to normal mode automatically. The **no system mode maintenance timeout** command disables the timer • **on-reload reset-reason** *reason*—Boots the switch into maintenance mode automatically in the event of a specified system crash. The **no system mode maintenance on-reload reset-reason** command prevents the switch from being brought up in maintenance mode in the event of a system crash. The maintenance mode reset reasons are as follows: • HW_ERROR—Hardware error • SVC_FAILURE—Critical service failure • KERN_FAILURE—Kernel panic • WDOG_TIMEOUT—Watchdog timeout • FATAL_ERROR—Fatal error • MANUAL_RELOAD---Manual reload • MAINTENANCE—Reloads the switch in maintenance mode if the switch was already in maintenance mode before reload. • MATCH_ANY—Any of the above reasons • ANY_OTHER—Any reload reason not specified above. The system prompts you to continue. Enter y to continue or n to terminate the process. **Note** We recommend configuring the reset reason and saving it to the startup configuration. This enables the switch to go into the maintenance mode after a switch reloads due to any reason.

	Command or Action	Purpose
		• **always-use-custom-profile**—Use this option to apply the existing custom maintenance mode profile and prevent creation of autogenerated maintenance mode profile. This option forces the dont-generate-profile option to be used even if it has not been specified using the **system mode maintenance** command. You cannot use the "shutdown" option when this option is being used.
Step 3	(Optional) switch# **show system mode**	Displays the current system mode. This command also displays the current state of the maintenance mode timer when the switch is in maintenance mode.

Example

This example shows how to put all the protocols in maintenance mode using the **system mode maintenance** command on a switch running the Cisco NX-OS Release 7.3(0)N1(1):

```
switch# configure terminal
switch(config)# system mode maintenance
Following configuration will be applied:

router bgp 100
  isolate
router ospf 100
  isolate
router isis 100
  isolate

Do you want to continue (y/n)? [no] y

Generating a snapshot before going into maintenance mode

Starting to apply commands...

Applying : router bgp 100
Applying :   isolate
Applying : router ospf 100
Applying :   isolate
Applying : router isis 100
Applying :   isolate

Maintenance mode operation successful.
```

This example shows how to shut down all protocols and interfaces on the switch:

```
switch# configure terminal
switch(config)# system mode maintenance shutdown
Following configuration will be applied:

router bgp 64581
  shutdown
```

```
router eigrp p2
  shutdown
  address-family ipv6 unicast
    shutdown
router eigrp 0
  shutdown
  address-family ipv6 unicast
    shutdown
router ospf 200
  shutdown
router isis 70
  shutdown
vpc domain 2
  shutdown
system interface shutdown

NOTE: 'system interface shutdown' will shutdown all interfaces excluding mgmt 0
Do you want to continue (yes/no)? [no] yes

Generating a snapshot before going into maintenance mode

Starting to apply commands...

Applying : router bgp 64581
Applying :   shutdown
Applying : router eigrp p2
Applying :   shutdown
Applying :   address-family ipv6 unicast
Applying :     shutdown
Applying : router eigrp 0
Applying :   shutdown
Applying :   address-family ipv6 unicast
Applying :     shutdown
Applying : router ospf 200
Applying :   shutdown
Applying : router isis 70
Applying :   shutdown
Applying : vpc domain 2
Applying :   shutdown2016 Jan 15 11:10:36.080386 CP-BL26-N7K-1A %$ VDC-1 %$
%VPC-2-VPC_SHUTDOWN: vPC shutdown status is ON

Applying : system interface shutdown

Maintenance mode operation successful.
switch(config)# 2016 Jan 15 11:10:42.057678 switch %$ VDC-1 %$ %MMODE-2-MODE_CHANGED: System
 changed to "maintenance" mode.
2016 Jan 15 11:10:42.058167 switch %$ VDC-1 %$ %MMODE-2-MODE_CHANGE_WARN: System will be
moved to "normal" mode in 5 minutes
```

This example shows how to keep the switch in maintenance mode for a specific number of minutes:

```
switch# configure terminal
switch (config)# system mode maintenance timeout 25

switch# show system mode
System Mode: Maintenance
Maintenance Mode Timer: 24 minutes 55 seconds remaining
```

This example shows how to automatically boot the switch into maintenance mode if a fatal error occurs:

```
switch# configure terminal
switch(config)# system mode maintenance on-reload reset-reason fatal_error
```

This example shows how to place the switch in maintenance mode by using a previously created maintenance mode profile :

```
switch# configure terminal
switch(config)# system mode maintenance dont-generate-profile

Following configuration will be applied:

router bgp 100
  isolate
sleep instance 1 10
interface Ethernet1/1
  shutdown

Do you want to continue (y/n)? [no] y

Generating a snapshot before going into maintenance mode

Starting to apply commands...

Applying : router bgp 100
Applying :   isolate
Applying : sleep instance 1 10
Applying : interface Ethernet1/1
Applying :   shutdown

Maintenance mode operation successful.
```

This example shows how to apply the existing custom maintenance mode profile and prevent creation of auto-generated maintenance mode profile:

```
switch# configure terminal
switch(config)# system mode maintenance always-use-custom-profile
```

This example shows how to put the switch in maintenance mode without presenting any switch prompts:

```
switch# configure terminal
switch(config)# system mode maintenance non-interactive
System mode switch to maintenance mode started. Will continue in background.
switch(config)# 2016 Dec  5 08:46:42 switch %$ VDC-1 %$ %MMODE-2-MODE_CHANGED: System changed
 to "maintenance" mode.

switch(maint-mode)(config)#
```

This example shows how to change the snapshot delay timer value:

```
switch# configure terminal
switch(config)# system mode maintenance snapshot-delay 150
```

Returning to Normal Mode

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	Required: switch# **no system mode maintenance** [**dont-generate-profile** | **non-interactive**]	Executes a previously created normal mode profile file or a dynamically created normal mode profile file. The **dont-generate-profile** keyword suppresses the creation of the normal mode maintenance profile and also prevents reusing the existing normal mode maintenance profile. The **non-interactive** keyword enables the switch to exit the maintenance mode without presenting any switch prompts. The system prompts you to continue. Enter **y** to continue or **n** to terminate the process. **Note** For large configurations, the interfaces will be up after a certain interval of time.

Example

This example shows how to return to normal mode from maintenance mode on a switch running the Cisco NX-OS Release 7.3(0)N1(1):

```
switch# configure terminal
switch(config)# no system mode maintenance
Following configuration will be applied:

interface Ethernet1/1
  no shutdown
sleep instance 1 20
router bgp 100
  no isolate

Do you want to continue (y/n)? [no] yes

Starting to apply commands...

Applying : interface Ethernet1/1
Applying :   no shutdown
Applying : sleep instance 1 20
Applying : router bgp 100
Applying :   no isolate

Maintenance mode operation successful.

Generating Current Snapshot

Please use 'show snapshots compare before_maintenance after_maintenance' to check the health
```

```
 of the system
switch(config)#


switch(config)# show system mode
System Mode: Normal
```

This example shows how to return to normal mode from maintenance mode by using the **dont-generate-profile** keyword:

```
switch(config)# no system mode maintenance dont-generate-profile
Following configuration will be applied:

interface Ethernet1/1
  no shutdown
sleep instance 1 20
router bgp 100
  no isolate

Do you want to continue (y/n)? [no] yes

Starting to apply commands...

Applying : interface Ethernet1/1
Applying :   no shutdown
Applying : sleep instance 1 20
Applying : router bgp 100
Applying :   no isolate

Maintenance mode operation successful.

The after_maintenance snapshot will be generated in 120 seconds
After that time, please use 'show snapshots compare before_maintenance after_maintenance'
to check the health of the system
switch(config)# 2016 Dec  5 08:51:46 switch %$ VDC-1 %$ %MMODE-2-MODE_CHANGED: System changed
 to "normal" mode.


switch(config)# show system mode
System Mode: Normal
```

Deleting a Maintenance Profile

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	Required: switch# **no configure maintenance profile** {**normal-mode** | **maintenance-mode**}	Deletes the normal mode or maintenance mode profiles.

Example

This example shows how to delete a maintenance profile:

```
switch# configure terminal
switch(config)# no configure maintenance profile maintenance-mode
```

Configuration Examples for GIR

This example shows how to create custom maintenance mode profile:

```
switch# configure maintenance profile maintenance-mode
Please configure 'system mode maintenance always-use-custom-profile' if you want to use
custom
profile always for maintenance mode.
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-mm-profile)# router bgp 100
switch(config-mm-profile-router)# isolate
switch(config-mm-profile-router)# exit
switch(config-mm-profile)# sleep instance 1 10
switch(config-mm-profile)# interface ethernet 1/1
switch(config-mm-profile-if-verify)# shutdown
switch(config-mm-profile-if-verify)# end
Exit maintenance profile mode.
```

This example shows how to create custom normal mode profile:

```
switch# configure maintenance profile normal-mode
Please configure 'system mode maintenance always-use-custom-profile' if you want to use
custom
profile always for maintenance mode.
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-mm-profile)# interface ethernet 1/1
switch(config-mm-profile-if-verify)# no shutdown
switch(config-mm-profile-if-verify)# exit
switch(config-mm-profile)# sleep instance 1 20
switch(config-mm-profile)# router bgp 100
switch(config-mm-profile-router)# no isolate
switch(config-mm-profile-router)# end
Exit maintenance profile mode.
```

This example shows how to create a custom maintenance mode and normal mode profile for IPv6 protocols:

```
switch# configure terminal
switch(config)# configure maintenance profile maintenance-mode
Please configure 'system mode maintenance always-use-custom-profile' if you want to use
custom
profile always for maintenance mode.
switch(config-mm-profile)# router ospfv3 ospf_ipv6
switch(config-mm-profile-router)# shutdown
switch(config-mm-profile-router)# exit
switch(config-mm-profile)# router eigrp 660
switch(config-mm-profile-router)# address-family ipv6 unicast
switch(config-mm-profile-router-af)# shutdown
switch(config-mm-profile-router-af)# exit
switch(config-mm-profile)# router isis isp
switch(config-mm-profile-router)# set-overload-bit always
switch(config-mm-profile-router)# address-family ipv6 unicast
switch(config-mm-profile-router-af)# shutdown
switch(config-mm-profile-router-af)# exit


switch# configure terminal
switch(config)# configure maintenance profile normal-mode
```

```
Please configure 'system mode maintenance always-use-custom-profile' if you want to use
custom
profile always for maintenance mode.
switch(config-mm-profile)# router isis isp
switch(config-mm-profile-router)# no set-overload-bit always
switch(config-mm-profile-router)# address-family ipv6 unicast
switch(config-mm-profile-router-af)# no shutdown
switch(config-mm-profile-router-af)# exit
switch(config-mm-profile)# router eigrp 660
switch(config-mm-profile-router)# address-family ipv6 unicast
switch(config-mm-profile-router-af)# no shutdown
switch(config-mm-profile-router-af)# exit
switch(config-mm-profile)# router ospfv3 ospf_ipv6
switch(config-mm-profile-router)# no shutdown
switch(config-mm-profile-router)# exit


switch# show maintenance profile
[Normal mode]
router isis isp
  no set-overload-bit always
  address-family ipv6 unicast
   no shutdown
router eigrp 660
  address-family ipv6 unicast
   no shutdown
router ospfv3 ospf_ipv6
   no shutdown
[Maintenance Mode]
router ospfv3 ospf_ipv6
  shutdown
router eigrp 660
  address-family ipv6 unicast
   shutdown
router isis isp
  set-overload-bit always
  address-family ipv6 unicast
   shutdown
```

This example shows how to create a custom maintenance mode profile and custom normal mode profile for VPC:

```
switch# configure terminal
switch(config)# configure maintenance profile maintenance-mode
switch(config-mm-profile)# router bgp 100
switch(config-mm-profile-router)# isolate
switch(config-mm-profile-router)# exit
switch(config-mm-profile)# interface port channel 5
switch(config-mm-profile-if-verify)# vpc orphan port suspend
switch(config-mm-profile-if-verify)# exit
switch(config-mm-profile)# interface port channel 6
switch(config-mm-profile-if-verify)# vpc orphan port
suspend switch(config-mm-profile-if-verify)# exit
switch(config-mm-profile)# sleep instance 1 5
switch(config-mm-profile)# vpc domain 1
switch(config-mm-profile-vpc-domain)# shutdown


switch# configure terminal
switch(config)# configure maintenance profile normal-mode
switch(config-mm-profile)# vpc domain 1
switch(config-mm-profile-vpc-domain)# no shutdown
switch(config-mm-profile-vpc-domain)# exit
switch(config-mm-profile)# sleep instance 1 60
```

```
switch(config-mm-profile)# interface port channel 5
switch(config-mm-profile-if-verify)# no vpc orphan port suspend
switch(config-mm-profile-if-verify)# exit
switch(config-mm-profile)# interface port channel 6
switch(config-mm-profile-if-verify)# no vpc orphan port suspend
switch(config-mm-profile-if-verify)# exit
switch(config-mm-profile)# router bgp 100
switch(config-mm-profile-router)# no isolate


switch# show maintenance profile
[Normal Mode]
vpc domain 1
  no shutdown
sleep instance 1 60
interface port-channel 5
  no vpc orphan-port suspend
interface port-channel 6
  no vpc orphan-port suspend router
bgp 100
  no isolate

[Maintenance Mode]
router bgp 100
  isolate
interface port-channel 5 vpc
  orphan-port suspend
interface port-channel 6 vpc
  orphan-port suspend
sleep instance 1 5
  vpc domain 1 shutdown
```

This example shows how to use the **isolate** command to put all protocols into maintenance mode:

```
switch(config)# system mode maintenance

Following configuration will be applied:

router bgp 100
  isolate
router ospf 100
  isolate
router isis 100
  isolate

Do you want to continue (y/n)? [no] y

Generating a snapshot before going into maintenance mode

Starting to apply commands...

Applying : router bgp 100
Applying :   isolate
Applying : router ospf 100
Applying :   isolate
Applying : router isis 100
Applying :   isolate

Maintenance mode operation successful.
```

This example shows how to shut down all protocols and interfaces on the switch:

```
switch# configure terminal
switch(config)# system mode maintenance shutdown

Following configuration will be applied:

router bgp 64581
  shutdown
router eigrp p2
  shutdown
  address-family ipv6 unicast
    shutdown
router eigrp 0
  shutdown
  address-family ipv6 unicast
    shutdown
router ospf 200
  shutdown
router isis 70
  shutdown
vpc domain 2
  shutdown
system interface shutdown

NOTE: 'system interface shutdown' will shutdown all interfaces excluding mgmt 0
Do you want to continue (yes/no)? [no] yes

Generating a snapshot before going into maintenance mode

Starting to apply commands...

Applying : router bgp 64581
Applying :   shutdown
Applying : router eigrp p2
Applying :   shutdown
Applying :   address-family ipv6 unicast
Applying :     shutdown
Applying : router eigrp 0
Applying :   shutdown
Applying :   address-family ipv6 unicast
Applying :     shutdown
Applying : router ospf 200
Applying :   shutdown
Applying : router isis 70
Applying :   shutdown
Applying : vpc domain 2
Applying :   shutdown2016 Jan 15 11:10:36.080386 CP-BL26-N7K-1A %$ VDC-1 %$
%VPC-2-VPC_SHUTDOWN: vPC shutdown status is ON

Applying : system interface shutdown

Maintenance mode operation successful.
switch(config)# 2016 Jan 15 11:10:42.057678 CP-BL26-N7K-1A %$ VDC-1 %$ %MMODE-2-MODE_CHANGED:
 System changed to "maintenance" mode.
2016 Jan 15 11:10:42.058167 CP-BL26-N7K-1A %$ VDC-1 %$ %MMODE-2-MODE_CHANGE_WARN: System
will be moved to "normal" mode in 5 minutes
```

This example shows how to return to normal mode from maintenance mode:

```
switch# configure terminal
switch(config)# no system mode maintenance dont-generate-profile

Following configuration will be applied:
interface Ethernet1/1
  no shutdown
```

```
  sleep instance 1 20
router bgp 100
  no isolate
Do you want to continue (y/n)? [no] yes
Starting to apply commands...
Applying : interface Ethernet1/1
Applying :   no shutdown
Applying : sleep instance 1 20
Applying : router bgp 100
Applying :   no isolate
Maintenance mode operation successful.
Generating Current Snapshot
Please use 'show snapshots compare before_maintenance after_maintenance' to check the
health of the system
```

This example shows how to create custom maintenance mode and normal mode profiles for FabricPath:

```
switch# configure maintenance profile maintenance-mode
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-mm-profile)# fabricpath domain default
switch(config-mm-profile-fabricpath-isis)# set-overload-bit always
switch(config-mm-profile-fabricpath-isis)# end
Exit maintenance profile mode.
switch#

switch# configure maintenance profile normal-mode
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-mm-profile)# fabricpath domain default
switch(config-mm-profile-fabricpath-isis)# no set-overload-bit always
switch(config-mm-profile-fabricpath-isis)# end
Exit maintenance profile mode.
switch#


switch# show maintenance profile
[Normal Mode]
fabricpath domain default
  no set-overload-bit always
[Maintenance Mode]
fabricpath domain default
  set-overload-bit always
```

This example shows how to create custom maintenance mode and normal mode profiles for a virtual Port Channel (vPC):

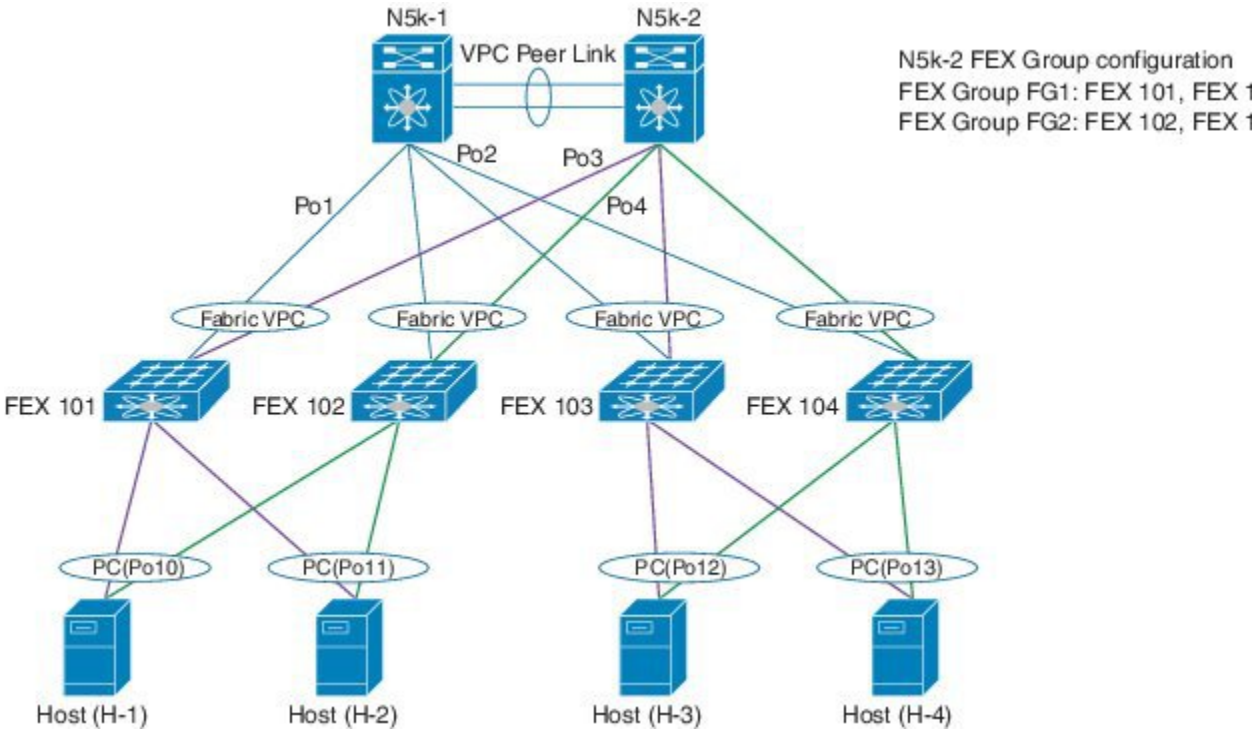
```
switch# configure maintenance profile maintenance-mode
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-mm-profile)# vpc domain 1
switch(config-mm-profile-vpc-domain)# shutdown
switch(config-mm-profile-vpc-domain)# exit
switch(config-mm-profile)# system interface shutdown
switch(config-mm-profile)# end
Exit maintenance profile mode.
switch#


switch# configure maintenance profile normal-mode
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-mm-profile)# vpc domain 1
switch(config-mm-profile-vpc-domain)# no shutdown
switch(config-mm-profile-vpc-domain)# exit
switch(config-mm-profile)# no system interface shutdown
switch(config-mm-profile)# end
```

```
Exit maintenance profile mode.
switch#
```

```
switch# show maintenance profile
[Normal Mode]
vpc domain 1
  no shutdown
no system interface shutdown
[Maintenance Mode]
vpc domain 1
  shutdown
system interface shutdown
```

Note Use the **fex-group** *name* command to create a FEX group. Use the **fex** *range* command to add or remove a FEX from the FEX-group. Use the **system fex-group** *name* **shutdown** command to shut down a FEX group. Use the**no system fex-group** *name* **shutdown** command to bring up a FEX group

This example shows how to create a maintenance mode profile and normal mode profile for upgrading vPC with FEX (refer topology below):



N5K-1 configuration:

```
switch# configure terminal
switch(config)# configure maintenance profile maintenance-mode
switch(config-mm-profile)# vpc domain 1
switch(config-mm-profile-vpc-domain)# shutdown
switch(config-mm-profile)# system interface shutdown
switch# configure terminal
switch(config)# configure maintenance profile normal-mode
```

```
switch(config-mm-profile)# no system interface shutdown
switch(config-mm-profile)# vpc domain 1
switch(config-mm-profile-vpc-domain)# no shutdown
```

N5K-2 configuration:

```
switch# configure terminal
switch(config)# configure maintenance profile maintenance-mode
switch(config-mm-profile)# system fex-group fg1 shutdown
switch(config-mm-profile)# sleep 900
switch(config-mm-profile)# system fex-group fg2 shutdown
switch(config-mm-profile)# vpc domain 1
switch(config-mm-profile-vpc-domain)# shutdown
switch(config-mm-profile-vpc-domain)# exit
switch(config-mm-profile)# system interface shutdown

switch# configure terminal
switch(config)# configure maintenance profile normal-mode
switch(config-mm-profile)# no system interface shutdown
switch(config-mm-profile)# vpc domain 1
switch(config-mm-profile-vpc-domain)# no shutdown
switch(config-mm-profile-vpc-domain)#  exit
switch(config-mm-profile)# no system fex-group fg2 shutdown
switch(config-mm-profile)# no system fex-group fg1 shutdown
```

This example shows the configuration to be used when there are port-channel or regular L2 ethernet interfaces (except vPC peer link) which carry vPC VLAN traffic and when the corresponding Switch Virtual Interface (SVI) state should not be controlled by these interfaces:

```
Port-channel configuration
switch(config)# interface port-channel3
switch(config-if)# description "L2-Cross Link eth3/3 eth4/3 eth5/3 eth6/3"
switch(config-if)# switchport
switch(config-if)# switchport mode trunk
switch(config-if)# switchport trunk allowed vlan 1101-1500
switch(config-if)# spanning-tree port type network
switch(config-if)# lacp min-links 2
switch(config-if)# switchport autostate exclude vlan 1101-1500


L2 Ethernet configuration
switch(config)# interface ethernet 3/3
switch(config-if)# switchport
switch(config-if)# switchport mode trunk
switch(config-if)# switchport trunk allowed vlan 1101-1500
switch(config-if)# switchport autostate exclude vlan 1101-1500
```

The "redistribute direct" configuration under Border Gateway Protocol (BGP) will attract traffic as the BGP **isolate** mode does not withdraw direct routes. This example shows how to use the **route-map** command to enable BGP to withdraw direct routes in **isolate** mode:

Policy Configuration

Use **route-map my-rmap-deny** in maintenance mode configuration to exclude SVIs having tag 200 configuration.

```
switch(config)# route-map my-rmap-deny deny 10
switch(config-route-map)# match tag 200
switch(config-route-map)# exit
switch(config)# route-map my-rmap-deny permit 20
```

Use **route-map my-rmap-permit** in normal mode configuration to include SVIs having tag 200 configuration.

```
switch(config)# route-map my-rmap-permit permit 10
switch(config-route-map)# match tag 200
switch(config-route-map)# exit
switch(config)# route-map my-rmap-permit permit 20
```

Virtual IP (vIP)/ Switch Virtual Interface (SVI) configuration

```
switch(config)# interface loopback 200
switch(config-if)# ip address 192.0.2.100/8 tag 200
switch(config)# interface vlan 2
switch(config-if)# ip address 192.0.2.108/8 tag 200
....
switch(config)# interface vlan 3
switch(config-if)# ip address 192.0.2.102/8 tag 200
```

BGP configuration

```
switch(config)# feature bgp
switch(config)# router bgp 100
switch(config-router)# neighbor 192.0.2.100
....
```

Maintenance mode profile

```
switch# configure maintenance profile maintenance-mode
switch(config-mm-profile)# router bgp 200
switch(config-mm-profile-router)# address-family ipv4 unicast
switch(config-mm-profile-router-af)# redistribute direct route-map my-rmap-deny
switch(config-mm-profile-router-af)# exit
switch(config-mm-profile)# sleep instance 1 10
```

Normal mode profile

```
switch# configure maintenance profile normal-mode
switch(config-mm-profile)# router bgp 100
switch(config-mm-profile-router)# address-family ipv4 unicast
switch(config-mm-profile-router-af)# redistribute direct route-map my-rmap-permit
switch(config-mm-profile-router-af)# exit
switch(config-mm-profile)# sleep instance 1 20
```

Verifying GIR

Use the following commands to verify the configuration:

Command	Purpose
show interface brief	Displays abbreviated interface information.
show maintenance on-reload reset-reason	Displays the reset reasons for which the switch comes up in maintenance mode.
show maintenance profile [maintenance-mode | normal-mode]	Displays the details of the maintenance mode or normal mode profile.
show maintenance snapshot-delay	Displays the after_maintenance snapshot-delay timer value.

Command	Purpose
show maintenance timeout	Displays the maintenance mode timeout period, after which the switch automatically returns to normal mode.
show tech-support mmode	Displays maintenance mode information for Cisco technical support.
show {running--config | startup--config} mmode [all]	Displays the maintenance-mode section of the running or startup configuration. The **all** option includes the default values.
show snapshots	Displays snapshots present on the switch.
show snapshots compare *snapshot-name-1 snapshot-name-2* [**summary | ipv4routes | ipv6routes**]	Displays a comparison of two snapshots. The **summary** option displays just enough information to see the overall changes between the two snapshots. The **ipv4routes** and the **ipv6routes** options display the changes in IPv4 and IPv6 routes between the two snapshots.
show snapshots dump *snapshot-name*	Displays content of the various sections in a generated snapshot.
show snapshots sections	Displays the user-specified snapshot sections.
show system mode	Displays the current system mode. This command also displays the current state of the maintenance mode timer when the switch is in maintenance mode.

Verifying GIR at Protocol Level

BGP (Maintenance mode)

Use the **show bgp process** command to display BGP status in maintenance mode:

```
switch# show bgp process

BGP Process Information
BGP Process ID                 : 11725
BGP Protocol Started, reason:  : configuration
BGP Protocol Tag               : 100
BGP Protocol State         : Running (Isolate)
BGP MMODE                  : Initialized
BGP Memory State               : OK
BGP asformat                   : asplain

BGP attributes information
Number of attribute entries    : 1
HWM of attribute entries       : 1
Bytes used by entries          : 100
Entries pending delete         : 0
HWM of entries pending delete  : 0
BGP paths per attribute HWM    : 3
BGP AS path entries            : 0
```

```
Bytes used by AS path entries  : 0
```

Use the **show bgp internal all statistics** command to display the number of BGP IPv4 and IPv6 prefixes that have been programmed and also the number of BGP IPv4 and IPv6 prefixes that have not been programmed:

```
BGP internal statistics information for VRF default, address family IPv4 Unicast
  Total prefixes in BGP Table: 3
  Total prefixes pending programming in HW: 0
BGP internal statistics information for VRF default, address family IPv6 Unicast
  Total prefixes in BGP Table: 0
  Total prefixes pending programming in HW: 0
```

EIGRP (Maintenance mode)

Use the **show ip eigrp** command to display EIGRP status in maintenance mode:

```
switch# show ip eigrp
IP-EIGRP AS 100 ID 30.1.1.1 VRF default
  Process-tag: 100
  Instance Number: 1
  Status: running (isolate)
  Authentication mode: none
  Authentication key-chain: none
  Metric weights: K1=1 K2=0 K3=1 K4=0 K5=0
  IP proto: 88 Multicast group: 224.0.0.10
  Int distance: 90 Ext distance: 170
  Max paths: 8
  Number of EIGRP interfaces: 1 (0 loopbacks)
  Number of EIGRP passive interfaces: 0
  Number of EIGRP peers: 1
  Redistributing:
    direct route-map passall
    static route-map passall
  Graceful-Restart: Enabled
  Stub-Routing: Disabled
  NSF converge time limit/expiries: 120/0
  NSF route-hold time limit/expiries: 240/6
  NSF signal time limit/expiries: 20/0
  Redistributed max-prefix: Disabled
  MMODE: Initialized
  Suppress-FIB-Pending Configured
```

ISIS (Maintenance mode)

Use the **show isis protocol** command to display ISIS status in maintenance mode:

```
switch# show isis protocol
ISIS process : 100
 Instance number :  1
 UUID: 1090519320
 Process ID 6969
VRF: default
  System ID : 0300.0000.0004  IS-Type : L2
  SAP : 412  Queue Handle : 16
  Maximum LSP MTU: 1492
  Stateful HA enabled
  Graceful Restart enabled. State: Inactive
  Last graceful restart status : none
  Start-Mode Complete
  BFD IPv4 is globally disabled for ISIS process: 100
  BFD IPv6 is globally disabled for ISIS process: 100
  Topology-mode is base
  Metric-style : advertise(wide), accept(narrow, wide)
```

```
  Area address(es) :
    10
  Process is up and running (isolate)
  VRF ID: 1
  Stale routes during non-graceful controlled restart
  Interfaces supported by IS-IS :
    Ethernet1/2
```

OSPF (Maintenance mode)

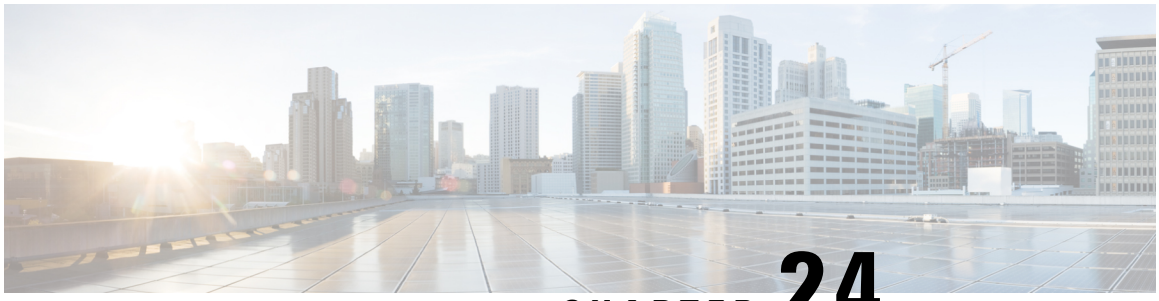
Use the **show ip ospf internal** command to display OSPF status in maintenance mode:

```
switch# show ip ospf internal

ospf 100
ospf process tag 100
ospf process instance number 1
ospf process uuid 1090519321
ospf process linux pid 6968
ospf process state running (isolate)
System uptime 6d06h
SUP uptime 2 6d06h

Server up : L3VM|IFMGR|RPM|AM|CLIS|URIB|U6RIB|IP|IPv6|SNMP|MMODE
Server required  : L3VM|IFMGR|RPM|AM|CLIS|URIB|IP|SNMP
Server registered: L3VM|IFMGR|RPM|AM|CLIS|URIB|IP|SNMP|MMODE
Server optional  : MMODE

Early hello : OFF
Force write PSS: FALSE
OSPF mts pkt sap 324
OSPF mts base sap 320
```

Feature History for GIR

The table below summarizes the new and changed features for this document and shows the releases in which each feature is supported. Your software release might not support all the features in this document. For the latest caveats and feature information, see the Bug Search Tool at https://tools.cisco.com/bugsearch/ and the release notes for your software release.

Feature Name	Release	Information
Graceful Insertion and Removal (GIR)	7.3(0)N1(1)	The default mode for GIR is “isolate”. Support for Unplanned Maintenance, Maintenance Mode timer, Supress FIB Pending, Adding Show commands to snapshots and dumping snapshot sections.
Graceful Insertion and Removal (GIR)	7.1(0)N1(1)	This feature was introduced. The default mode for GIR is “shutdown”. Refer Configuring GIR (Cisco NX-OS Release 7.1(0)N1(1)).



CHAPTER 24

Configuring GIR (Cisco NX-OS Release 7.1(0)N1(1))

This chapter contains the following section

Information About GIR

You can use Graceful Insertion and Removal (GIR) to isolate a switch from the network in order to perform debugging or an upgrade. When switch maintenance is complete, you can return the switch to normal mode.

When you place the switch in maintenance mode, all protocols are gracefully brought down and all physical ports are shut down. When normal mode is restored, all the protocols and ports are brought back up.

The following protocols are supported:

- Border Gateway Protocol (BGP)
- BGPv6
- Enhanced Interior Gateway Routing Protocol (EIGRP)
- EIGRPv6
- Intermediate System-to-Intermediate System (ISIS)
- Open Shortest Path First (OSPF)
- OSPFv3

Also supported are:

- Virtual port channel (vPC) switches

Note GIR is not supported on vPC+ switches.

- Interfaces
- FabricPath

You can create a maintenance mode profile file before you put the switch in maintenance mode or you can allow the system to create a maintenance mode profile file when you enter the **[no] system mode maintenance** command.

Use the **snapshot** command to capture the running states of selected features and to store them on the persistent storage media.

Snapshots are useful to compare the state of a switch before it went into maintenance mode and after it came back to normal mode. The snapshot process consists of three parts:

- Creating a snapshot of the states of a few preselected features on the switch and storing them on the persistent storage media.
- Listing the snapshots taken at various time intervals and managing them.
- Comparing snapshots and showing the summary and details of each feature.

Guidelines and Limitations for GIR

Graceful Insertion and Removal (GIR) has the following guidelines and limitations:

- You can create maintenance mode or normal-mode profile files by using the **config profile maintenance-mode type admin** and **config profile normal-mode type admin** commands respectively.
- We recommend not using GIR maintenance mode when a switch is being upgraded to Cisco NX-OS Release 7.3(0)N1(1) or higher from any release prior to Cisco NX-OS Release 7.3(0)N1(1). If a switch that is in GIR maintenance mode has completed upgrading to Cisco NX-OS Release 7.3(0)N1(1) or higher from any release prior to Cisco NX-OS Release 7.3(0)N1(1), the switch will be isolated from the network. In such a scenario, the GIR maintenance mode configuration that was applied before upgrading the switch has to be manually removed from the switch to restore normal operation. This process of manually removing the GIR maintenance mode configuration will have to be done whenever the switch is rebooted.

  However, this issue can be permanently resolved by downgrading the switch to the previous version, which can be any release prior to Cisco NX-OS Release 7.3(0)N1(1), removing the GIR configuration, and then upgrading to Cisco NX-OS Release 7.3(0)N1(1) or higher without GIR.

Performing the GIR Cycle

Procedure

Step 1 (Optional) Create the maintenance mode profile file.

See Configuring the Maintenance Mode Profile File, on page 297.

Step 2 (Optional) Create the normal mode profile file.

See Configuring the Normal Mode Profile File, on page 293.

Step 3 Take a snapshot before entering maintenance mode.

See Creating a Snapshot, on page 295.

Step 4 Put the switch into maintenance mode.

See Entering Maintenance Mode, on page 295 .

Step 5 (Optional) Enter the **copy running-config startup-config** command.

Step 6 Return the switch to normal mode.

See Returning to Normal Mode, on page 296.

Step 7 Take a snapshot after returning to normal mode.

See Creating a Snapshot, on page 295.

Configuring the Normal Mode Profile File

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **configure profile normal-mode type admin**	Enters a configuration session for the normal mode profile file. **Note** Depending on which protocols you have configured, you must now enter the appropriate commands to bring up the protocols.
Step 3	switch# **end**	Closes the normal mode profile file.

Example

This example shows how to create a normal mode profile file:

```
switch# configure terminal
switch(config)# configure profile normal-mode type admin
switch(config-profile)# router ospf 100
switch(config-profile-router)# no shutdown
switch(config-profile-router)# exit
switch(config-profile)# router eigrp 101
switch(config-profile-router)# no shutdown
switch(config-profile-router)# exit
switch(config-profile)# router isis 102
switch(config-profile-router)# no shutdown
switch(config-profile-router)# no set-overload-bit always
switch(config-profile-router)# exit
switch(config-profile)# router bgp 103
switch(config-profile-router)# no shutdown
switch(config-profile-router)# exit
switch(config-profile)# vpc domain 20
switch(config-profile-router)# no shutdown
switch(config-profile-router)# exit
switch(config-profile)# no system interface shutdown
switch(config-profile)# end
Exit configure profile mode.
switch#
```

This example shows how to create a normal mode custom profile file:

```
switch# configure terminal
switch(config)# configure profile normal-mode type admin
switch(config-profile)# router bgp 65501
switch(config-profile-router)# no shutdown
switch(config-profile-router)# exit
switch(config-profile)# router eigrp 100
switch(config-profile-router)# no shutdown
switch(config-profile-router)# exit
switch(config-profile)# address-family ipv6 unicast
switch(config-profile)# no shutdown
switch(config-profile)# router eigrp 600
switch(config-profile-router)# no shutdown
switch(config-profile-router)# exit
switch(config-profile)# address-family ipv6 unicast
switch(config-profile-router)# no shutdown
switch(config-profile-router)# exit
switch(config-profile)# router ospf 100
switch(config-profile-router)# no shutdown
switch(config-profile-router)# exit
switch(config-profile)# router ospfv3 ospf_ipv6
switch(config-profile-router)# no shutdown
switch(config-profile-router)# exit
switch(config-profile)# router isis isp
switch(config-profile-router)# no set-overload-bit always
switch(config-profile-router)# exit
switch(config-profile)# vpc domain 2
switch(config-profile-router)# no shutdown
switch(config-profile-router)# exit
switch(config-profile)# no system interface shutdown
switch(config-profile)# end
Exit configure profile mode.
switch#
```

Creating a Snapshot

Procedure

	Command or Action	Purpose
Step 1	switch# **snapshot create** *name description*	Creates a snapshot. The *name* variable can be 64 characters in length. The *description* variable can be 256 characters in length.

Example

This example shows how to create a snapshot:

```
switch# snapshot create snap1 For documentation purposes.
Executing show interface... Done
Executing show bgp sessions vrf all... Done
Executing show ip eigrp topology summary... Done
Executing show ipv6 eigrp topology summary... Done
Executing show vpc... Done
Executing show ip ospf vrf all... Done
Feature 'ospfv3' not enabled, skipping...
Executing show isis vrf all... Done
Snapshot 'snap1' created
switch#
```

Entering Maintenance Mode

Before you begin

If you are going to create your own profile rather than let the **system mode maintenance** command do it for you, see Configuring the Maintenance Mode Profile File, on page 297.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **system mode maintenance** [**dont-generate-profile**]	Executes a previously created maintenance mode profile file or dynamically creates a maintenance mode profile file. The **dont-generate-profile** option suppresses the creation of the maintenance mode profile file. **Note** The system prompts you to continue. Enter **y** to continue or **n** to terminate the process.

The switch is now in maintenance mode.

Note It is not possible to perform an in-service software downgrade (ISSD) in maintenance mode.

Example

This example shows how to place the switch in maintenance mode by using a previously created maintenance mode profile file:

```
switch# configure terminal
switch(config)# system mode maintenance dont-generate-profile
Do you want to continue (y/n)? [n] y

Progressing.....................................Done.

System mode operation completed successfully
switch(config)#
```

Returning to Normal Mode

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **no system mode maintenance** [**dont-generate-profile**]	Executes a previously created normal mode profile file or a dynamically created normal mode profile file. The **dont-generate-profile** option suppresses the creation of the normal mode profile file. **Note** The system prompts you to continue. Enter **y** to continue or **n** to terminate the process. The switch is now in normal mode.

Example

This example shows how to return to normal mode from maintenance mode:

```
switch# configure terminal
switch(config)# no system mode maintenance dont-generate-profile
Do you want to continue (y/n)? [n] y

Progressing.............Done.

System mode operation completed successfully
```

```
switch(config)#
```

Configuring the Maintenance Mode Profile File

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **configure profile maintenance-mode type admin**	Enters a configuration session for the maintenance mode profile file. **Note** Depending on which protocols you have configured, you must now enter the appropriate commands to bring down the protocols.
Step 3	switch# **end**	Closes the maintenance mode profile file.

Example

This example shows how to create a maintenance mode profile file:

```
switch# configure terminal
switch(config)# configure profile maintenance-mode type admin
switch(config-profile)# router ospf 100
switch(config-profile-router)# shutdown
switch(config-profile-router)# exit
switch(config-profile)# router eigrp 101
switch(config-profile-router)# shutdown
switch(config-profile-router)# exit
switch(config-profile)# router isis 102
switch(config-profile-router)# shutdown
switch(config-profile-router)# set-overload-bit always
switch(config-profile-router)# exit
switch(config-profile)# router bgp 103
switch(config-profile-router)# shutdown
switch(config-profile-router)# exit
switch(config-profile)# vpc domain 20
switch(config-profile-router)# shutdown
switch(config-profile-router)# exit
switch(config-profile)# system interface shutdown
switch(config-profile)# end
Exit configure profile mode.
switch#
```

This example shows how to create a maintenance mode custom profile file:

```
switch# configure terminal
switch(config)# configure profile maintenance-mode type admin
switch(config-profile)# router bgp 65501
```

```
switch(config-profile-router)# shutdown
switch(config-profile-router)# exit
switch(config-profile)# address-family ipv6 unicast
switch(config-profile)# shutdown
switch(config-profile)# router eigrp 600
switch(config-profile-router)# shutdown
switch(config-profile-router)# exit
switch(config-profile)# address-family ipv6 unicast
switch(config-profile-router)# shutdown
switch(config-profile-router)# exit
switch(config-profile)# router ospf 100
switch(config-profile-router)# shutdown
switch(config-profile-router)# exit
switch(config-profile)# router ospfv3 ospf_ipv6
switch(config-profile-router)# shutdown
switch(config-profile-router)# exit
switch(config-profile)# router isis isp
switch(config-profile-router)# set-overload-bit always
switch(config-profile-router)# exit
switch(config-profile)# vpc domain 2
switch(config-profile-router)# shutdown
switch(config-profile-router)# exit
switch(config-profile)# system interface shutdown
switch(config-profile)# end
Exit configure profile mode.
switch#
```

This example shows how to create a maintenance mode profile for IPv6 protocols:

```
switch# configure terminal
switch(config)# configure profile maintenance-mode type admin
switch(config-profile)# router ospfv3 ospf_ipv6
switch(config-profile-router)# shutdown
switch(config-profile-router)# exit
switch(config-profile)# router eigrp 660
switch(config-profile-router)# address-family ipv6 unicast
switch(config-profile-router-af)# shutdown
switch(config-profile-router-af)# exit
switch(config-profile-router)# router isis isp
switch(config-profile-router)# set-overload-bit always
switch(config-profile-router)# exit
switch(config-profile)# router bgp 655551
switch(config-profile)# address-family ipv6 unicast
switch(config-profile-router)# shutdown
switch(config-profile-router)# exit
switch(config-profile)#
```

Verifying GIR

Use one of the following commands to verify the configuration:

Command	Purpose
show system mode	Displays current system mode.
show interface brief	Displays abbreviated interface information.

Command	Purpose
show snapshots *before-maintenance-mode description*	Displays snapshots present on the switch.
show config-profile *name*	Displays the details of the config-profile files.

show system mode Command

```
switch# show system mode
System Mode : Maintenance
```

show interface brief Command

```
switch# show interface brief

--------------------------------------------------------------------------------
Ehternet       VLAN     Type Mode   Status  Reason               Speed   Port
Interface                                                                Ch #
--------------------------------------------------------------------------------
Eth1/1         - -      eth  routed  down   sysIntfShut           10G(D) - -
Eth1/2         - -      eth  routed  down   sysIntfShut           10G(D) - -
Eth1/3         - -      eth  routed  down   sysIntfShut           10G(D) - -
Eth1/4         - -      eth  routed  down   sysIntfShut           10G(D) - -
Eth1/5         - -      eth  routed  down   sysIntfShut           10G(D) - -
Eth1/6         - -      eth  routed  down   sysIntfShut           10G(D) - -
Eth1/7         - -      eth  routed  down   SFP not inserted      10G(D) - -
Eth1/8         - -      eth  routed  down   SFP not inserted      10G(D) - -
Eth1/9         - -      eth  routed  down   SFP not inserted      10G(D) - -
Eth1/10        - -      eth  routed  down   SFP not inserted      10G(D) - -
Eth1/12        - -      eth  routed  down   SFP not inserted      10G(D) - -
Eth1/13        - -      eth  routed  down   SFP not inserted      10G(D) - -
Eth1/14        - -      eth  routed  down   SFP not inserted      10G(D) - -
Eth1/15        - -      eth  routed  down   SFP not inserted      10G(D) - -
Eth1/16        - -      eth  routed  down   SFP not inserted      10G(D) - -

--------------------------------------------------------------------------------
Port-channel  VLAN    Type Mode    Status  Reason                   Speed   Protocol
Interface
--------------------------------------------------------------------------------
Po1           1       eth  access  down    No operational members   auto(I)  none
Po100         1       eth  access  down    No operational members   auto(I)  none

--------------------------------------------------------------------------------
Port     VRF          Status IP Address                              Speed    MTU
--------------------------------------------------------------------------------
mgmt0    - -          up     192.0.0.1                               1000     1500
switch#
```

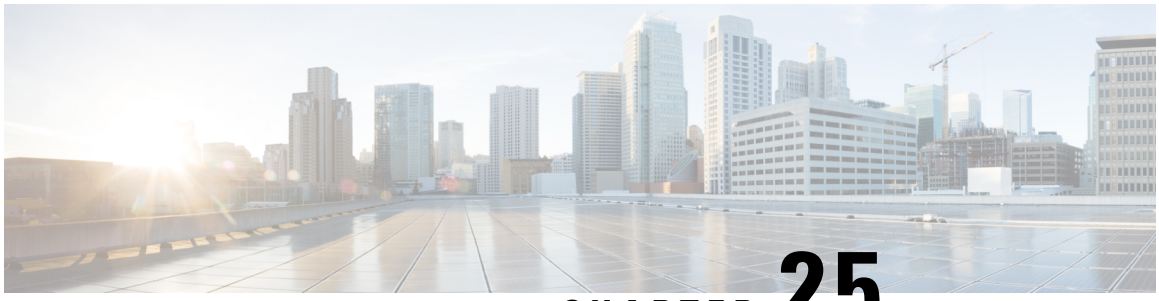
show snapshots Command

```
switch# show snapshots
Snapshot Name
-----------------------------------------------------------------------------------------
snapshot_before_maintenance      Wed Sep 10 20:19:31 2014      system-internal-snapshot
```

```
snapshot_after_maintenance       Wed Sep 10 20:29:54 2014      system-internal-snapshot
snap1                            Wed Sep 10 20:36:15 2014      For testing
```

show config-profile Command

```
switch# show config-profile

config-profile maintenance-mode type admin
  router ospf 100
   shutdown
  router eigrp 101
   shutdown
  router isis 102
   set-overload-bit always
  router bgp 103
   shutdown
  vpc domain 20
   shutdown
  system interface shutdown exclude fex-fabric

config-profile normal-mode type admin
  router ospf 100
   no shutdown
  router eigrp 101
   no shutdown
  router isis 102
   no set-overload-bit always
  router bgp 103
   no shutdown
  vpc domain 20
   no shutdown
  no system interface shutdown
```



CHAPTER 25

Class-based Quality-of-Service MIB

This chapter contains the following sections:

Class-based Quality-of-Service MIB

The Class-based Quality-of-Service MIB (cbQoSMIB) feature provides the Simple Network Management Protocol (SNMP) MIB that enables retrieval of class-map and policy-map configuration and statistics.

Information About Class-based Quality-of-Service MIB

CoPP and QoS policies now support Class-based Quality-of-Service MIB (cbQoSMIB). cbQoSMIB is the SNMP MIB that provides access to Modular QoS CLI (MQC) configuration and statistics.

The following cbQoSMIB tables are supported by QoS policies and CoPP:

- cbQosClassMapCfg
- cbQosMatchStmtCfg
- cbQosPoliceStats
- cbQosPolicyMapCfg
- cbQosPoliceCfg

The following cbQoSMIB tables are supported by QoS policies:

- cbQosInterfacePolicy
- cbQosObjects
- cbQosQueueingCfg
- cbQosServicePolicy
- cbQosSetCfg

Class-based Quality-of-Service MIB Phase 2

Beginning from Cisco NX-OS Release 7.3(0)N1(1), the following cbQoSMIB tables are also supported by QoS policies:

- cbQosClassMapStats
- cbQosMatchStmtStats
- cbQosQueueingStats

More detailed information on cbQoSMIB tables and elements is available at the following url: http://tools.cisco.com/Support/SNMP/do/BrowseOID.do?local=en&translate=Translate&objectInput=1.3.6.1.4.1.9.9.166

Licensing Requirements for Class-based Quality-of-Service MIB

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.

Prerequisite for Class-based Quality-of-Service MIB

- You must enable QoS Statistics under **show policy-map interface** to view statistics under Class-based Quality-of-Service MIB. For more information, refer Enabling QoS Statistics under show policy-map interface.

Guidelines and Limitations for Class-based Quality-of-Service MIB

The guidelines and limitations for viewing statistics are as follows:

- Statistics can be viewed per Access Control Entry (ACE) in an Access Control List (ACL) if there is no policer attached.
- Statistics can be viewed per ACE in an ACL, if there is only one ACE in the ACL and if a policer is attached.
- Statistics cannot be viewed per ACE in an ACL, if there are more than one ACEs in an ACL and a policer is attached.
- The limitations above apply to QoS-based matches as well, such as **match dscp** *dscp-list* , **match precedence** *precedence-list* and so on.
- Statistics cannot be viewed with **match-all** rules.
- Statistics can be viewed only with **match-any**.
- For instances when the statistics do not get enabled without a policer, follow these steps:
  - Create a class. You can add as many rules as required.
  - Create a policy-map and attach the above class to it.
  - Add a dummy policer to it (if you do not require a real policer).

- Apply the policy to the interface.
- Remove the dummy policer to display the statistics.

- Statistics are shown per policy and not at an interface level.
- Use the **show ip access-list** command to display statistics for matches based on access group. These statistics cannot be viewed with the **show policy-map interface** command.

Configuring a QoS Policy

The following configuration is a generic example to configure a QoS policy.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **snmp-server community** *com-name* **rw**	Creates Simple Network Management Protocol (SNMP) communities for SNMPv1 or SNMPv2c.
Step 3	switch(config) # **snmp-server community** *com-name* **rw**	Creates Simple Network Management Protocol (SNMP) communities for SNMPv1 or SNMPv2c.
Step 4	switch(config) # **class-map type qos match-any** *class-map-name*	Specifies the component type qos for the class map and enters the class-map type qos configuration mode.
Step 5	switch(config-cmap-qos) # **description** *text*	Adds a description for the class-map.
Step 6	switch(config-cmap-qos) # **match cos** *cos-list*	Defines the class of traffic using the class of service (CoS) value in a type qos class map.
Step 7	switch(config-cmap-qos) # **match dscp** *dscp-list*	(Optional) Specifies differentiated services code point (DSCP) values in the DiffServ field of the IP Header (either IPv4 or IPv6) as a match criterion.
Step 8	switch(config-cmap-qos) # **exit**	Exits the class-map type qos configuration mode.
Step 9	switch(config) # **policy-map type qos** *qos-policy-map-name*	Specifies the type qos policy map and enters the policy-map qos configuration mode.
Step 10	switch(config-pmap-qos) # **description** *text*	Configures the policy-map description.
Step 11	switch(config-pmap-qos) # **class** *class-map-name*	Configures the service policy for a class-map.
Step 12	switch(config-pmap-c-qos) # **set qos-group** *qos-group-value*	Assigns the QoS group identifier for a class of traffic in a type qos policy map.

	Command or Action	Purpose
Step 13	switch(config-pmap-c-qos) # **exit**	Exits the policy-map type qos class configuration mode.
Step 14	switch(config-pmap-qos) # **exit**	Exits the policy-map qos configuration mode.
Step 15	switch(config) # **interface** *type number*	Enters the interface configuration mode.
Step 16	switch(config-if) # **service-policy type qos input** *policy-map-name*	Applies the service policy map to packets coming into the mentioned interface.
Step 17	switch(config-if) # **exit**	Exits the interface configuration mode.
Step 18	(Optional) switch(config) # **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to configure a QoS policy on a switch:

Note This is a generic example to configure a QoS policy.

```
switch# configure terminal
switch(config)# snmp-server community public rw
switch(config)# snmp-server community private rw
switch(config)# class-map type qos match-any cmap1
switch(config-cmap-qos) # description class map 1
switch(config-cmap-qos) # match cos 4
switch(config-cmap-qos) # match dscp 48
switch(config-cmap-qos) # exit
switch(config) # policy-map type qos pmap1
switch(config-pmap-qos) # description policy map 1
switch(config-pmap-qos) # class cmap1
switch(config-pmap-c-qos) # set qos-group 4
switch(config-pmap-c-qos) # exit
switch(config-pmap-qos) # exit
switch(config) # interface ethernet 1/3
switch(config-if) # service-policy type qos input pmap1
```

Displaying Class-based Quality-of-Service MIB Configuration and Statistics

Procedure

	Command or Action	Purpose
Step 1	$ **snmpwalk -v2c -c** *community-name ip-address oid*	Displays class-map and policy-map configuration and statistics.

	Command or Action	Purpose
		Note Use the **snmpwalk** command on an SNMP-enabled server.

Example

The following examples show how to display class map and policy map configuration and statistics:

Use the **show interface snmp-ifindex** command to display the mapping of ifindices to interfaces:

```
switch(config)# show interface snmp-ifindex

--------------------------------------------------------------------------------
Port             IFMIB Ifindex (hex)
--------------------------------------------------------------------------------
Eth1/1          436207616  (0x1a000000)
Eth1/2          436211712  (0x1a001000)
Eth1/3          436215808  (0x1a002000)
Eth1/4          436219904  (0x1a003000)
Eth1/5          436224000  (0x1a004000)
```

Use the **show policy-map interface** *type number* command to display statistics and the configured policy maps on a specified interface:

```
switch# show policy-map interface ethernet 1/3

Global statistics status :   enabled

 NOTE: Type qos policy-map configured on VLAN will take precedence
       over system-qos policy-map for traffic on the VLAN


Ethernet1/3

  Service-policy (qos) input:   pmap1
    policy statistics status:   enabled

    Class-map (qos):   cmap1 (match-any)
      14 packets
      Match: cos 4
        10 Match packets
      Match: dscp 48
        4 Match packets
      set qos-group 4

    Class-map (qos):   class-default (match-any)
      0 packets
```

Use the **snmpwalk** command on the Service Policy Table:

```
$ snmpwalk -v2c -c public A.B.C.D cbQosServicePolicy


Service Policy Table (QoS only table) – corresponding to the service policy applied on
eth1/3

CISCO-CLASS-BASED-QOS-MIB::cbQosIfType.285212681 = INTEGER: mainInterface(1)
CISCO-CLASS-BASED-QOS-MIB::cbQosPolicyDirection.285212681 = INTEGER: input(1)
```

```
CISCO-CLASS-BASED-QOS-MIB::cbQosIfIndex.285212681 = INTEGER: 436215808 //436215808 is the
IFMIB Interface Index value
CISCO-CLASS-BASED-QOS-MIB::cbQosVlanIndex.285212681 = Gauge32: 1

//The interface is Eth1/3.
```

Use the **snmpwalk** command on the Objects Table:

```
$ snmpwalk -v2c -c public A.B.C.D cbQosObjects


Objects Table (QoS only table)  corresponding to the policy-map, class-map, match & set
Statements

CISCO-CLASS-BASED-QOS-MIB::cbQosConfigIndex.285212681.285212681 = Gauge32: 285212836
//285212836 is the Policy Map Config Index
CISCO-CLASS-BASED-QOS-MIB::cbQosConfigIndex.285212681.285212682 = Gauge32: 285212833
//285212833 is a Class Map Config Index
CISCO-CLASS-BASED-QOS-MIB::cbQosConfigIndex.285212681.285212683 = Gauge32: 285212834
//285212834 is a Match Statement Config Index
CISCO-CLASS-BASED-QOS-MIB::cbQosConfigIndex.285212681.285212684 = Gauge32: 285212835
//285212835 is a Match Statement Config Index


CISCO-CLASS-BASED-QOS-MIB::cbQosObjectsType.285212681.285212681 = INTEGER: policymap(1)
CISCO-CLASS-BASED-QOS-MIB::cbQosObjectsType.285212681.285212682 = INTEGER: classmap(2)
CISCO-CLASS-BASED-QOS-MIB::cbQosObjectsType.285212681.285212683 = INTEGER: matchStatement(3)
CISCO-CLASS-BASED-QOS-MIB::cbQosObjectsType.285212681.285212684 = INTEGER: matchStatement(3)

CISCO-CLASS-BASED-QOS-MIB::cbQosParentObjectsIndex.285212681.285212681 = Gauge32: 0
CISCO-CLASS-BASED-QOS-MIB::cbQosParentObjectsIndex.285212681.285212682 = Gauge32: 285212681
CISCO-CLASS-BASED-QOS-MIB::cbQosParentObjectsIndex.285212681.285212683 = Gauge32: 285212682
CISCO-CLASS-BASED-QOS-MIB::cbQosParentObjectsIndex.285212681.285212684 = Gauge32: 285212682
```

Use the **snmpwalk** command on the Policy Map Table:

```
$ snmpwalk -v2c -c public A.B.C.D cbQosPolicyMapCfg | grep 285212836
//285212836 is the Policy Map Config Index obtained from the Objects Table


Policy Map Table corresponding to the policy-map configured above

CISCO-CLASS-BASED-QOS-MIB::cbQosPolicyMapName.285212836 = STRING: pmap1 //pmap1 is the
policy map name
CISCO-CLASS-BASED-QOS-MIB::cbQosPolicyMapDesc.285212836 = STRING: policy map 1 //Policy map
 description
```

Use the **snmpwalk** command on the Class Map Table:

```
$ snmpwalk -v2c -c public A.B.C.D cbQosClassMapCfg | grep 285212833
//285212833 is the Class Map Config Index obtained from the Objects Table

Class Map Table corresponding to the class-map configured above

CISCO-CLASS-BASED-QOS-MIB::cbQosCMName.285212833 = STRING: cmap1 //class-map on which the
service-policy is configured
CISCO-CLASS-BASED-QOS-MIB::cbQosCMDesc.285212833 = STRING: class map 1  //class-map
description
```

Use the **snmpwalk** command on the Match Statement Table:

```
$ snmpwalk -v2c -c public A.B.C.D cbQosMatchStmtCfg | grep 285212834

Match Stmt Table corresponding to the match statement configured above

CISCO-CLASS-BASED-QOS-MIB::cbQosMatchStmtName.285212834 = STRING: match cos 4
```

Use the **snmpwalk** command on the Queuing Config Table:

```
$ snmpwalk -v2c -c public A.B.C.D cbQosQueueingCfg
Queueing Config Table(QoS only table, taken from default QoS policies)

CISCO-CLASS-BASED-QOS-MIB::cbQosQueueingCfgBandwidth.301990031 = INTEGER: 100
CISCO-CLASS-BASED-QOS-MIB::cbQosQueueingCfgBandwidthUnits.301990031 = INTEGER: percentage(2)
CISCO-CLASS-BASED-QOS-MIB::cbQosQueueingCfgPriorityEnabled.301990031 = INTEGER: false(2)
CISCO-CLASS-BASED-QOS-MIB::cbQosQueueingCfgQLimitUnits.301990031 = INTEGER: 0
CISCO-CLASS-BASED-QOS-MIB::cbQosQueueingCfgAggregateQLimit.301990031 = Gauge32: 0
```

Use the **snmpwalk** command on the Set Action Table:

```
$ snmpwalk -v2c -c public A.B.C.D cbQosSetCfg
Set Action Table (QoS only table) corresponding to the set statement configured above

CISCO-CLASS-BASED-QOS-MIB::cbQosSetCfgIpDSCPValue.285212829 = INTEGER: 0
CISCO-CLASS-BASED-QOS-MIB::cbQosSetCfgIpPrecedenceValue.285212829 = INTEGER: 0
CISCO-CLASS-BASED-QOS-MIB::cbQosSetCfgQosGroupValue.285212838 = INTEGER: 4
CISCO-CLASS-BASED-QOS-MIB::cbQosSetCfgL2CosValue.285212829 = INTEGER: 0
```

Use the **snmpwalk** command on the Policing Config Table:

```
$ snmpwalk -v2c -c public A.B.C.D cbQosPoliceCfg
Policing Config Table(no QoS config, displays only CoPP statistics)

CISCO-CLASS-BASED-QOS-MIB::cbQosPoliceCfgBurstSize.721420367 = Gauge32: 65535 Octets
CISCO-CLASS-BASED-QOS-MIB::cbQosPoliceCfgConformAction.721420367 = INTEGER: transmit(1)
CISCO-CLASS-BASED-QOS-MIB::cbQosPoliceCfgViolateAction.721420367 = INTEGER: drop(5)
CISCO-CLASS-BASED-QOS-MIB::cbQosPoliceCfgRate64.721420367 = Counter64: 1048576 bits/second

CISCO-CLASS-BASED-QOS-MIB::cbQosPoliceCfgRateType.721420367 = INTEGER: bps(1)
CISCO-CLASS-BASED-QOS-MIB::cbQosPoliceCfgConditional.721420367 = INTEGER: false(2)
```

Use the **snmpwalk** command on the Policing Stats Table:

```
$ snmpwalk -v2c -c public A.B.C.D cbQosPoliceStats
Policing Stats Table(no QoS config, displays only CoPP statistics)

CISCO-CLASS-BASED-QOS-MIB::cbQosPoliceConformedByte64.721420366.721420376 = Counter64: 80121
 Octets
CISCO-CLASS-BASED-QOS-MIB::cbQosPoliceViolatedByte64.721420366.721420367 = Counter64: 0
Octets
```

Note All CoPP configurations are available by default.

The sample snmpwalk outputs below display the cbQosMatchStmtStats and cbQosClassMapStats tables that are supported by the QoS policies starting from Cisco NX-OS Release 7.3(0)N1(1):

```
$ snmpwalk -v2c -c public A.B.C.D cbQosMatchStmtStats
CISCO-CLASS-BASED-QOS-MIB::cbQosMatchPrePolicyPkt64.285212681.285212683 = Counter64: 10
//The config indices match the objects displayed in the Objects Table above
CISCO-CLASS-BASED-QOS-MIB::cbQosMatchPrePolicyPkt64.285212681.285212684 = Counter64: 4
CISCO-CLASS-BASED-QOS-MIB::cbQosMatchPrePolicyPkt64.285212681.285212687 = Counter64: 0
CISCO-CLASS-BASED-QOS-MIB::cbQosMatchPrePolicyByte64.285212681.285212683 = Counter64: 0
CISCO-CLASS-BASED-QOS-MIB::cbQosMatchPrePolicyByte64.285212681.285212684 = Counter64: 0
CISCO-CLASS-BASED-QOS-MIB::cbQosMatchPrePolicyByte64.285212681.285212687 = Counter64: 0
CISCO-CLASS-BASED-QOS-MIB::cbQosMatchPrePolicyBitRate.285212681.285212683 = Gauge32: 0 bits
 per second
CISCO-CLASS-BASED-QOS-MIB::cbQosMatchPrePolicyBitRate.285212681.285212684 = Gauge32: 0 bits
 per second
CISCO-CLASS-BASED-QOS-MIB::cbQosMatchPrePolicyBitRate.285212681.285212687 = Gauge32: 0 bits
 per second


$snmpwalk -v2c -c public A.B.C.D cbQosClassMapStats
CISCO-CLASS-BASED-QOS-MIB::cbQosCMPrePolicyPkt64.285212681.285212682 = Counter64: 14 //The
 config indices match the objects displayed in the Objects Table above
CISCO-CLASS-BASED-QOS-MIB::cbQosCMPrePolicyPkt64.285212681.285212686 = Counter64: 0
CISCO-CLASS-BASED-QOS-MIB::cbQosCMPrePolicyByte64.285212681.285212682 = Counter64: 0
CISCO-CLASS-BASED-QOS-MIB::cbQosCMPrePolicyByte64.285212681.285212686 = Counter64: 0
CISCO-CLASS-BASED-QOS-MIB::cbQosCMPrePolicyBitRate.285212681.285212682 = Gauge32: 0 bits
per second
CISCO-CLASS-BASED-QOS-MIB::cbQosCMPrePolicyBitRate.285212681.285212686 = Gauge32: 0 bits
per second
CISCO-CLASS-BASED-QOS-MIB::cbQosCMPostPolicyByte64.285212681.285212682 = Counter64: 0
CISCO-CLASS-BASED-QOS-MIB::cbQosCMPostPolicyByte64.285212681.285212686 = Counter64: 0
CISCO-CLASS-BASED-QOS-MIB::cbQosCMDropPkt64.285212681.285212682 = Counter64: 0
CISCO-CLASS-BASED-QOS-MIB::cbQosCMDropPkt64.285212681.285212686 = Counter64: 0
CISCO-CLASS-BASED-QOS-MIB::cbQosCMDropByte64.285212681.285212682 = Counter64: 0
CISCO-CLASS-BASED-QOS-MIB::cbQosCMDropByte64.285212681.285212686 = Counter64: 0
CISCO-CLASS-BASED-QOS-MIB::cbQosCMDropBitRate.285212681.285212682 = Gauge32: 0 bits per
second
CISCO-CLASS-BASED-QOS-MIB::cbQosCMDropBitRate.285212681.285212686 = Gauge32: 0 bits per
second
```

Use the **show policy-map interface control-plane** command to display control plane statistics:

```
switch# show policy-map interface control-plane

Control Plane

  service-policy  input: copp-system-policy-default

    class-map copp-system-class-igmp (match-any)
      match protocol igmp
      police cir 1024 kbps , bc 65535 bytes
        conformed 0 bytes; action: transmit
        violated 0 bytes;
    class-map copp-system-class-pim-hello (match-any)
      match protocol pim
      police cir 1024 kbps , bc 4800000 bytes
        conformed 0 bytes; action: transmit
        violated 0 bytes;
    class-map copp-system-class-bridging (match-any)
      match protocol bridging
      police cir 20000 kbps , bc 4800000 bytes
        conformed 0 bytes; action: transmit
        violated 0 bytes;
    class-map copp-system-class-arp (match-any)
      match protocol arp
      match protocol nd
```

```
      police cir 1024 kbps , bc 3600000 bytes
        conformed 0 bytes; action: transmit
        violated 0 bytes;
    class-map copp-system-class-dhcp (match-any)
      match protocol dhcp
      police cir 1024 kbps , bc 4800000 bytes
        conformed 0 bytes; action: transmit
        violated 0 bytes;
    class-map copp-system-class-wccp (match-any)
      match protocol wccp
      police cir 1060 kbps , bc 4800000 bytes
        conformed 0 bytes; action: transmit
        violated 0 bytes;
.
.
.
```

Additional References for Class-based Quality-of-Service MIB

This section provides additional information related to Class-based Quality-of-Service MIB.

Related Documents

Related Topic	Document Title
Licensing	Cisco NX-OS Licensing Guide
Command reference	Cisco Nexus 5500 Series NX-OS QoS Command Reference Cisco Nexus 5500 Series NX-OS System Management Command Reference Cisco Nexus 6000 Series NX-OS QoS Command Reference Cisco Nexus 6000 Series NX-OS System Management Command Reference

Feature History for Class-based Quality-of-Service MIB

Table 34: Feature History for Class-based Quality-of-Service MIB

Feature Name	Releases	Feature Information
Class-based Quality-of-Service MIB Phase 2	7.3(0)N1(1)	The following cbQoSMIB tables are supported by QoS policies: cbQosClassMapStats, cbQosMatchStmtStats and cbQosQueueingStats
Class-based Quality-of-Service MIB	7.1(1) N1(1)	This feature was introduced.

INDEX

E

F

G

H

I

L

M

N

O

P

R

S

T

U

V