



## **System Management Configuration Guide for Cisco 8000 Series Routers, IOS XR Release 24.1.x , 24.2.x , 24.3.x**

**First Published:** 2024-03-14

### **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

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at [www.cisco.com/go/offices](http://www.cisco.com/go/offices).

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: <https://www.cisco.com/c/en/us/about/legal/trademarks.html>. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2024 Cisco Systems, Inc. All rights reserved.



## CONTENTS

---

<b>PREFACE</b>	<b>Preface</b> ix
	Changes to this Document ix

---

<b>CHAPTER 1</b>	<b>New and Changed System Management Features</b> 1
	System Management Features Added or Modified in IOS XR Release 24.X.X 1

---

<b>CHAPTER 2</b>	<b>YANG Data Models for System Management Features</b> 3
	Using YANG Data Models 3

---

<b>CHAPTER 3</b>	<b>Configuring Physical and Virtual Terminals</b> 5
	Prerequisites for Implementing Physical and Virtual Terminals 5
	Information About Implementing Physical and Virtual Terminals 5
	Line Templates 5
	Line Template Configuration Mode 6
	Line Template Guidelines 6
	Terminal Identification 7
	vty Pools 7
	How to Implement Physical and Virtual Terminals on Cisco IOS XR Software 8
	Modifying Templates 8
	Creating and Modifying vty Pools 9
	Monitoring Terminals and Terminal Sessions 11
	Configuration Examples for Implementing Physical and Virtual Terminals 12

---

<b>CHAPTER 4</b>	<b>Configuring Simple Network Management Protocol</b> 15
	Prerequisites for Implementing SNMP 15
	Restrictions for SNMP use on Cisco IOS XR Software 15

Information about Implementing SNMP	16
SNMP Functional Overview	16
SNMP Manager	16
SNMP Agent	16
MIB	16
SNMP Versions	17
Comparison of SNMPv1, v2c, and v3	18
Security Models and Levels for SNMPv1, v2, v3	19
SNMPv3 Benefits	20
SNMPv3 Costs	20
User-Based Security Model	20
View-Based Access Control Model	21
IP Precedence and DSCP Support for SNMP	21
Custom MIB Support Using SNMP Operation Script	22
Restrictions for Custom MIB	23
Create Custom MIB Using SNMP Script	23
Session MIB support on subscriber sessions	24
SNMP Notifications	25
Session Types	25
How to Implement SNMP on Cisco IOS XR Software	26
Configuring SNMPv3	26
Configure to Drop Error PDUs	28
Configuring SNMPv3: Examples	28
Configuring SNMP Trap Notifications	32
Configure to Drop Error PDUs	34
Configuring Trap Notifications: Example	34
Setting the Contact, Location, and Serial Number of the SNMP Agent	35
Defining the Maximum SNMP Agent Packet Size	36
Changing Notification Operation Values	37
Setting IP Precedence and DSCP Values	38
Setting IPv6 Precedence and DSCP Values	39
Setting an IP Precedence Value for SNMP Traffic: Example	39
Setting an IP DSCP Value for SNMP Traffic: Example	40
Displaying SNMP Context Mapping	40

Monitoring Packet Loss	41
Configuring MIB Data to be Persistent	41
Configuring LinkUp and LinkDown Traps for a Subset of Interfaces	42
Polling BRIDGE-MIB	44

---

**CHAPTER 5**

<b>Configuring Periodic MIB Data Collection and Transfer</b>	<b>45</b>
Prerequisites for Periodic MIB Data Collection and Transfer	45
Information About Periodic MIB Data Collection and Transfer	45
SNMP Objects and Instances	45
Bulk Statistics Object Lists	46
Bulk Statistics Schemas	46
Bulk Statistics Transfer Options	46
Benefits of Periodic MIB Data Collection and Transfer	46
How to Configure Periodic MIB Data Collection and Transfer	47
Configuring a Bulk Statistics Object List	47
Configuring a Bulk Statistics Schema	48
Configuring Bulk Statistics Transfer Options	49
Periodic MIB Data Collection and Transfer: Example	52

---

**CHAPTER 6**

<b>Configuring Cisco Discovery Protocol</b>	<b>55</b>
Prerequisites for Implementing CDP	55
Information About Implementing CDP	55
How to Implement CDP on Cisco IOS XR Software	57
Enabling CDP	57
Modifying CDP Default Settings	57
Monitoring CDP	58
Examples	59

---

**CHAPTER 7**

<b>Configuring Call Home</b>	<b>61</b>
About Call Home	61
Benefits of Using Call Home	62
Prerequisites for Call Home	62
How to Configure Call Home	63
Configuring Contact Information	63

Destination Profiles	65
Configuring and Activating Destination Profiles	65
Call Home Alert Groups	67
Call Home Message Levels	68
Associating an Alert Group with a Destination Profile	69
Configuring Email	71
Configuring a HTTPS Proxy Server	72
Sending Call-home Data through an Email	73
Sending Call-home Data through HTTPS	75
Configuring Call Home to use VRF	76
Configuring Call Home Data Privacy	77
Sending Smart License Data	78

**CHAPTER 8****Managing Router Hardware 81**

RP Redundancy and Switchover	81
Establishing RP Redundancy	81
Determining the Active RP in a Redundant Pair	83
Role of the Standby RP	84
Summary of Redundancy Commands	84
Automatic Switchover	84
RP Redundancy During RP Reload	85
Manual Switchover	85
Communicating with a Standby RP	86
NPU Power Optimization	86
Limitations	87
Configuring NPU Power Mode	88
Dynamic Power Management	91
Disabling Dynamic Power Management	98
On-demand transfer of Redundant Power Modules to Power Reservation Pool	98
Power Redundancy Protection	103
Guidelines and Restrictions for Power Redundancy Protection	104
Configure Power Redundancy Protection	104
Ability to Set Maximum Power Limit for the Router	107
Configuring the Compatibility Mode for Various ASIC Types	108

Storage Media Sanitization	115
Excluding Sensitive Information in Show Running Configurations Output	120
Fabric Link Management for Uncorrectable Errors	123
Monitor FEC Fabric Links	124
Fault Recovery Handling	126
How to Configure the Fault Recovery Attempts	126

---

<b>CHAPTER 9</b>	<b>The Network Configuration Protocol</b>	<b>129</b>
	Netconf Sessions and Operations	129
	The Yang data model	130
	Netconf and Yang	131
	Supported Yang Models	132
	Denial of Services Defense for Netconf-Yang	132
	Enabling NETCONF over SSH	133
	Examples: Netconf over SSH	134

---

<b>CHAPTER 10</b>	<b>Configuring PTP</b>	<b>137</b>
	PTP Overview	138
	Restrictions for PTP	140
	PTP Support Information	141
	Timing Profile and Class Support Matrix	141
	Configuring PTP Delay Asymmetry	142
	Performance Monitoring for PTP Networks	145
	Configure PTP Performance Monitoring	147
	ITU-T Telecom Profiles for PTP	150
	G.8265.1	150
	G.8263 Standard	153
	G.8275.1	154
	ITU-T Telecom Profile Examples:	160
	G.8265.1 Profile Configuration Examples	160
	G.8275.1	161

---

<b>CHAPTER 11</b>	<b>Configuration and File System Management</b>	<b>167</b>
	Secure file transfer from the Router	167

Prerequisites for secure file transfer	168
Secure file transfer using SFTP	168
Secure file transfer using SCP	169
Auto-Save Configuration	170
Configure Auto-Save	171
Auto-Save and Copy Router Configuration Using Public Key Authentication	172
Configuration Example for Auto-Save Using Public Key Authentication	173
Increasing Commit Limit	174
Guidelines and Restrictions for Increasing the Commit Limit	175
Increasing the Rebase Limits	176
Perform ASCII Backup and Rest ASCII Backup Timer	176
Concurrent Configuration Rebase during Commit	177
View VRF-specific Configuration	178
View Running Configuration for a specific VRF	178





# Preface

---

This guide describes the System Management configuration details for Cisco IOS XR software. This chapter contains details on the changes made to this document.

- [Changes to this Document, on page ix](#)

## Changes to this Document

*Table 1: Changes to this Document*

Date	Summary
February 2024	Initial release of this document





# CHAPTER 1

## New and Changed System Management Features

This chapter lists all the features that have been added or modified in this guide. The table also contains references to these feature documentation sections.

- [System Management Features Added or Modified in IOS XR Release 24.X.X, on page 1](#)

### System Management Features Added or Modified in IOS XR Release 24.X.X

Feature	Description	Changed in Release	Where Documented
Concurrent Configuration Rebase during Commit	This feature is introduced.	Release 24.3.1	<a href="#">Concurrent Configuration Rebase during Commit, on page 177</a>
Performance Monitoring for PTP Networks	This feature is introduced.	Release 24.3.1	<a href="#">Performance Monitoring for PTP Networks, on page 145</a>
Power Redundancy Protection	This feature is introduced.	Release 24.1.1	<a href="#">Power Redundancy Protection , on page 103</a>
View VRF-specific Configuration	This feature is introduced.	Release 24.2.11	<a href="#">View VRF-specific Configuration, on page 178</a>
Fault Recovery Handling	This feature is introduced.	Release 24.2.11	<a href="#">Fault Recovery Handling, on page 126</a>
Increasing Commit Limit	This feature is introduced.	Release 24.2.11	<a href="#">Increasing Commit Limit, on page 174</a>
Fabric Link Management for Uncorrectable Errors	This feature is introduced.	Release 24.2.11	<a href="#">Fabric Link Management for Uncorrectable Errors, on page 123</a>





## CHAPTER 2

# YANG Data Models for System Management Features

---

This chapter provides information about the YANG data models for System Management features.

- [Using YANG Data Models, on page 3](#)

## Using YANG Data Models

Cisco IOS XR supports a programmatic way of configuring and collecting operational data of a network device using YANG data models. Although configurations using CLIs are easier and human-readable, automating the configuration using model-driven programmability results in scalability.

The data models are available in the release image, and are also published in the [Github](#) repository. Navigate to the release folder of interest to view the list of supported data models and their definitions. Each data model defines a complete and cohesive model, or augments an existing data model with additional XPath. To view a comprehensive list of the data models supported in a release, navigate to the **Available-Content.md** file in the repository.

You can also view the data model definitions using the [YANG Data Models Navigator](#) tool. This GUI-based and easy-to-use tool helps you explore the nuances of the data model and view the dependencies between various containers in the model. You can view the list of models supported across Cisco IOS XR releases and platforms, locate a specific model, view the containers and their respective lists, leaves, and leaf lists presented visually in a tree structure. This visual tree form helps you get insights into nodes that can help you automate your network.

To get started with using the data models, see the *Programmability Configuration Guide*.





## CHAPTER 3

# Configuring Physical and Virtual Terminals

Line templates define standard attribute settings for incoming and outgoing transport over physical and virtual terminal lines (vty). Vty pools are used to apply template settings to ranges of vtys.

This module describes the tasks you need to implement physical and virtual terminals on your Cisco IOS XR network.

- [Prerequisites for Implementing Physical and Virtual Terminals, on page 5](#)
- [Information About Implementing Physical and Virtual Terminals, on page 5](#)
- [How to Implement Physical and Virtual Terminals on Cisco IOS XR Software, on page 8](#)
- [Configuration Examples for Implementing Physical and Virtual Terminals, on page 12](#)

## Prerequisites for Implementing Physical and Virtual Terminals

You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

## Information About Implementing Physical and Virtual Terminals

To implement physical and virtual terminals, you need to understand the concepts in this section.



**Tip** You can programmatically manage the physical and virtual terminals using `openconfig-system-terminal.yang` OpenConfig data model. To get started with using data models, see the *Programmability Configuration Guide for Cisco 8000 Series Routers*.

## Line Templates

The following line templates are available in the Cisco IOS XR software.

- Default line template—The default line template that applies to a physical and virtual terminal lines.
- Console line template—The line template that applies to the console line.

- User-defined line templates—User-defined line templates that can be applied to a range of virtual terminal lines.

## Line Template Configuration Mode

Changes to line template attributes are made in line template configuration mode. To enter line template configuration mode, issue the **line** command from XR Config mode, specifying the template to be modified. These line templates can be configured with the **line** command:

- console—console template
- default—default template
- template—user-defined template

After you specify a template with the **line** command, the router enters line template configuration mode where you can set the terminal attributes for the specified line. This example shows how to specify the attributes for the console:

```
RP/0/RP0/CPU0:router (config) # line console
RP/0/RP0/CPU0:router (config-line) #
```

From line template configuration mode, use the online help feature ( ? ) to view all available options. Some useful options include:

- absolute-timeout—Specifies a timeout value for line disconnection.
- escape-character—Changes the line escape character.
- exec-timeout—Specifies the EXEC timeout.
- length—Sets the number of lines displayed on the screen.
- session-limit—Specifies the allowable number of outgoing connections.
- session-timeout—Specifies an interval for closing the connection if there is no input traffic.
- timestamp—Displays the timestamp before each command.
- width—Specifies the width of the display terminal.




---

**Note** The *default* session-limit for line template is applicable to Telnet sessions only. It is not applicable for SSH sessions.

---

## Line Template Guidelines

The following guidelines apply to modifying the console template and to configuring a user-defined template:

- Modify the templates for the physical terminal lines on the router (the console port) from line template configuration mode. Use the **line console** command from XR Config mode to enter line template configuration mode for the console template.



- Modify the template for virtual lines by configuring a user-defined template with the **line** *template-name* command, configuring the terminal attributes for the user-defined template from line template configuration, and applying the template to a range of virtual terminal lines using the **vty pool** command.

Attributes not defined in the console template, or any virtual template, are taken from the default template.

The default settings for the default template are described for all commands in line template configuration mode in the *Terminal Services Commands on* module in *System Management Command Reference for Cisco 8000 Series Routers*.



---

**Note** Before creating or modifying the vty pools, enable the telnet server using the **telnet server** command in XR Config mode. See *IP Addresses and Services Configuration Guide for Cisco 8000 Series Routers* and *IP Addresses and Services Command Reference for Cisco 8000 Series Routers* for more information.

---

## Terminal Identification

The physical terminal lines for the console port is identified by its location, expressed in the format of *rack/slot/module*, on the active or standby route processor (RP) where the respective console port resides. For virtual terminals, physical location is not applicable; the Cisco IOS XR software assigns a vty identifier to vtys according to the order in which the vty connection has been established.

## vty Pools

Each virtual line is a member of a pool of connections using a common line template configuration. Multiple vty pools may exist, each containing a defined number of vtys as configured in the vty pool. The Cisco IOS XR software supports the following vty pools by default:

- Default vty pool—The default vty pool consists of five vtys (vtys 0 through 4) that each reference the default line template.
- Default fault manager pool—The default fault manager pool consists of six vtys (vtys 100 through 105) that each reference the default line template.

In addition to the default vty pool and default fault manager pool, you can also configure a user-defined vty pool that can reference the default template or a user-defined template.

When configuring vty pools, follow these guidelines:

- The vty range for the default vty pool must start at vty 0 and must contain a minimum of five vtys.
- The vty range from 0 through 99 can reference the default vty pool.
- The vty range from 5 through 99 can reference a user-defined vty pool.
- The vty range from 100 is reserved for the fault manager vty pool.
- The vty range for fault manager vty pools must start at vty 100 and must contain a minimum of six vtys.
- A vty can be a member of only one vty pool. A vty pool configuration will fail if the vty pool includes a vty that is already in another pool.
- If you attempt to remove an active vty from the active vty pool when configuring a vty pool, the configuration for that vty pool will fail.

# How to Implement Physical and Virtual Terminals on Cisco IOS XR Software

## Modifying Templates

This task explains how to modify the terminal attributes for the console and default line templates. The terminal attributes that you set will modify the template settings for the specified template.

### SUMMARY STEPS

1. **configure**
2. **line {console | default}**
3. Configure the terminal attribute settings for the specified template using the commands in line template configuration mode.
4. Use one of the following commands:
  - **end**
  - **commit**

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>configure</b> <b>Example:</b> RP/0/RP0/CPU0:router# configure	Enters mode.
<b>Step 2</b>	<b>line {console   default}</b> <b>Example:</b> RP/0/RP0/CPU0:router(config)# line console or RP/0/RP0/CPU0:router(config)# line default	Enters line template configuration mode for the specified line template. <ul style="list-style-type: none"> <li>• <b>console</b> —Enters line template configuration mode for the console template.</li> <li>• <b>default</b> —Enters line template configuration mode for the default line template.</li> </ul>
<b>Step 3</b>	Configure the terminal attribute settings for the specified template using the commands in line template configuration mode.	—
<b>Step 4</b>	Use one of the following commands: <ul style="list-style-type: none"> <li>• <b>end</b></li> <li>• <b>commit</b></li> </ul> <b>Example:</b> RP/0/RP0/CPU0:router(config-line)# end	Saves configuration changes. <ul style="list-style-type: none"> <li>• When you issue the <b>end</b> command, the system prompts you to commit changes:</li> </ul> <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre>

	Command or Action	Purpose
	or  RP/0/RP0/CPU0:router(config-line)# commit	<ul style="list-style-type: none"> <li>• Entering <b>yes</b> saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.</li> <li>• Entering <b>no</b> exits the configuration session and returns the router to EXEC mode without committing the configuration changes.</li> <li>• Entering <b>cancel</b> leaves the router in the current configuration session without exiting or committing the configuration changes.</li> <li>• Use the <b>commit</b> command to save the configuration changes to the running configuration file and remain within the configuration session.</li> </ul>

## Creating and Modifying vty Pools

This task explains how to create and modify vty pools.

### SUMMARY STEPS

1. **configure**
2. **telnet {ipv4 | ipv6} server max-servers limit**
3. **line template template-name**
4. Configure the terminal attribute settings for the specified line template using the commands in line template configuration mode.
5. **exit**
6. **vty-pool {default | pool-name | eem} first-vty last-vty [line-template {default | template-name}]**
7. Use the **commit** or **end** command.

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>configure</b>  <b>Example:</b>  RP/0/RP0/CPU0:router# configure	Enters mode.
<b>Step 2</b>	<b>telnet {ipv4   ipv6} server max-servers limit</b>  <b>Example:</b>  RP/0/RP0/CPU0:router(config)# telnet ipv4 server max-servers 10	Specifies the number of allowable Telnet servers. Up to 100 Telnet servers are allowed.  <b>Note</b> By default no Telnet servers are allowed. You must configure this command in order to enable the use of Telnet servers.

	Command or Action	Purpose
<b>Step 3</b>	<b>line template</b> <i>template-name</i> <b>Example:</b> <pre>RP/0/RP0/CPU0:router(config)# line template 1</pre>	Enters line template configuration mode for a user-defined template.
<b>Step 4</b>	Configure the terminal attribute settings for the specified line template using the commands in line template configuration mode.	—
<b>Step 5</b>	<b>exit</b> <b>Example:</b> <pre>RP/0/RP0/CPU0:router(config-line)# exit</pre>	Exits line template configuration mode and returns the router to global configuration mode.
<b>Step 6</b>	<b>vtty-pool</b> { <b>default</b>   <i>pool-name</i>   <b>eem</b> } <i>first-vty last-vty</i> [ <b>line-template</b> { <b>default</b>   <i>template-name</i> }] <b>Example:</b> <pre>RP/0/RP0/CPU0:router(config)# vty-pool default 0 5 line-template default</pre> or <pre>RP/0/RP0/CPU0:router(config)# vty-pool pool1 5 50 line-template template1</pre> or <pre>RP/0/RP0/CPU0:router(config)# vty-pool eem 100 105 line-template template1</pre>	Creates or modifies vty pools. <ul style="list-style-type: none"> <li>If you do not specify a line template with the <b>line-template</b> keyword, a vty pool defaults to the default line template.</li> <li><b>default</b> —Configures the default vty pool. <ul style="list-style-type: none"> <li>The default vty pool must start at vty 0 and must contain a minimum of five vtys (vtys 0 through 4).</li> <li>You can resize the default vty pool by increasing the range of vtys that compose the default vty pool.</li> </ul> </li> <li><i>pool-name</i> —Creates a user-defined vty pool. <ul style="list-style-type: none"> <li>A user-defined pool must start at least at vty 5, depending on whether the default vty pool has been resized.</li> <li>If the range of vtys for the default vty pool has been resized, use the first range value free from the default line template. For example, if the range of vtys for the default vty pool has been configured to include 10 vtys (vty 0 through 9), the range value for the user-defined vty pool must start with vty 10.</li> </ul> </li> <li><b>eem</b> —Configures the embedded event manager pool. <ul style="list-style-type: none"> <li>The default embedded event manager vty pool must start at vty 100 and must contain a minimum of six vtys (vtys 100 through 105).</li> </ul> </li> </ul>

	Command or Action	Purpose
		<ul style="list-style-type: none"> <li>• <b>line-template</b> <i>template-name</i> —Configures the vty pool to reference a user-defined template.</li> </ul>
<b>Step 7</b>	Use the <b>commit</b> or <b>end</b> command.	<p><b>commit</b> —Saves the configuration changes and remains within the configuration session.</p> <p><b>end</b> —Prompts user to take one of these actions:</p> <ul style="list-style-type: none"> <li>• <b>Yes</b> — Saves configuration changes and exits the configuration session.</li> <li>• <b>No</b> —Exits the configuration session without committing the configuration changes.</li> <li>• <b>Cancel</b> —Remains in the configuration session, without committing the configuration changes.</li> </ul>

## Monitoring Terminals and Terminal Sessions

This task explains how to monitor terminals and terminal sessions using the **show EXEC** commands available for physical and terminal lines.



**Note** The commands can be entered in any order.

### SUMMARY STEPS

1. (Optional) **show line** [**aux location** *node-id* | **console location** *node-id* | **vty number**]
2. (Optional) **show terminal**
3. (Optional) **show users**

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	(Optional) <b>show line</b> [ <b>aux location</b> <i>node-id</i>   <b>console location</b> <i>node-id</i>   <b>vty number</b> ]  <b>Example:</b>  <pre>RP/0/RP0/CPU0:router# show line</pre>	Displays the terminal parameters of terminal lines. <ul style="list-style-type: none"> <li>• Specifying the <b>show line aux location</b> <i>node-id</i> EXEC command displays the terminal parameters of the auxiliary line.</li> <li>• Specifying the <b>show line console location</b> <i>node-id</i> EXEC command displays the terminal parameters of the console.               <ul style="list-style-type: none"> <li>• For the <b>location</b> <i>node-id</i> keyword and argument, enter the location of the Route Processor (RP) on which the respective auxiliary or console port resides.</li> </ul> </li> </ul>

	Command or Action	Purpose
		<ul style="list-style-type: none"> <li>The <i>node-id</i> argument is expressed in the format of <i>rack/slot/module</i>.</li> <li>Specifying the <b>show line vty number EXEC</b> command displays the terminal parameters for the specified vty.</li> </ul>
<b>Step 2</b>	(Optional) <b>show terminal</b> <b>Example:</b> <pre>RP/0/RP0/CPU0:router# show terminal</pre>	Displays the terminal attribute settings for the current terminal line.
<b>Step 3</b>	(Optional) <b>show users</b> <b>Example:</b> <pre>RP/0/RP0/CPU0:router# show users</pre>	Displays information about the active lines on the router.

## Configuration Examples for Implementing Physical and Virtual Terminals

### Modifying the Console Template: Example

This configuration example shows how to modify the terminal attribute settings for the console line template:

```
line console
  exec-timeout 0 0
  escape-character 0x5a
  session-limit 10
  disconnect-character 0x59
  session-timeout 100
  transport input telnet
  transport output telnet
```

In this configuration example, the following terminal attributes are applied to the console line template:

- The EXEC time out for terminal sessions is set to 0 minutes, 0 seconds. Setting the EXEC timeout to 0 minutes and 0 seconds disables the EXEC timeout function; thus, the EXEC session for the terminal session will never time out.
- The escape character is set to the 0x5a hexadecimal value (the 0x5a hexadecimal value translates into the “Z” character).
- The session limit for outgoing terminal sessions is set to 10 connections.
- The disconnect character is set to 0x59 hexadecimal value (the 0x59 hexadecimal character translates into the “Y” character).

- The session time out for outgoing terminal sessions is set to 100 minutes (1 hour and 40 minutes).
- The allowed transport protocol for incoming terminal sessions is Telnet.
- The allowed transport protocol for outgoing terminal sessions is Telnet.

To verify that the terminal attributes for the console line template have been applied to the console, use the **show line** command:

```
RP/0/RP0/CPU0:router# show line console location 0/0/CPU0

Tty          Speed      Modem  Uses   Noise  Overruns      Acc I/O
* con0/0/CPU0  9600      -      -      -      0/0          -/-

Line con0_0_CPU0, Location "Unknown", Type "Unknown"
Length: 24 lines, Width: 80 columns
Baud rate (TX/RX) is 9600, 1 parity, 2 stopbits, 8 databits
Template: console
Config:
Allowed transports are telnet.
```

### Modifying the Default Template: Example

This configuration example shows how to override the terminal settings for the default line template:

```
line default
  exec-timeout 0 0
  width 512
  length 512
```

In this example, the following terminal attributes override the default line template default terminal attribute settings:

- The EXEC timeout for terminal sessions is set to 0 minutes and 0 seconds. Setting the EXEC timeout to 0 minutes and 0 seconds disables the EXEC timeout function; thus, the EXEC session for the terminal session will never time out (the default EXEC timeout for the default line template is 10 minutes).
- The width of the terminal screen for the terminals referencing the default template is set to 512 characters (the default width for the default line template is 80 characters).
- The length, the number of lines that will display at one time on the terminal referencing the default template, is set to 512 lines (the default length for the default line template is 24 lines).

### Configuring a User-Defined Template to Reference the Default vty Pool: Example

This configuration example shows how to configure a user-defined line template (named test in this example) for vtys and to configure the line template test to reference the default vty pool:

```
line template test
  exec-timeout 100 0
  width 100
  length 100
```

```
exit
vty-pool default 0 4 line-template test
```

### Configuring a User-Defined Template to Reference a User-Defined vty Pool: Example

This configuration example shows how to configure a user-defined line template (named test2 in this example) for vtys and to configure the line template test to reference a user-defined vty pool (named pool1 in this example):

```
line template test2
  exec-timeout 0 0
  session-limit 10
  session-timeout 100
  transport input all
  transport output all
  exit
vty-pool pool1 5 50 line-template test2
```

### Configuring a User-Defined Template to Reference the Fault Manager vty Pool: Example

This configuration example shows how to configure a user-defined line template (named test3 in this example) for vtys and to configure the line template test to reference the fault manager vty pool:

```
line template test3
  width 110
  length 100
  session-timeout 100
  exit
vty-pool eem 100 106 line-template test3
```





## CHAPTER 4

# Configuring Simple Network Management Protocol

---

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.

This module describes the tasks you need to implement SNMP on your Cisco IOS XR network.

- [Prerequisites for Implementing SNMP, on page 15](#)
- [Restrictions for SNMP use on Cisco IOS XR Software, on page 15](#)
- [Information about Implementing SNMP, on page 16](#)
- [Custom MIB Support Using SNMP Operation Script, on page 22](#)
- [Session MIB support on subscriber sessions , on page 24](#)
- [How to Implement SNMP on Cisco IOS XR Software, on page 26](#)

## Prerequisites for Implementing SNMP

You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

## Restrictions for SNMP use on Cisco IOS XR Software

SNMP outputs are only 32-bits wide and therefore cannot display any information greater than  $2^{32}$ .  $2^{32}$  is equal to 4.29 Gigabits.



---

**Note** A 10 Gigabit interface is greater than  $2^{32}$ , so if you are trying to display speed information regarding the interface, you might see concatenated results.

---

To display correct speed of an interface greater than 10 Gigabit, ifHighSpeed can be used.

The recommended maximum number of object identifiers (OIDs) that can be accommodated in a single SNMP request is 75. A request with more than 75 OIDs can result in SNMP requests being dropped with SNMP polling timeout.

# Information about Implementing SNMP

To implement SNMP, you need to understand the concepts described in this section.

## SNMP Functional Overview

The SNMP framework consists of three parts:

- SNMP manager
- SNMP agent
- Management Information Base (MIB)

## SNMP Manager

The SNMP manager is the system used to control and monitor the activities of network hosts using SNMP. The most common managing system is called a *network management system* (NMS). The term NMS can be applied to either a dedicated device used for network management, or the applications used on such a device. A variety of network management applications are available for use with SNMP. These features range from simple command-line applications to feature-rich graphical user interfaces (such as the CiscoWorks 2000 line of products).

## SNMP Agent

The SNMP agent is the software component within the managed device that maintains the data for the device and reports these data, as needed, to managing systems. The agent and MIB reside on the router. To enable the SNMP agent, you must define the relationship between the manager and the agent.

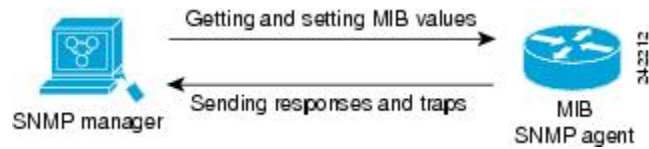
## MIB

The *Management Information Base* (MIB) is a virtual information storage area for network management information, which consists of collections of managed objects. Within the MIB there are collections of related objects, defined in MIB modules. MIB modules are written in the SNMP MIB module language, as defined in STD 58, RFC 2578, RFC 2579, and RFC 2580. Note that individual MIB modules are also referred to as MIBs; for example, the Interfaces Group MIB (IF-MIB) is a MIB module within the MIB on your system.

The SNMP agent contains MIB variables whose values the SNMP manager can request or change through Get or Set operations. A manager can get a value from an agent or store a value into that agent. The agent gathers data from the MIB, the repository for information about device parameters and network data. The agent can also respond to manager requests to get or set data.

This figure illustrates the communications relationship between the SNMP manager and agent. A manager can send the agent requests to get and set MIB values. The agent can respond to these requests. Independent of this interaction, the agent can send unsolicited notifications (traps) to the manager to notify the manager of network conditions.

**Figure 1: Communication Between an SNMP Agent and Manager**



### IP-MIB Support

RFC4293 IP-MIB was specifically designed to provide IPv4 and IPv6 statistics individually. The **ipIfStatsTable** defined in RFC 4293, lists the interface specific statistics. IPv6 statistics support in **ipIfStatsTable** was added earlier but, IOS-XR implementation of IP-MIB did not support IPv4 statistics as per RFC4293 in earlier releases.

IOS-XR implementation of IP-MIB supports IPv4 statistics as per RFC4293. This will enable you to collect the IPV4 and IPv6 statistics separately for each interface. The **ipIfStatsTable** is indexed by two **sub-ids address type (IPv4 or IPv6)** and the **interface ifindex[1]**. The implementation of IP-MIB support for IPv4 and IPv6 is separated for better readability and maintainability.

The list of OIDs added to the **ipIfStatsTable** for IPv4 statistics are:

- ipIfStatsInReceives
- ipIfStatsHCInReceives
- ipIfStatsInOctets
- ipIfStatsHCInOctets
- ipIfStatsOutTransmits
- ipIfStatsHCOutTransmits
- ipIfStatsOutOctets
- ipIfStatsHCOutOctets
- ipIfStatsDiscontinuityTime

For more information on the list of new OIDs added for IPv4 statistics, see [SNMP OID Navigator](#).

## SNMP Versions

Cisco IOS XR software supports the following versions of SNMP:

- Simple Network Management Protocol Version 1 (SNMPv1)
- Simple Network Management Protocol Version 2c (SNMPv2c)
- Simple Network Management Protocol Version 3 (SNMPv3)

Both SNMPv1 and SNMPv2c use a community-based form of security. The community of managers able to access the agent MIB is defined by an IP address access control list and password.

SNMPv2c support includes a bulk retrieval mechanism and more detailed error message reporting to management stations. The bulk retrieval mechanism supports the retrieval of tables and large quantities of information, minimizing the number of round-trips required. The SNMPv2c improved error handling support

includes expanded error codes that distinguish different kinds of error conditions; these conditions are reported through a single error code in SNMPv1. Error return codes now report the error type. Three kinds of exceptions are also reported: no such object exceptions, no such instance exceptions, and end of MIB view exceptions.

SNMPv3 is a security model. A *security model* is an authentication strategy that is set up for a user and the group 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 will determine which security mechanism is employed when an SNMP packet is handled. See [Security Models and Levels for SNMPv1, v2, v3, on page 19](#) for a list of security levels available in SNMPv3. The SNMPv3 feature supports RFCs 3411 to 3418.

You must configure the SNMP agent to use the version of SNMP supported by the management station. An agent can communicate with multiple managers; for this reason, you can configure the Cisco IOS-XR software to support communications with one management station using the SNMPv1 protocol, one using the SNMPv2c protocol, and another using SMNPv3.

## Comparison of SNMPv1, v2c, and v3

SNMP v1, v2c, and v3 all support the following operations:

- **get-request**—Retrieves a value from a specific variable.
- **get-next-request**—Retrieves the value following the named variable; this operation is often used to retrieve variables from within a table. With this operation, an SNMP manager does not need to know the exact variable name. The SNMP manager searches sequentially to find the needed variable from within the MIB.
- **get-response**—Operation that replies to a get-request, get-next-request, and set-request sent by an NMS.
- **set-request**—Operation that stores a value in a specific variable.
- **trap**—Unsolicited message sent by an SNMP agent to an SNMP manager when some event has occurred.

This table identifies other key SNMP features supported by the SNMP v1, v2c, and v3.

**Table 2: SNMPv1, v2c, and v3 Feature Support**

Feature	SNMP v1	SNMP v2c	SNMP v3
Get-Bulk Operation	No	Yes	Yes
Inform Operation	No	Yes	Yes
64 Bit Counter	No	Yes	Yes
Textual Conventions	No	Yes	Yes
Authentication	No	No	Yes
Privacy (Encryption)	No	No	Yes
Authorization and Access Controls (Views)	No	No	Yes

## Security Models and Levels for SNMPv1, v2, v3

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

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

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

The below table identifies what the combinations of security models and levels mean.

**Table 3: SNMP Security Models and Levels**

Model	Level	Authentication	Encryption	What Happens
v1	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
v2c	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
v3	noAuthNoPriv	Username	No	Uses a username match for authentication.
v3	authNoPriv	HMAC-MD5 or HMAC-SHA	No	Provides authentication based on the HMAC <sup>1</sup> -MD5 <sup>2</sup> algorithm or the HMAC-SHA <sup>3</sup> .
v3	authPriv	HMAC-MD5 or HMAC-SHA	DES	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides DES <sup>4</sup> 56-bit encryption in addition to authentication based on the CBC <sup>5</sup> DES (DES-56) standard.
v3	authPriv	HMAC-MD5 or HMAC-SHA	3DES	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides 168-bit 3DES <sup>6</sup> level of encryption.
v3	authPriv	HMAC-MD5 or HMAC-SHA	AES	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides 128-bit AES <sup>7</sup> level of encryption.

<sup>1</sup> Hash-Based Message Authentication Code

<sup>2</sup> Message Digest 5

<sup>3</sup> Secure Hash Algorithm

<sup>4</sup> Data Encryption Standard

<sup>5</sup> Cipher Block Chaining

<sup>6</sup> Triple Data Encryption Standard

<sup>7</sup> Advanced Encryption Standard

Use of 3DES and AES encryption standards requires that the security package be installed. For information on installing software packages, see *Upgrading and Managing Cisco IOS XR Software*.

## SNMPv3 Benefits

SNMPv3 provides secure access to devices by providing authentication, encryption and access control. These added security benefits secure SNMP against the following security threats:

- **Masquerade**—The threat that an SNMP user may assume the identity of another SNMP user to perform management operations for which that SNMP user does not have authorization.
- **Message stream modification**—The threat that messages may be maliciously reordered, delayed, or replayed (to an extent that is greater than can occur through the natural operation of a subnetwork service) to cause SNMP to perform unauthorized management operations.
- **Disclosure**—The threat that exchanges between SNMP engines could be eavesdropped. Protecting against this threat may be required as a matter of local policy.

In addition, SNMPv3 provides access control over protocol operations on SNMP managed objects.

## SNMPv3 Costs

SNMPv3 authentication and encryption contribute to a slight increase in the response time when SNMP operations on MIB objects are performed. This cost is far outweighed by the security advantages provided by SNMPv3.

This table shows the order of response time (from least to greatest) for the various security model and security level combinations.

**Table 4: Order of Response Times from Least to Greatest**

Security Model	Security Level
SNMPv2c	noAuthNoPriv
SNMPv3	noAuthNoPriv
SNMPv3	authNoPriv
SNMPv3	authPriv

## 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**—Ensures that the claimed identity of the user on whose behalf received data was originated is confirmed.
- **Message confidentiality**—Ensures that information is not made available or disclosed to unauthorized individuals, entities, or processes.

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

USM uses two authentication protocols:

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

USM uses Cipher Block Chaining (CBC)-DES (DES-56) as the privacy protocol for message encryption.

## View-Based Access Control Model

The View-Based Access Control Model (VACM) enables SNMP users to control access to SNMP managed objects by supplying read, write, or notify access to SNMP objects. It prevents access to objects restricted by views. These access policies can be set when user groups are configured with the **snmp-server group** command.

### MIB Views

For security reasons, it is often valuable to be able to restrict the access rights of some groups to only a subset of the management information within the management domain. To provide this capability, access to a management object is controlled through MIB views, which contain the set of managed object types (and, optionally, the specific instances of object types) that can be viewed.

### Access Policy

Access policy determines the access rights of a group. The three types of access rights are as follows:

- read-view access—The set of object instances authorized for the group when objects are read.
- write-view access—The set of object instances authorized for the group when objects are written.
- notify-view access—The set of object instances authorized for the group when objects are sent in a notification.

## IP Precedence and DSCP Support for SNMP

SNMP IP Precedence and differentiated services code point (DSCP) support delivers QoS specifically for SNMP traffic. You can change the priority setting so that SNMP traffic generated in a router is assigned a specific QoS class. The IP Precedence or IP DSCP code point value is used to determine how packets are handled in weighted random early detection (WRED).

After the IP Precedence or DSCP is set for the SNMP traffic generated in a router, different QoS classes cannot be assigned to different types of SNMP traffic in that router.

The IP Precedence value is the first three bits in the type of service (ToS) byte of an IP header. The IP DSCP code point value is the first six bits of the differentiate services (DiffServ Field) byte. You can configure up to eight different IP Precedence markings or 64 different IP DSCP markings.

# Custom MIB Support Using SNMP Operation Script

Table 5: Feature History Table

Feature Name	Release Information	Feature Description
Custom MIB Support Using SNMP Operations Script	Release 7.5.3	<p>Now you don't have to upgrade to the latest Cisco IOS XR Software release to access a new Management Information Base (MIB). This feature allows you to add a custom script to get support for custom MIB that is not implemented on Cisco IOS XR Software. Custom MIB fetches the required data from an operational database that is already available on the router and returns it on polling the Object Identifier (OID).</p> <p>This feature introduces the following commands:</p> <ul style="list-style-type: none"> <li>• <a href="#">snmp-server script</a></li> <li>• <a href="#">script snmp</a></li> </ul> <p>This feature also adds the following unified models, you can access these unified models in the <a href="#">Github</a> repository.</p> <ul style="list-style-type: none"> <li>• <a href="#">Cisco-IOS-XR-um-script-cfg</a></li> <li>• <a href="#">Cisco-IOS-XR-um-script-server-cfg</a></li> </ul>

The MIB is a virtual information storage area for network management information, which consists of collections of managed objects. The OID acts as an identifier to fetch the required data from MIB.

This feature introduces support for custom MIBs that are not implemented in Cisco IOS XR Software. Typically, developing a new MIB is a long and tedious process. Also, you must upgrade to a particular release to get the support of the new MIB.

With this feature, you can define a custom script for a given OID. This custom OID gets the data in the operational database already present on the router and returns it on polling the newly configured OID. SNMP request is sent from Network Management System (NMS) over User Datagram Protocol (UDP) to SNMP daemon. This request spawns customer scripts to fetch data that is related to OID in the request and the output of the script is converted to SNMP protocol data unit and sent to NMS.

## Prerequisites

- In the script, the Cython API `snmp_send_response` should be called with data of OID.



## Restrictions for Custom MIB

- The length of string data type OIDs must not cross 400 bytes.

## Create Custom MIB Using SNMP Script

### Configuration Example

In the below example, we create a script which creates OID 1.3.6.1.4.1.9.9.999998.10 to read lldp state.

1. Create a script to fetch required data from the operational database on the router.
2. Use the **describe** command, to fetch the process which executes the command.

```
Router#describe show lldp
The command is defined in lldp_cmds.parser
```

User needs ALL of the following taskids:

```
ethernet-services (READ) or optical (READ)
```

It will take the following actions:

```
Spawn the process:
  lldp_command "-s" "-g"
```

The output **lldp\_command "-s" "-g"** is used in the following script.

Here is a sample script named **show\_lldp\_string.py**. This is the command syntax used in the script.

```
import iosxr.snmp
import time
import subprocess as sp
import re
oid = iosxr.snmp.snmp_get_oid()
access_type = iosxr.snmp.snmp_get_access_type()
value = sp.getoutput("lldp_command \"-s\" \"-g\" ")
iosxr.snmp.snmp_send_response("1.3.6.1.4.1.9.9.999998.10", str(value), "OctetString")
```

3. Copy the script file to this location: `haddisk:/mirror/script-mgmt/snmp/`.
4. Use the **sha256sum file-name** command to generate the checksum of the script file.

```
Router:/haddisk:/mirror/script-mgmt/snmp]$sha256sum show_lldp_string.py
```

Here is a sample command output.

```
156345c2cbfc1a2725b5f5ecdfb23d30d9a25e894604890d88929d724946e7b3 show_lldp_string.py
```

5. Enter the configuration mode of the router.
6. Use the **snmp-server community public RW** command to enable read-write community string, where **public** is the read-write community.
7. Use the **snmp-server script script-oid OID-number script-filename file-name** command to map the script file to the custom OID.

```
Router#configure
```

```
Router(config)#snmp-server community public RW
```

```
Router(config)#snmp-server script script-oid 1.3.6.1.4.1.9.9.99998.10 script-filename
show_lldp_string.py
```

8. Use the **script snmp file-name checksum sha256 checksum-value** command to configure the checksum of the script file.

```
Router(config)#script snmp show_lldp_string.py checksum sha256
156345c2cbfc1a2725b5f5ecdfb23d30d9a25e894604890d88929d724946e7b3
```

**Note**

- The root OID number 1.3.6.1.4.1.9.9.99998 must be used and you can write any Custom OID number after the root OID number.

**Yang Data Model for Custom MIB**

You can programmatically perform the same configuration using the following unified data models also. To get started with using data models, see the *Programmability Configuration Guide for Cisco 8000 Series Routers*.

Operational Data	Unified Data Model	CLI Commands
Maps script file to the custom OID.	Cisco-IOS-XR-um-script-server-cfg	<b>snmp-server script script-oid OID-number script-filename file-name</b>
Configures checksum for the newly added file-name in the Custom OID.	Cisco-IOS-XR-um-script-cfg	<b>script snmp file-name checksum sha256 checksum-value</b>

**Verification**

When snmp receives get request for the custom OID, following output is generated:

```
Router # snmpwalk -v2c -c public 5.36.7.100 1.3.6.1.4.1.9.9.99998.10
SNMPv2-SMI::enterprises.9.9.99998.10.0 = STRING: Global LLDP information:
  Status: ACTIVE
  LLDP Chassis ID: 0032.176e.a0df
  LLDP Chassis ID Subtype: MAC Address (IEEE 802-2001) Chassis Subtype
  LLDP System Name: POD-TN3
  LLDP advertisements are sent every 30 seconds
  LLDP hold time advertised is 120 seconds
  LLDP interface reinitialisation delay is 2 seconds
```

## Session MIB support on subscriber sessions

SNMP monitoring requires information about subscribers of all types. The CISCO-SUBSCRIBER-SESSION-MIB is defined to model per-subscriber data as well as aggregate subscriber (PPPoE) data. It is required to support notifications (traps) for aggregate session counts crossing configured thresholds. Generic MIB Data Collector Manager (DCM) support for CISCO-SUBSCRIBER-SESSION-MIB, helps faster data collection and also better handling of parallel data.

## 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. On Cisco IOS XR software, unsolicited (asynchronous) notifications can be generated only as *traps*. Traps are messages alerting the SNMP manager to a condition on the network. Notifications can indicate improper user authentication, restarts, the closing of a connection, loss of connection to a neighbor router, or other significant events.



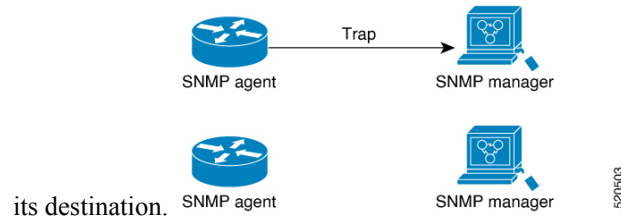
**Note** Inform requests (inform operations) are supported in Cisco IOS XR software.

Traps are less reliable than informs because the receiver does not send any acknowledgment when it receives a trap. The sender 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 manager does not receive an inform request, it does not send a response. If the sender never receives a response, the inform request can be sent again. Thus, informs are more likely to reach their intended destination.

However, traps are often preferred because informs consume more resources in the router and in the network. Unlike a trap, which is discarded as soon as it is sent, an inform request must be held in memory until a response is received or the request times out. Also, traps are sent only once, and an inform may be retried several times. The retries increase traffic and contribute to a higher overhead on the network. Thus, traps and inform requests provide a trade-off between reliability and resources.

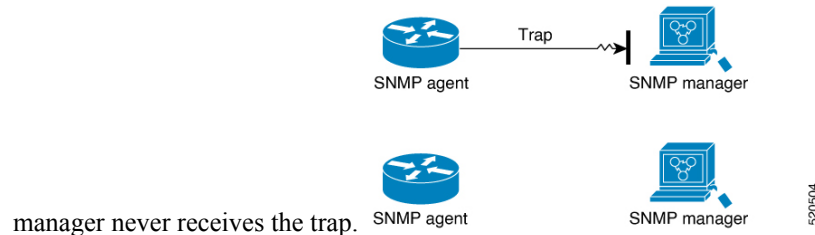
**Figure 2: Trap Received by the SNMP Manager**

In this illustration, the agent router sends a trap to the SNMP manager. Although the manager receives the trap, it does not send any acknowledgment to the agent. The agent has no way of knowing that the trap reached



**Figure 3: Trap Not Received by the SNMP Manager**

In this illustration, the agent sends a trap to the manager, but the trap does not reach the manager. Because the agent has no way of knowing that the trap did not reach its destination, the trap is not sent again. The



## Session Types

The supported session types are:

- PPPoE

- IP SUB PKT
- IP SUB DHCP

## How to Implement SNMP on Cisco IOS XR Software

This section describes how to implement SNMP.

The **snmp-server** commands enable SNMP on Management Ethernet interfaces by default. For information on how to enable SNMP server support on other inband interfaces, see the *Implementing Management Plane Protection on Cisco IOS XR Software* module in *System Security Configuration Guide for Cisco 8000 Series Routers*.

### Configuring SNMPv3

This task explains how to configure SNMPv3 for network management and monitoring.



**Note** No specific command enables SNMPv3; the first **snmp-server** global configuration command (config), that you issue enables SNMPv3. Therefore, the sequence in which you issue the **snmp-server** commands for this task does not matter.

#### Step 1 **configure**

**Example:**

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

#### Step 2 (Optional) **snmp-server engineid local engine-id**

**Example:**

```
RP/0/RP0/CPU0:router# snmp-server engineID
local 00:00:00:09:00:00:00:a1:61:6c:20:61
```

Specifies the identification number of the local SNMP engine.

#### Step 3 **snmp-server view view-name oid-tree {included | excluded}**

**Example:**

```
RP/0/RP0/CPU0:router# snmp-server view
view_name 1.3.6.1.2.1.1.5 included
```

Creates or modifies a view record.

#### Step 4 **snmp-server group name {v1 | v2c | v3 {auth | noauth | priv}} [read view] [write view] [notify view] [access-list-name]**

**Example:**

```
RP/0/RP0/CPU0:router# snmp-server group
group_name v3 noauth read view_name1 write view_name2
```

Configures a new SNMP group or a table that maps SNMP users to SNMP views.

**Step 5** `snmp-server user username groupname {v1 | v2c | v3 [auth {md5 | sha} {clear | encrypted} auth-password [priv-des56 {clear | encrypted} priv-password]]} [access-list-name]`

**Example:**

```
RP/0/RP0/CPU0:router# snmp-server user
noauthuser group_name v3
```

Configures a new user to an SNMP group.

**Step 6** Use the **commit** or **end** command.

**commit**—Saves the configuration changes and remains within the configuration session.

**end**—Prompts user to take one of these actions:

- **Yes**—Saves configuration changes and exits the configuration session.
- **No**—Exits the configuration session without committing the configuration changes.
- **Cancel**—Remains in the configuration session, without committing the configuration changes.

**Step 7** (Optional) **show snmp**

**Example:**

```
RP/0/RP0/CPU0:router# show snmp
```

Displays information about the status of SNMP.

**Step 8** (Optional) **show snmp engineid**

**Example:**

```
RP/0/RP0/CPU0:router# show snmp engineid
```

Displays information about the local SNMP engine.

**Step 9** (Optional) **show snmp group**

**Example:**

```
RP/0/RP0/CPU0:router# show snmp group
```

Displays information about each SNMP group on the network.

**Step 10** (Optional) **show snmp users**

**Example:**

```
RP/0/RP0/CPU0:router# show snmp users
```

Displays information about each SNMP username in the SNMP users table.

**Step 11** (Optional) **show snmp view**

**Example:**

```
RP/0/RP0/CPU0:router# show snmp view
```

Displays information about the configured views, including the associated MIB view family name, storage type, and status.

## Configure to Drop Error PDUs

Perform this configuration to avoid error PDUs being sent out of router when polled with incorrect SNMPv3 user name. If the configuration is not set, it will respond with error PDUs by default. After applying this configuration, when router is polled with unknown SNMPv3 user name, the NMS will get time out instead of getting unknown user name error code.

### Step 1 **configure**

**Example:**

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

### Step 2 **snmp-server drop unknown-user**

**Example:**

```
RP/0/RP0/CPU0:router(config)# snmp-server drop unknown-user
```

Drop the error PDUs when the router is polled with incorrect SNMPv3 user name.

### Step 3 Use the **commit** or **end** command.

**commit** —Saves the configuration changes and remains within the configuration session.

**end** —Prompts user to take one of these actions:

- **Yes** — Saves configuration changes and exits the configuration session.
- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

## Configuring SNMPv3: Examples

### Setting an Engine ID

This example shows how to set the identification of the local SNMP engine:

```
snmp-server engineID local 00:00:00:09:00:00:00:a1:61:6c:20:61
```



---

**Note** After the engine ID has been configured, the SNMP agent restarts.

---

### Verifying the Identification of the Local SNMP Engines

This example shows how to verify the identification of the local SNMP engine:

```
config
  show snmp engineid

SNMP engineID 00000009000000a1ffffffff
```

### Creating a View

There are two ways to create a view:

- You can include the object identifier (OID) of an ASN.1 subtree of a MIB family from a view by using the **included** keyword of the **snmp-server view** command.
- You can exclude the OID subtree of the ASN.1 subtree of a MIB family from a view by using the **excluded** keyword of the **snmp-server view** command.

This example shows how to create a view that includes the sysName (1.3.6.1.2.1.1.5) object:

```
config
  snmp-server view SNMP_VIEW1 1.3.6.1.2.1.1.5 included
```

This example shows how to create a view that includes all the OIDs of a system group:

```
config
  snmp-server view SNMP_VIEW1 1.3.6.1.2.1.1 included
```

This example shows how to create a view that includes all the OIDs under the system group except the sysName object (1.3.6.1.2.1.1.5), which has been excluded:

```
config
  snmp-server view SNMP_VIEW1 1.3.6.1.2.1.1 included
  snmp-server view SNMP_VIEW1 1.3.6.1.2.1.1.5 excluded
```

### Verifying Configured Views

This example shows how to display information about the configured views:

```
RP/0/RP0/CPU0:router# show snmp view

v1default 1.3.6.1 - included nonVolatile active
SNMP_VIEW1 1.3.6.1.2.1.1 - included nonVolatile active
```

```
SNMP_VIEW1 1.3.6.1.2.1.1.5 - excluded nonVolatile active
```

### Creating Groups

If you do not explicitly specify a notify, read, or write view, the Cisco IOS XR software uses the v1 default (1.3.6.1). This example shows how to create a group that utilizes the default view:

```
RP/0/RP0/CPU0:router# snmp-server group group-name v3 auth
```

The following configuration example shows how to create a group that has read access to all the OIDs in the system except the sysUpTime object (1.3.6.1.2.1.1.3), which has been excluded from the view applied to the group, but write access only to the sysName object (1.3.6.1.2.1.1.5):

```
!
snmp-server view view_name1 1.3.6.1.2.1.1 included
snmp-server view view_name1 1.3.6.1.2.1.1.3 excluded
snmp-server view view_name2 1.3.6.1.2.1.1.5 included
snmp-server group group_name1 v3 auth read view_name1 write view_name2
!
```

### Verifying Groups

This example shows how to verify the attributes of configured groups:

```
RP/0/RP0/CPU0:router# show snmp group

groupname: group_name1                security model:usm
readview  : view_name1                writeview: view_name2
notifyview: v1default
row status: nonVolatile
```

### Creating and Verifying Users

Given the following SNMPv3 view and SNMPv3 group configuration:

```
!
snmp-server view view_name 1.3.6.1.2.1.1 included
snmp-server group group_name v3 noauth read view_name write view-name
!
```

This example shows how to create a noAuthNoPriv user with read and write view access to a system group:

```
config
snmp-server user noauthuser group_name v3
```






---

**Note** The user must belong to a noauth group before a noAuthNoPriv user can be created.

---

This example shows how to verify the attributes that apply to the SNMP user:

```
RP/0/RP0/CPU0:router# show snmp user

User name: noauthuser
Engine ID: localSnmPID
storage-type: nonvolatile active
```

Given the following SNMPv3 view and SNMPv3 group configuration:

```
!
snmp-server view SNMP_VIEW1 1.3.6.1.2.1.1 included
snmp-server group SNMP_GROUP1 v3 auth notify SNMP_VIEW1 read SNMP_VIEW1 write SNMP_VIEW1
!
```

This example shows how to create a user with authentication (including encryption), read, and write view access to a system group:

```
config
snmp-server user userv3authpriv SNMP_GROUP1 v3 auth md5 password123 priv aes 128 password123
```

Given the following SNMPv3 view and SNMPv3 group configuration:

```
!
snmp-server view view_name 1.3.6.1.2.1.1 included
snmp group group_name v3 priv read view_name write view_name
!
```

This example shows how to create authNoPriv user with read and write view access to a system group:

```
RP/0/RP0/CPU0:router# snmp-server user authuser group_name v3 auth md5 clear auth_passwd
```




---

**Note** As the group is configured at a security level of Auth, the user must be configured as “auth” at a minimum to access this group (“priv” users could also access this group). The authNoPriv user configured in this group, authuser, must supply an authentication password to access the view. In the example, auth\_passwd is set as the authentication password string. Note that **clear** keyword is specified before the auth\_passwd password string. The **clear** keyword indicates that the password string being supplied is unencrypted.

---

This example shows how to verify the attributes that apply to SNMP user:

```
RP/0/RP0/CPU0:router# show snmp user

User name: authuser
```

```
Engine ID: localSnmpID
storage-type: nonvolatile active
```

Given the following SNMPv3 view and SNMPv3 group configuration:

```
!
snmp view view_name 1.3.6.1.2.1.1 included
snmp group group_name v3 priv read view_name write view_name
!
```

This example shows how to create an authPriv user with read and write view access to a system group:

```
config
snmp-server user privuser group_name v3 auth md5 clear auth_passwd priv des56 clear
priv_passwd
```




---

**Note** As the group has a security level of Priv, the user must be configured as a “priv” user to access this group. In this example, the user, `privuser`, must supply both an authentication password and privacy password to access the OIDs in the view.

---

This example shows how to verify the attributes that apply to the SNMP user:

```
RP/0/RP0/CPU0:router# show snmp user

User name: privuser
Engine ID: localSnmpID
storage-type: nonvolatile active
```

## Configuring SNMP Trap Notifications

The following example shows how to configure the router to send SNMP trap notifications.

### Step 1 **configure**

#### **Example:**

```
RP/0/RP0/CPU0:router# configure

Enters mode.
```

### Step 2 **snmp-servergroupname {v1v2v3 {auth | noauth | priv}} [readview]writeview [notifyview] [access-list-name]**

#### **Example:**

```
RP/0/RP0/CPU0:router# snmp-server group group_name v3 noauth read view_name1 writer view_name2

Configures a new SNMP group or a table that maps SNMP users to SNMP views.
```

**Step 3** `snmp-server user groupname {v1v2cv3 {auth | md5 | sha} {clear | encrypted} auth-password} [priv des56 {clear | access-list-name}]`

**Example:**

```
RP/0/RP0/CPU0:router# snmp-server group group_name v3 noauth read view_name1 writer view_name2
```

Configures a new SNMP group or a table that maps SNMP users to SNMP views.

**Step 4** `snmp-server user username groupname {v1v2cv3 {auth | md5 | sha} {clear | encrypted} auth-password} [priv des56 {clear | access-list-name}]`

**Example:**

```
RP/0/RP0/CPU0:routerconfig# snmp-server user noauthuser group_name v3
```

Configures a new SNMP group or a table that maps SNMP users to SNMP views.

**Step 5** `[ snmp-server host address [traps] [version {1 | 2c | 3 [auth | noauth | priv]}] community-string [udp-port port] [notification-type]`

**Example:**

```
RP/0/RP0/CPU0:router(config)# snmp-server host 12.26.25.61 traps version 3
noauth userV3noauth
```

Specifies SNMP trap notifications, the version of SNMP to use, the security level of the notifications, and the recipient (host) of the notifications.

**Step 6** `snmp-server traps [notification-type]`

**Example:**

```
RP/0/RP0/CPU0:router(config)# snmp-server traps bgp
```

Enables the sending of trap notifications and specifies the type of trap notifications to be sent.

- If a trap is not specified with the *notification-type* argument, all supported trap notifications are enabled on the router. To display which trap notifications are available on your router, enter the `snmp-server traps ?` command.

**Step 7** Use the `commit` or `end` command.

**commit** —Saves the configuration changes and remains within the configuration session.

**end** —Prompts user to take one of these actions:

- **Yes** — Saves configuration changes and exits the configuration session.
- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

**Step 8** (Optional) `show snmp host`

**Example:**

```
RP/0/RP0/CPU0:router# show snmp host
```

Displays information about the configured SNMP notification recipient (host), port number, and security model.

## Configure to Drop Error PDUs

Perform this configuration to avoid error PDUs being sent out of router when polled with incorrect SNMPv3 user name. If the configuration is not set, it will respond with error PDUs by default. After applying this configuration, when router is polled with unknown SNMPv3 user name, the NMS will get time out instead of getting unknown user name error code.

### Step 1 **configure**

#### Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

### Step 2 **snmp-server drop unknown-user**

#### Example:

```
RP/0/RP0/CPU0:router(config)# snmp-server drop unknown-user
```

Drop the error PDUs when the router is polled with incorrect SNMPv3 user name.

### Step 3 Use the **commit** or **end** command.

**commit** —Saves the configuration changes and remains within the configuration session.

**end** —Prompts user to take one of these actions:

- **Yes** — Saves configuration changes and exits the configuration session.
- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

## Configuring Trap Notifications: Example

The following example configures an SNMP agent to send out different types of traps. The configuration includes a v2c user, a noAuthNoPriv user, anauthNoPriv user, and an AuthPriv user.



**Note** The default User Datagram Protocol (UDP) port is 161. If you do not specify a UDP port with the **udp-port** keyword and *port* argument, then the configured SNMP trap notifications are sent to port 161.

```
!
snmp-server host 10.50.32.170 version 2c public udp-port 2345
snmp-server host 10.50.32.170 version 3 auth userV3auth udp-port 2345
snmp-server host 10.50.32.170 version 3 priv userV3priv udp-port 2345
snmp-server host 10.50.32.170 version 3 noauth userV3noauth udp-port 2345
snmp-server user userv2c groupv2c v2c
```

```

snmp-server user userV3auth groupV3auth v3 auth md5 encrypted 140F0A13
snmp-server user userV3priv groupV3priv v3 auth md5 encrypted 021E1C43 priv des56 encrypted
1110001C
snmp-server user userV3noauth groupV3noauth v3 LROwner
snmp-server view view_name 1.3 included
snmp-server community public RW
snmp-server group groupv2c v2c read view_name
snmp-server group groupV3auth v3 auth read view_name
snmp-server group groupV3priv v3 priv read view_name
snmp-server group groupV3noauth v3 noauth read view_name
!
```

This example shows how to verify the configuration SNMP trap notification recipients host, the recipients of SNMP trap notifications. The output displays the following information:

- IP address of the configured notification host
- UDP port where SNMP notification messages are sent
- Type of trap configured
- Security level of the configured user
- Security model configured

```

config
show snmp host

Notification host: 10.50.32.170 udp-port: 2345 type: trap
user: userV3auth security model: v3 auth

Notification host: 10.50.32.170 udp-port: 2345 type: trap
user: userV3noauth security model: v3 noauth

Notification host: 10.50.32.170 udp-port: 2345 type: trap
user: userV3priv security model: v3 priv

Notification host: 10.50.32.170 udp-port: 2345 type: trap
user: userv2c security model: v2c
```

## Setting the Contact, Location, and Serial Number of the SNMP Agent

This task explains how to set the system contact string, system location string, and system serial number of the SNMP agent.



**Note** The sequence in which you issue the **snmp-server** commands for this task does not matter.

### Step 1 configure

#### Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

**Step 2** (Optional) **snmp-server contact** *system-contact-string*

**Example:**

```
RP/0/RP0/CPU0:router(config)# snmp-server contact
Dial System Operator at beeper # 27345
```

Sets the system contact string.

**Step 3** (Optional) **snmp-server location** *system-location*

**Example:**

```
RP/0/RP0/CPU0:router(config)# snmp-server location
Building 3/Room 214
```

Sets the system location string.

**Step 4** (Optional) **snmp-server chassis-id** *serial-number*

**Example:**

```
RP/0/RP0/CPU0:router(config)# snmp-server chassis-id 1234456
```

Sets the system serial number.

**Step 5** Use the **commit** or **end** command.

**commit** —Saves the configuration changes and remains within the configuration session.

**end** —Prompts user to take one of these actions:

- **Yes** — Saves configuration changes and exits the configuration session.
- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

---

## Defining the Maximum SNMP Agent Packet Size

This task shows how to configure the largest SNMP packet size permitted when the SNMP server is receiving a request or generating a reply.




---

**Note** The sequence in which you issue the **snmp-server** commands for this task does not matter.

---

**Step 1** **configure**

**Example:**

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

**Step 2** (Optional) **snmp-server packetsize** *byte-count*

**Example:**

```
RP/0/RP0/CPU0:router(config)# snmp-server packetsize 1024
```

Sets the maximum packet size.

**Step 3** Use the **commit** or **end** command.

**commit** —Saves the configuration changes and remains within the configuration session.

**end** —Prompts user to take one of these actions:

- **Yes** — Saves configuration changes and exits the configuration session.
- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

## Changing Notification Operation Values

After SNMP notifications have been enabled, you can specify a value other than the default for the source interface, message queue length, or retransmission interval.

This task explains how to specify a source interface for trap notifications, the message queue length for each host, and the retransmission interval.



**Note** The sequence in which you issue the **snmp-server** commands for this task does not matter.

**Step 1** **configure**

**Example:**

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

**Step 2** (Optional) **snmp-server trap-source** *type interface-path-id*

**Example:**

```
RP/0/RP0/CPU0:router(config)# snmp-server trap-source POS 0/0/1/0
```

Specifies a source interface for trap notifications.

**Step 3** (Optional) **snmp-server queue-length** *length*

**Example:**

```
RP/0/RP0/CPU0:router(config)# snmp-server queue-length 20
```

Establishes the message queue length for each notification.

**Step 4** (Optional) **snmp-server trap-timeout** *seconds*

**Example:**

```
RP/0/RP0/CPU0:router(config)# snmp-server trap-timeout 20
```

Defines how often to resend notifications on the retransmission queue.

**Step 5** Use the **commit** or **end** command.

**commit** —Saves the configuration changes and remains within the configuration session.

**end** —Prompts user to take one of these actions:

- **Yes** — Saves configuration changes and exits the configuration session.
- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

## Setting IP Precedence and DSCP Values

This task describes how to configure IPv4 Precedence or IPv4 DSCP for SNMP traffic.

### Before you begin

SNMP must be configured.

**Step 1** **configure**

**Example:**

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

**Step 2** Use one of the following commands:

- **snmp-server ipv4 precedence** *value*
- **snmp-server ipv4 dscp** *value*

**Example:**

```
RP/0/RP0/CPU0:router(config)# snmp-server dscp 24
```

Configures an IPv4 precedence or IPv4 DSCP value for SNMP traffic.

**Step 3** Use the **commit** or **end** command.

**commit** —Saves the configuration changes and remains within the configuration session.



**end** —Prompts user to take one of these actions:

- **Yes** — Saves configuration changes and exits the configuration session.
- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

---

## Setting IPv6 Precedence and DSCP Values

This task describes how to configure IPv6 Precedence or IPv6 DSCP for SNMP traffic.

### Before you begin

SNMP must be configured.

---

#### Step 1 **configure**

##### Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

#### Step 2 Use one of the following commands:

- **snmp-server ipv6 precedence** *value*
- **snmp-server ipv6 dscp** *value*

##### Example:

```
RP/0/RP0/CPU0:router(config)# snmp-server dscp 24
```

Configures an IPv6 precedence or IPv6 DSCP value for SNMP traffic.

#### Step 3 Use the **commit** or **end** command.

**commit** —Saves the configuration changes and remains within the configuration session.

**end** —Prompts user to take one of these actions:

- **Yes** — Saves configuration changes and exits the configuration session.
- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

---

## Setting an IP Precedence Value for SNMP Traffic: Example

The following example shows how to set the SNMP IPv4 Precedence value to 7:

```

configure
  snmp-server ipv4 precedence 7
  exit

Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: y

```

The following example shows how to set the SNMP IPv6 Precedence value to 7:

```

configure
  snmp-server ipv6 precedence 7
  exit

Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: y

```

## Setting an IP DSCP Value for SNMP Traffic: Example

The following example shows how to set the IPv4 DSCP value of SNMP traffic to 45:

```

configure
  snmp-server ipv4 dscp 45
  exit

Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: y

```

The following example shows how to set the IPv6 DSCP value of SNMP traffic to 45:

```

configure
  snmp-server ipv6 dscp 45
  exit

Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: y

```

## Displaying SNMP Context Mapping

The SNMP agent serves queries based on SNMP contexts created by the client features. There is a context mapping table. Each entry in the context mapping table includes a context name, the name of the feature that created the context, and the name of the specific instance of the feature.

---

**show snmp context-mapping**

**Example:**

```
RP/0/RP0/CPU0:router# show snmp context-mapping
```

Displays the SNMP context mapping table.

---

## Monitoring Packet Loss

It is possible to monitor packet loss by configuring the generation of SNMP traps when packet loss exceeds a specified threshold. The configuration described in this task enables the creation of entries in the MIB tables of the EVENT-MIB. This can then be monitored for packet loss using SNMP GET operations.

### Before you begin



**Note** Entries created in the EVENT-MIB MIB tables using the configuration described in this task cannot be altered using an SNMP SET.

Entries to the EVENT-MIB MIB tables created using an SNMP SET cannot be altered using the configuration described in this task.

---

**snmp-server mibs eventmib packet-loss** *type interface-path-id* **falling** *lower-threshold* **interval** *sampling-interval*  
**rising** *upper-threshold*

### Example:

```
RP/0/RP0/CPU0:router(config)# snmp-server mibs eventmib packet-loss falling 1 interval 5 rising 2
```

Generates SNMP EVENT-MIB traps for the interface when the packet loss exceeds the specified thresholds. Up to 100 interfaces can be monitored.

**falling** *lower-threshold* —Specifies the lower threshold. When packet loss between two intervals falls below this threshold and an `mteTriggerRising` trap was generated previously, a SNMP `mteTriggerFalling` trap is generated. This trap is not generated until the packet loss exceeds the upper threshold and then falls back below the lower threshold.

**interval** *sampling-interval* —Specifies how often packet loss statistics are polled. This is a value between 5 and 1440 minutes, in multiples of 5.

**rising** *upper-threshold* —Specifies the upper threshold. When packet loss between two intervals increases above this threshold, a SNMP `mteTriggreRising` trap is generated. This trap is not generated until the packet loss drops below the lower threshold and then rises above the upper threshold.

---

## Configuring MIB Data to be Persistent

Many SNMP MIB definitions define arbitrary 32-bit indices for their object tables. MIB implementations often do a mapping from the MIB indices to some internal data structure that is keyed by some other set of data. In these MIB tables the data contained in the table are often other identifiers of the element being modelled. For example, in the ENTITY-MIB, entries in the `entPhysicalTable` are indexed by the 31-bit value, `entPhysicalIndex`, but the entities could also be identified by the `entPhysicalName` or a combination of the other objects in the table.

Because of the size of some MIB tables, significant processing is required to discover all the mappings from the 32-bit MIB indices to the other data which the network management station identifies the entry. For this reason, it may be necessary for some MIB indices to be persistent across process restarts, switchovers, or device reloads. The ENTITY-MIB entPhysicalTable and CISCO-CLASS-BASED-QOS-MIB are two such MIBs that often require index values to be persistent.

Also, because of query response times and CPU utilization during CISCO-CLASS-BASED-QOS-MIB statistics queries, it is desirable to cache service policy statistics.

---

**Step 1** (Optional) **snmp-server mibs cbqosmib persist**

**Example:**

```
RP/0/RP0/CPU0:router(config)# snmp-server mibs cbqosmib persist
```

Enables persistent storage of the CISCO-CLASS-BASED-QOS-MIB data.

**Step 2** (Optional) **snmp-server cbqosmib cache refresh time *time***

**Example:**

```
RP/0/RP0/CPU0:router(config)# snmp-server mibs cbqosmib cache
refresh time 45
```

Enables QoS MIB caching with a specified cache refresh time.

**Step 3** (Optional) **snmp-server cbqosmib cache service-policy count *count***

**Example:**

```
RP/0/RP0/CPU0:router(config)# snmp-server mibs cbqosmib cache
service-policy count 50
```

Enables QoS MIB caching with a limited number of service policies to cache.

**Step 4** **snmp-server ifindex persist**

**Example:**

```
RP/0/RP0/CPU0:router(config)# snmp-server ifindex persist
```

Enables if Index persistence globally on all Simple Network Management Protocol (SNMP) interfaces.

---

## Configuring LinkUp and LinkDown Traps for a Subset of Interfaces

By specifying a regular expression to represent the interfaces for which you are interested in setting traps, you can enable or disable linkUp and linkDown traps for a large number of interfaces simultaneously.

**Before you begin**

SNMP must be configured.

---

**Step 1** **configure**

**Example:**

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

**Step 2** **snmp-server interface subset** *subset-number* **regular-expression** *expression***Example:**

```
RP/0/RP0/CPU0:router(config)# snmp-server interface subset 10
  regular-expression "^Gig[a-zA-Z]+[0-9/]+\."
RP/0/RP0/CPU0:router(config-snmp-if-subset)#
```

Enters snmp-server interface mode for the interfaces identified by the regular expression.

The *subset-number* argument identifies the set of interfaces, and also assigns a priority to the subset in the event that an interface is included in more than one subset. Lower numbers have higher priority and their configuration takes precedent over interface subsets with higher numbers.

The *expression* argument must be entered surrounded by double quotes.

**Step 3** **notification linkupdown disable****Example:**

```
RP/0/RP0/CPU0:router(config-snmp-if-subset)# notification linkupdown disable
```

Disables linkUp and linkDown traps for all interfaces being configured. To enable previously disabled interfaces, use the **no** form of this command.

**Step 4** Use the **commit** or **end** command.

**commit** —Saves the configuration changes, and remains within the configuration session.

**end** —Prompts user to take one of these actions:

- **Yes** — Saves configuration changes and exits the configuration session.
- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration mode, without committing the configuration changes.

**Step 5** (Optional) **show snmp interface notification subset** *subset-number***Example:**

```
RP/0/RP0/CPU0:router# show snmp interface notification subset 10
```

Displays the linkUp and linkDown notification status for all interfaces identified by the subset priority.

**Step 6** (Optional) **show snmp interface notification regular-expression** *expression***Example:**

```
RP/0/RP0/CPU0:router# show snmp interface notification
  regular-expression "^Gig[a-zA-Z]+[0-9/]+\."
```

Displays the linkUp and linkDown notification status for all interfaces identified by the regular expression.

**Step 7** (Optional) **show snmp interface notification type** *interface-path-id*

**Example:**

```
RP/0/RP0/CPU0:router# show snmp interface notification
tengige 0/4/0/3.10
```

Displays the linkUp and linkDown notification status for the specified interface.

## Polling BRIDGE-MIB

BRIDGE-MIB defines the managed objects for MAC-bridges between LAN segments, based on the IEEE802.1d standard. This MIB also supports managing Transparent Bridges, which includes Control-Ethernet and VPLS bridges.

To poll this MIB, do one of the following:

- For SNMPv2: Use a community and map to the context with proper name
- For SNMPv3: Use a group attached to the context

To display the SNMP context mapping table, use the **show snmp context-mapping** command:

```
RP/0/RP0/CPU0:router# show snmp context-mapping
Context-name          Feature-name          Feature
ControlEthernet0_RP0_CPU0_S0  ControlEthernet0_RP0_CPU0_S0  BRIDGEINST
ControlEthernet0_RP1_CPU0_S0  ControlEthernet0_RP1_CPU0_S0  BRIDGEINST
```

```
RP/0/RP0/CPU0:router# show running-config snmp-server
snmp-server community cebridge1 RW SystemOwner
snmp-server context ControlEthernet0_RP0_CPU0_S0
snmp-server community-map cebridge1 context ControlEthernet0_RP0_CPU0_S0
```

In the above example, the community name is **cebridge1**, and the context name is **ControlEthernet0\_RP0\_CPU0\_S0**.

The format of the context name is as follows:

- Control-Ethernet bridges – **ControlEthernetrack\_slot\_module\_[S0|S1]**
- VPLS bridges – **vpls\_bridge\_domain\_name**

To configure the recipient of an SNMP notification operation, use the **snmp-server host** command:

```
RP/0/RSP0/CPU0:router(config)# snmp-server host 223.255.254.249 traps version 2c cebridge1
udp-port 1567
```

To enable BRIDGE-MIB trap notifications, use the **snmp-server traps bridgemib** command:

```
RP/0/RSP0/CPU0:router(config)# snmp-server traps bridgemib
```



## CHAPTER 5

# Configuring Periodic MIB Data Collection and Transfer

---

This document describes how to periodically transfer selected MIB data from your router to a specified Network Management System (NMS). The periodic MIB data collection and transfer feature is also known as bulk statistics.

- [Prerequisites for Periodic MIB Data Collection and Transfer, on page 45](#)
- [Information About Periodic MIB Data Collection and Transfer, on page 45](#)
- [How to Configure Periodic MIB Data Collection and Transfer, on page 47](#)
- [Periodic MIB Data Collection and Transfer: Example, on page 52](#)

## Prerequisites for Periodic MIB Data Collection and Transfer

To use periodic MIB data collection and transfer, you should be familiar with the Simple Network Management Protocol (SNMP) model of management information. You should also know what MIB information you want to monitor on your network devices, and the OIDs or object names for the MIB objects to be monitored.

## Information About Periodic MIB Data Collection and Transfer

### SNMP Objects and Instances

A type (or class) of SNMP management information is called an object. A specific instance from a type of management information is called an object instance (or SNMP variable). To configure a bulk statistics collection, you must specify the object types to be monitored using a bulk statistics object list and the specific instances of those objects to be collected using a bulk statistics schema.

MIBs, MIB tables, MIB objects, and object indices can all be specified using a series of numbers called an object identifier (OID). OIDs are used in configuring a bulk statistics collection in both the bulk statistics object lists (for general objects) and in the bulk statistics schemas (for specific object instances).

## Bulk Statistics Object Lists

To group the MIB objects to be polled, you need to create one or more object lists. A bulk statistics object list is a user-specified set of MIB objects that share the same MIB index. Object lists are identified using a name that you specify. Named bulk statistics object lists allow the same configuration to be reused in different bulk statistics schemas.

All the objects in an object list must share the same MIB index. However, the objects do not need to be in the same MIB and do not need to belong to the same MIB table. For example, it is possible to group `ifInOctets` and a `CISCO-IF-EXTENSION-MIB` object in the same schema, because the containing tables for both objects are indexed by the `ifIndex`.

## Bulk Statistics Schemas

Data selection for the Periodic MIB Data Collection and Transfer Mechanism requires the definition of a schema with the following information:

- Name of an object list.
- Instance (specific instance or series of instances defined using a wild card) that needs to be retrieved for objects in the specified object list.
- How often the specified instances need to be sampled (polling interval). The default polling interval is 5 minutes.

A bulk statistics schema is also identified using a name that you specify. This name is used when configuring the transfer options.

## Bulk Statistics Transfer Options

After configuring the data to be collected, a single virtual file (VFile or *bulk statistics file*) with all collected data is created. This file can be transferred to a network management station using FTP or TFTP. You can specify how often this file should be transferred. The default transfer interval is once every 30 minutes. You can also configure a secondary destination for the file to be used if, for whatever reason, the file cannot be transferred to the primary network management station.

The value of the transfer interval is also the collection period (collection interval) for the local bulk statistics file. After the collection period ends, the bulk statistics file is frozen, and a new local bulk statistics file is created for storing data. The frozen bulk statistics file is then transferred to the specified destination.

By default, the local bulk statistics file is deleted after successful transfer to an network management station.

## Benefits of Periodic MIB Data Collection and Transfer

Periodic MIB data collection and transfer (bulk statistics feature) allows many of the same functions as the bulk file MIB (`CISCO-BULK-FILE-MIB.my`), but offers some key advantages. The main advantage is that this feature can be configured through the CLI and does not require an external monitoring application.

Periodic MIB data collection and transfer is mainly targeted for medium to high-end platforms that have sufficient local storage (volatile or permanent) to store bulk statistics files. Locally storing bulk statistics files helps minimize loss of data during temporary network outages.



This feature also has more powerful data selection features than the bulk file MIB; it allows grouping of MIB objects from different tables into data groups (object lists). It also incorporates a more flexible instance selection mechanism, where the application is not restricted to fetching an entire MIB table.

# How to Configure Periodic MIB Data Collection and Transfer

## Configuring a Bulk Statistics Object List

The first step in configuring the Periodic MIB Data Collection and Transfer Mechanism is to configure one or more object lists.

---

### Step 1 **configure**

#### **Example:**

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

### Step 2 **snmp-server mib bulkstat object-list** *list-name*

#### **Example:**

```
snmp-server mib bulkstat object-list ifMib
```

Defines an SNMP bulk statistics object list and enters bulk statistics object list configuration mode.

### Step 3 **add** {oid | *object-name*}

#### **Example:**

```
RP/0/RP0/CPU0:router(config-bulk-objects)# add 1.3.6.1.2.1.2.2.1.11
RP/0/RP0/CPU0:router(config-bulk-objects)# add ifAdminStatus
RP/0/RP0/CPU0:router(config-bulk-objects)# add ifDescr
```

Adds a MIB object to the bulk statistics object list. Repeat as desired until all objects to be monitored in this list are added.

**Note** All the objects in a bulk statistics object list have to be indexed by the same MIB index. However, the objects in the object list do not need to belong to the same MIB or MIB table.

When specifying an object name instead of an OID (using the add command), only object names with mappings shown in the **show snmp mib object** command output can be used.

### Step 4 Use the **commit** or **end** command.

**commit** —Saves the configuration changes and remains within the configuration session.

**end** —Prompts user to take one of these actions:

- **Yes** — Saves configuration changes and exits the configuration session.
- **No** —Exits the configuration session without committing the configuration changes.

- **Cancel** —Remains in the configuration session, without committing the configuration changes.

## Configuring a Bulk Statistics Schema

The second step in configuring periodic MIB data collection and transfer is to configure one or more schemas.

### Before you begin

The bulk statistics object list to be used in the schema must be defined.

#### Step 1 **configure**

##### Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

#### Step 2 **snmp-server mib bulkstat schema *schema-name***

##### Example:

```
RP/0/RP0/CPU0:router(config)# snmp-server mib
bulkstat schema intE0
RP/0/RP0/CPU0:router(config-bulk-sc)#
```

Names the bulk statistics schema and enters bulk statistics schema mode.

#### Step 3 **object-list *list-name***

##### Example:

```
RP/0/RP0/CPU0:router(config-bulk-sc)# object-list
ifMib
```

Specifies the bulk statistics object list to be included in this schema. Specify only one object list per schema. If multiple object-list commands are executed, the earlier ones are overwritten by newer commands.

#### Step 4 Do one of the following:

- **instance exact** {**interface** *interface-id* [**sub-if**] | **oid** *oid*}
- **instance wild** {**interface** *interface-id* [**sub-if**] | **oid** *oid*}
- **instance range** **start** *oid* **end** *oid*
- **instance repetition** *oid* **max** *repeat-number*

##### Example:

```
RP/0/RP0/CPU0:router(config-bulk-sc)# instance
wild oid 1
```

or

```
RP/0/RP0/CPU0:router(config-bulk-sc)# instance
exact interface TenGigE 0/1.25
```

or

```
RP/0/RP0/CPU0:router(config-bulk-sc)# instance  
range start 1 end 2
```

or

```
RP/0/RP0/CPU0:router(config-bulk-sc)# instance  
repetition 1 max 4
```

Specifies the instance information for objects in this schema:

- The **instance exact** command indicates that the specified instance, when appended to the object list, represents the complete OID.
- The **instance wild** command indicates that all subindices of the specified OID belong to this schema. The wild keyword allows you to specify a partial, “wild carded” instance.
- The **instance range** command indicates a range of instances on which to collect data.
- The **instance repetition** command indicates data collection to repeat for a certain number of instances of a MIB object.

**Note** Only one **instance** command can be configured per schema. If multiple **instance** commands are executed, the earlier ones are overwritten by new commands.

#### Step 5 **poll-interval** *minutes*

**Example:**

```
RP/0/RP0/CPU0:router(config-bulk-sc)# poll-interval 10
```

Sets how often data should be collected from the object instances specified in this schema, in minutes. The default is once every 5 minutes. The valid range is from 1 to 20000.

#### Step 6 Use the **commit** or **end** command.

**commit** —Saves the configuration changes and remains within the configuration session.

**end** —Prompts user to take one of these actions:

- **Yes** — Saves configuration changes and exits the configuration session.
- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

---

## Configuring Bulk Statistics Transfer Options

The final step in configuring periodic MIB data collection and transfer is to configure the transfer options. The collected MIB data are kept in a local file-like entity called a VFile (virtual file, referred to as a bulk statistics file in this document). This file can be transferred to a remote network management station at intervals you specify.

### Before you begin

The bulk statistics object lists and bulk statistics schemas must be defined before configuring the bulk statistics transfer options.

**Step 1**     **configure****Example:**

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

**Step 2**     **snmp-server mib bulkstat transfer-id** *transfer-id***Example:**

```
RP/0/RP0/CPU0:router(config)# snmp-server mib
bulkstat transfer bulkstat1
```

Identifies the transfer configuration with a name (*transfer-id* argument) and enters bulk statistics transfer configuration mode.

**Step 3**     **buffer-size** *bytes***Example:**

```
RP/0/RP0/CPU0:router(config-bulk-tr)# buffersize 3072
```

(Optional) Specifies the maximum size for the bulk statistics data file, in bytes. The valid range is from 1024 to 2147483647 bytes. The default buffer size is 2048 bytes.

**Note** If the maximum buffer size for a bulk statistics file is reached before the transfer interval time expires, all additional data received is deleted. To correct this behavior, you can decrease the polling frequency, or increase the size of the bulk statistics buffer.

**Step 4**     **Example:**

(Optional) Specifies the format of the bulk statistics data file (VFile). The default is schemaASCII.

**Note** Transfers can only be performed using schemaASCII (cdcSchemaASCII) format. SchemaASCII is a human-readable format that contains parser-friendly hints for parsing data values.

**Step 5**     **schema** *schema-name***Example:**

```
RP/0/RP0/CPU0:router(config-bulk-tr)# schema TenGigE 0/5/0/11/1
RP/0/RP0/CPU0:router(config-bulk-tr)# schema TenGigE/0-CAR
RP/0/RP0/CPU0:router(config-bulk-tr)# schema TenGigE 0/5/0/11/1
```

Specifies the bulk statistics schema to be transferred. Repeat this command as desired. Multiple schemas can be associated with a single transfer configuration; all collected data are placed in a single bulk data file (VFile).

**Step 6**     **transfer-interval** *minutes***Example:**

```
RP/0/RP0/CPU0:router(config-bulk-tr)# transfer-interval 20
```

(Optional) Specifies how often the bulk statistics file are transferred, in minutes. The default value is once every 30 minutes. The transfer interval is the same as the collection interval.

**Step 7**     **url** *primary url*

**Example:**

```
RP/0/RP0/CPU0:router(config-bulk-tr)# url primary
ftp://user:password@host/folder/bulkstat1
```

Specifies the network management system (host) that the bulk statistics data file is transferred to, and the protocol to use for transfer. The destination is specified as a Uniform Resource Locator (URL). FTP or TFTP can be used for the bulk statistics file transfer.

**Step 8**     **url secondary *url*****Example:**

```
RP/0/RP0/CPU0:router(config-bulk-tr)# url secondary
tftp://10.1.0.1/tftpboot/user/bulkstat1
```

(Optional) Specifies a backup transfer destination and protocol for use in the event that transfer to the primary location fails. FTP or TFTP can be used for the bulk statistics file transfer.

**Step 9**     **retry *number*****Example:**

```
RP/0/RP0/CPU0:router(config-bulk-tr)# retry 1
```

(Optional) Specifies the number of transmission retries. The default value is 0 (in other words, no retries). If an attempt to send the bulk statistics file fails, the system can be configured to attempt to send the file again using this command.

One retry includes an attempt first to the primary destination then, if the transmission fails, to the secondary location. For example, if the retry value is 1, an attempt is made first to the primary URL, then to the secondary URL, then to the primary URL again, then to the secondary URL again. The valid range is from 0 to 100.

If all retries fail, the next normal transfer occurs after the configured transfer-interval time.

**Step 10**    **retain *minutes*****Example:**

```
RP/0/RP0/CPU0:router(config-bulk-tr)# retain 60
```

(Optional) Specifies how long the bulk statistics file should be kept in system memory, in minutes, after the completion of the collection interval and a transmission attempt is made. The default value is 0. Zero (0) indicates that the file is deleted immediately after the transfer is attempted. The valid range is from 0 to 20000.

**Note** If the retry command is used, you should configure a retain interval larger than 0. The interval between retries is the retain interval divided by the retry number. For example, if **retain 10** and **retry 2** are configured, two retries are attempted once every 5 minutes. Therefore, if retain 0 is configured, no retries are attempted.

**Step 11**    **enable****Example:**

```
RP/0/RP0/CPU0:router(config-bulk-tr)# enable
```

Begins the bulk statistics data collection and transfer process for this configuration.

- For successful execution of this action, at least one schema with non-zero number of objects must be configured.
- Periodic collection and file transfer begins only if this command is configured. Conversely, the **no enable** command stops the collection process. A subsequent **enable** starts the operations again.
- Each time the collection process is started using the **enable** command, data is collected into a new bulk statistics file. When the **no enable** command is used, the transfer process for any collected data immediately begins (in other words, the existing bulk statistics file is transferred to the specified management station).

**Step 12**    *commit minutes***Example:**

```
RP/0/RP0/CPU0:router(config-bulk-tr)# retain 60
```

If the maximum buffer size for a bulk statistics file is reached before the transfer interval time expires, the transfer operation is still initiated, but any bulk statistics data received after the file was full, and before it was transferred, are deleted. To correct this behavior, you can decrease the polling frequency, or increase the size of the bulk statistics buffer.

If **retain 0** is configured, no retries are attempted. This is because the interval between retries is the retain value divided by the retry value. For example, if **retain 10** and **retry 2** are configured, retries are attempted once every 5 minutes. Therefore, if you configure the retry command, you should also configure an appropriate value for the retain command.

## Periodic MIB Data Collection and Transfer: Example

This example shows how to configure periodic MIB data collection and transfer:

```
snmp-server mib bulkstat object-list cempo
add cempMemPoolName
add cempMemPoolType
!
snmp-server mib bulkstat schema cempWild
object-list cempo
instance wild oid 8695772
poll-interval 1
!
snmp-server mib bulkstat schema cempRepeat
object-list cempo
instance repetition 8695772.1 max 4294967295
poll-interval 1
!
snmp-server mib bulkstat transfer-id cempt1
enable
url primary tftp://223.255.254.254/auto/tftp-sjc-users3/username/dumpdcm
schema cempWild
schema cempRepeat
transfer-interval 2
!
```

This example shows sample bulk statistics file content:

```
Schema-def cempt1.cempWild "%u, %s, %s, %d" Epochtime instanceoid
1.3.6.1.4.1.9.9.221.1.1.1.1.3 1.3.6.1.4.1.9.9.221.1.1.1.1.2
cempt1.cempWild: 1339491515, 8695772.1, processor, 2
cempt1.cempWild: 1339491515, 8695772.2, reserved, 11
cempt1.cempWild: 1339491515, 8695772.3, image, 12
cempt1.cempWild: 1339491575, 8695772.1, processor, 2
cempt1.cempWild: 1339491575, 8695772.2, reserved, 11
cempt1.cempWild: 1339491575, 8695772.3, image, 12
Schema-def cempt1.cempRepeat "%u, %s, %s, %d" Epochtime instanceoid
1.3.6.1.4.1.9.9.221.1.1.1.1.3 1.3.6.1.4.1.9.9.221.1.1.1.1.2
cempt1.cempRepeat: 1339491515, 8695772.1, processor, 2
cempt1.cempRepeat: 1339491515, 8695772.2, reserved, 11
cempt1.cempRepeat: 1339491515, 8695772.3, image, 12
```

```
cempt1.cempRepeat: 1339491515, 26932192.1, processor, 2
cempt1.cempRepeat: 1339491515, 26932192.2, reserved, 11
cempt1.cempRepeat: 1339491515, 26932192.3, image, 12
cempt1.cempRepeat: 1339491515, 35271015.1, processor, 2
cempt1.cempRepeat: 1339491515, 35271015.2, reserved, 11
cempt1.cempRepeat: 1339491515, 35271015.3, image, 12
cempt1.cempRepeat: 1339491515, 36631989.1, processor, 2
cempt1.cempRepeat: 1339491515, 36631989.2, reserved, 11
cempt1.cempRepeat: 1339491515, 36631989.3, image, 12
cempt1.cempRepeat: 1339491515, 52690955.1, processor, 2
cempt1.cempRepeat: 1339491515, 52690955.2, reserved, 11
cempt1.cempRepeat: 1339491515, 52690955.3, image, 12
```







## CHAPTER 6

# Configuring Cisco Discovery Protocol

*Cisco Discovery Protocol* (CDP) is a media- and protocol-independent protocol that runs on all Cisco-manufactured equipment including routers, bridges, access and communication servers, and switches. Using CDP, you can view information about all the Cisco devices that are directly attached to the device.

- [Prerequisites for Implementing CDP, on page 55](#)
- [Information About Implementing CDP, on page 55](#)
- [Enabling CDP, on page 57](#)
- [Modifying CDP Default Settings, on page 57](#)
- [Monitoring CDP, on page 58](#)

## Prerequisites for Implementing CDP

To enable CDP, you must install the CDP package on your router.

You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

## Information About Implementing CDP

CDP is primarily used to obtain protocol addresses of neighboring devices and discover the platform of those devices. CDP can also be used to display information about the interfaces your router uses. CDP is media- and protocol-independent, and runs on all equipment manufactured by Cisco, including routers, bridges, access servers, and switches.

Use of SNMP with the CDP MIB allows network management applications to learn the device type and the SNMP agent address of neighboring devices and to send SNMP queries to those devices. CDP uses the CISCO-CDP-MIB.

CDP runs on all media that support Subnetwork Access Protocol (SNAP), including LAN, Frame Relay, and ATM physical media. CDP runs over the data link layer only. Therefore, two systems that support different network-layer protocols can learn about each other.

Each device configured for CDP sends periodic messages, known as *advertisements*, to a multicast address. Each device advertises at least one address at which it can receive SNMP messages. The advertisements also contain time-to-live, or hold-time, information, which indicates the length of time a receiving device holds

CDP information before discarding it. Each device also listens to the periodic CDP messages sent by others to learn about neighboring devices and determine when their interfaces to the media go up or down.

CDP Version-2 (CDPv2) is the most recent release of the protocol and provides more intelligent device tracking features. These features include a reporting mechanism that allows for more rapid error tracking, thereby reducing costly downtime. Reported error messages can be sent to the console or to a logging server, and can cover instances of unmatching native VLAN IDs (IEEE 802.1Q) on connecting ports, and unmatching port duplex states between connecting devices.

CDPv2 **show** commands can provide detailed output on VLAN Trunking Protocol (VTP) management domain and duplex modes of neighbor devices, CDP-related counters, and VLAN IDs of connecting ports.

Type-length-value fields (TLVs) are blocks of information embedded in CDP advertisements. This table summarizes the TLV definitions for CDP advertisements.

**Table 6: Type-Length-Value Definitions for CDPv2**

TLV	Definition
Device-ID TLV	Identifies the device name in the form of a character string.
Address TLV	Contains a list of network addresses of both receiving and sending devices.
Port-ID TLV	Identifies the port on which the CDP packet is sent.
Capabilities TLV	Describes the functional capability for the device in the form of a device type; for example, a switch.
Version TLV	Contains information about the software release version on which the device is running.
Platform TLV	Describes the hardware platform name of the device, for example, Cisco 4500.
VTP Management Domain TLV	Advertises the system's configured VTP management domain name-string. Used by network operators to verify VTP domain configuration in adjacent network nodes.
Native VLAN TLV	Indicates, per interface, the assumed VLAN for untagged packets on the interface. CDP learns the native VLAN for an interface. This feature is implemented only for interfaces that support the IEEE 802.1Q protocol.
Full/Half Duplex TLV	Indicates status (duplex configuration) of CDP broadcast interface. Used by network operators to diagnose connectivity problems between adjacent network elements.

# How to Implement CDP on Cisco IOS XR Software

## Enabling CDP

To enable CDP, you must first enable CDP globally on the router and then enable CDP on a per-interface basis. This example explains how to enable CDP globally on the router and then enable CDP on an interface.

```
Router:# configure
Router(config):# cdp
Router(config):# commit

Router:# configure
Router(config):# interface hundredGigE 0/0/0/4
Router(config-if):# cdp
Router(config-if):# commit
```

## Modifying CDP Default Settings

This task explains how to modify the default version, hold-time setting, and timer settings.



---

**Note** The commands can be entered in any order.

---

---

### Step 1 **configure**

#### Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

### Step 2 **cdp advertise v1**

#### Example:

```
RP/0/RP0/CPU0:router(config)# cdp advertise v1
```

Configures CDP to use only version 1 (CDPv1) in communicating with neighboring devices.

- By default, when CDP is enabled, the router sends CDPv2 packets. CDP also sends and receives CDPv1 packets if the device with which CDP is interacting does not process CDPv2 packets.
- In this example, the router is configured to send and receive only CDPv1 packets.

### Step 3 **cdp holdtime *seconds***

#### Example:

```
RP/0/RP0/CPU0:router(config)# cdp holdtime 30
```

Specifies the amount of time that the receiving networking device will hold a CDP packet sent from the router before discarding it.

- By default, when CDP is enabled, the receiving networking device holds a CDP packet for 180 seconds before discarding it.

**Note** The CDP hold time must be set to a higher number of seconds than the time between CDP transmissions, which is set with the **cdp timer** command.

- In this example, the value of hold-time for the *seconds* argument is set to 30.

#### Step 4 **cdp timer** *seconds*

##### Example:

```
RP/0/RP0/CPU0:router(config)# cdp timer 20
```

Specifies the frequency at which CDP update packets are sent.

- By default, when CDP is enabled, CDP update packets are sent at a frequency of once every 60 seconds.

**Note** A lower timer setting causes CDP updates to be sent more frequently.

- In this example, CDP update packets are configured to be sent at a frequency of once every 20 seconds.

#### Step 5 Use the **commit** or **end** command.

**commit** —Saves the configuration changes and remains within the configuration session.

**end** —Prompts user to take one of these actions:

- **Yes** — Saves configuration changes and exits the configuration session.
- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

#### Step 6 (Optional) **show cdp**

##### Example:

```
RP/0/RP0/CPU0:router# show cdp
```

Displays global CDP information.

The output displays the CDP version running on the router, the hold time setting, and the timer setting.

## Monitoring CDP

This task shows how to monitor CDP.



**Note** The commands can be entered in any order.

**Step 1** `show cdp entry` *{\* | entry-name}* [**protocol** | **version**]

**Example:**

```
RP/0/RP0/CPU0:router# show cdp entry *
```

Displays information about a specific neighboring device or all neighboring devices discovered using CDP.

**Step 2** `show cdp interface` [*type interface-path-id* | **location node-id**]

**Example:**

```
RP/0/RP0/CPU0:router# show cdp interface pos 0/0/0/1
```

Displays information about the interfaces on which CDP is enabled.

**Step 3** `show cdp neighbors` [*type interface-path-id* | **location node-id**] [**detail**]

**Example:**

```
RP/0/RP0/CPU0:router# show cdp neighbors
```

Displays detailed information about neighboring devices discovered using CDP.

**Step 4** `show cdp traffic` [**location node-id**]

**Example:**

```
RP/0/RP0/CPU0:router# show cdp traffic
```

Displays information about the traffic gathered between devices using CDP.

## Examples

The following is sample output for the `show cdp neighbors` command:

```
RP/0/RP0/CPU0:router# show cdp neighbors

Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater

Device ID      Local Intrfce   Holdtme  Capability  Platform  Port ID
NCS5500        Hu0/0/0/4      15       R           NCS-5500  Hu0/0/0/4
```

The following is sample output for the `show cdp neighbors` command. In this example, the optional *type instance* arguments are used in conjunction with the **detail** optional keyword to display detailed information about a CDP neighbor. The output includes information on both IPv4 and IPv6 addresses.

```
RP/0/RP0/CPU0:router# show cdp neighbors hundredGigE 0/0/0/4 detail

-----
Device ID: NCS5500
SysName  : NCS5500
Entry address(es):
```

```

    IPv4 address: 40.0.0.2
    IPv6 address: 10:10:10:10::1
Platform: cisco NCS-5500, Capabilities: Router
Interface: HundredGigE0/0/0/4
Port ID (outgoing port): HundredGigE0/0/0/4
Holdtime : 13 sec

Version :
7.1.1.112I

advertisement version: 2
Duplex: full

```

The following is sample output for the **show cdp entry** command. In this example, the optional *entry* argument is used to display entry information related to a specific CDP neighbor.

```

RP/0/RP0/CPU0:router# show cdp entry NCS5500
-----
Device ID: NCS5500
SysName : NCS5500
Entry address(es):
  IPv4 address: 40.0.0.2
  IPv6 address: 10:10:10:10::1
Platform: cisco NCS-5500, Capabilities: Router
Interface: HundredGigE0/0/0/4
Port ID (outgoing port): HundredGigE0/0/0/4
Holdtime : 11 sec

Version :
7.1.1.112I

advertisement version: 2
Duplex: full

```

The following is sample output for the **show cdp interface** command. In this example, CDP information related to interface 0/0/0/4 is displayed.

```

RP/0/RP0/CPU0:router# show cdp interface hundredGigE 0/0/0/4

HundredGigE0/0/0/4 is Up
Encapsulation ether
Sending CDP packets every 60 seconds
Holdtime is 180 seconds

```

The following is sample output for the **show cdp traffic** command:

```

RP/0/RP0/CPU0:router# show cdp traffic

CDP counters :
  Packets output: 10, Input: 39
  Hdr syntax: 0, Chksum error: 0, Encaps failed: 0
  No memory: 0, Invalid packet: 0, Truncated: 0
  CDP version 1 advertisements output: 0, Input: 0
  CDP version 2 advertisements output: 10, Input: 39
  Unrecognize Hdr version: 0, File open failed: 0

```



## CHAPTER 7

# Configuring Call Home

This module describes the configuring of the Call Home feature.

**Table 7: Feature History for Configuring Call Home**

Release	Modification
Release 7.0.11	Call Home was introduced

This model contains the following topics:

- [About Call Home, on page 61](#)
- [Benefits of Using Call Home, on page 62](#)
- [Prerequisites for Call Home, on page 62](#)
- [How to Configure Call Home, on page 63](#)
- [Configuring Contact Information, on page 63](#)
- [Destination Profiles, on page 65](#)
- [Call Home Alert Groups, on page 67](#)
- [Configuring Email, on page 71](#)
- [Configuring a HTTPS Proxy Server , on page 72](#)
- [Sending Call-home Data through an Email, on page 73](#)
- [Sending Call-home Data through HTTPS, on page 75](#)
- [Configuring Call Home to use VRF, on page 76](#)
- [Configuring Call Home Data Privacy, on page 77](#)
- [Sending Smart License Data , on page 78](#)

## About Call Home

Call Home provides an email and HTTPS based notification for critical system policies. A range of message formats are available for compatibility with pager services or XML-based automated parsing applications. You can use this feature to page a network support engineer, or email a Network Operations Center. You can also use Cisco Smart Call Home services to generate a case with the Technical Assistance Center. The Call Home feature can deliver alert messages containing information about diagnostics and environmental faults and events.

The Call Home feature can deliver alerts to multiple recipients, referred to as Call Home destination profiles. Each profile includes configurable message formats and content categories. A predefined destination is

provided for sending alerts to the Cisco TAC, however you also can define your own destination profiles. When you configure Call Home to send messages, the appropriate CLI show command is executed and the command output is attached to the message. Call Home messages are delivered in the following formats:

- Short text format which provides a one or two line description of the fault that is suitable for pagers or printed reports.
- Full text format which provides fully formatted message with detailed information that is suitable for human reading.
- XML machine-readable format that uses Extensible Markup Language (XML) and Adaptive Messaging Language (AML) XML schema definition (XSD). The AML XSD is published on the Cisco.com website at <http://www.cisco.com>. The XML format enables communication with the Cisco Systems Technical Assistance Center.

The Call Home feature is enabled by default. The Cisco TAC-1 profile is created after the device starts. The default Call Home settings that includes destination address, transport methods, alert-group subscriptions, and more are saved in the CiscoTAC-1 profile. To check the default settings, use the **show call-home profile CiscoTAC-1** command.

## Benefits of Using Call Home

The Call Home feature offers the following benefits:

- Multiple message-format options:
  - Short Text—Suitable for pagers or printed reports.
  - Plain Text—Full formatted message information suitable for human reading.
  - XML—Matching readable format using Extensible Markup Language (XML) and Adaptive Markup Language (AML) document type definitions (DTDs). The XML format enables communication with the Cisco Smart Call Home server.
- Multiple concurrent message destinations.
- Multiple message categories, including configuration, environmental conditions, inventory, syslog, and crash events.
- Filtering of messages by severity and pattern matching.
- Scheduling of periodic message sending.

## Prerequisites for Call Home

How you configure Call Home depends on how you intend to use the feature. Consider the following requirements before you configure Call Home:

- Obtain e-mail, phone, and street address information for the Call Home contact to be configured so that the receiver can determine the origin of messages received.
- Identify the name or IPv4 address of a primary Simple Mail Transfer Protocol (SMTP) server and any backup servers, if using e-mail message delivery.



- Verify IP connectivity from the router to the e-mail server(s) or the destination HTTP server.
- If Cisco Smart Call Home is used, an active service contract covering the device is required to provide full SCH service.

## How to Configure Call Home

To configure the sending of Call Home messages, do the following:

1. Assign contact information.
2. Configure and enable one or more destination profiles.
3. Associate one or more alert groups to each profile.
4. Configure the email server options, if using e-mail message delivery.
5. Enable Call Home.

The above tasks are described in detail in the below procedures.



**Note** Before enabling Call-Home, you must configure the source interface for HTTPS over IPv6. However, for HTTPS over IPv4, Call-Home works without the source interface.

In case of a dual-stack call-home configuration on the device, the IPv4 address is preferred over the IPv6 address. This may result in IPv6 resolution failure. Due to this limitation, the IPv6 device registration with the licensing server may only be done with a single mode, that is, IPv6 only configuration.

Use the **http client source-interface ipv6** command to configure the source interface.

## Configuring Contact Information

Each router must include a contact e-mail address. You can optionally include other identifying information for your system installation.

**Step 1**     **configure**

**Example:**

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

**Step 2**     **call-home**

**Example:**

```
RP/0/RP0/CPU0:router(config)# call-home
RP/0/RP0/CPU0:router(config-call-home)#
```

Enters call home configuration mode.

**Step 3** **contact-email-addr** *email-address*

**Example:**

```
RP/0/RP0/CPU0:router (config-call-home) # contact-email-addr
user1@cisco.com
```

Configures the customer email address. Enter up to 200 characters in email address format with no spaces.

**Step 4** (Optional) **contract-id** *contract-id-string*

**Example:**

```
RP/0/RP0/CPU0:router (config-call-home) # contract-id
Contract-identifier
```

Configures the contract ID. Enter up to 64 characters. If you include spaces, you must enclose the entry in quotes ("").

**Step 5** (Optional) **customer-id** *customer-id-string*

**Example:**

```
RP/0/RP0/CPU0:router (config-call-home) # customer-id Customer1
```

Configures the customer ID. Enter up to 64 characters. If you include spaces, you must enclose the entry in quotes ("").

**Step 6** (Optional) **phone-number** *phone-number-string*

**Example:**

```
RP/0/RP0/CPU0:router (config-call-home) # phone-number +405-123-4567
```

Configures the customer phone number. The number must begin with a plus (+) prefix, and may contain only dashes (-) and numbers. Enter up to 16 characters.

**Step 7** (Optional) **street-address** *street-address*

**Example:**

```
RP/0/RP0/CPU0:router (config-call-home) # street-address "300 E. Tasman Dr.
San Jose, CA 95134"
```

Configures the customer street address where RMA equipment can be shipped. Enter up to 200 characters. If you include spaces, you must enclose the entry in quotes ("").

**Step 8** (Optional) **site-id** *site-id-string*

**Example:**

```
RP/0/RP0/CPU0:router (config-call-home) # site-id SJ-RouterRoom1
```

Configures the site ID for the system. Enter up to 200 characters. If you include spaces, you must enclose the entry in quotes ("").

**Step 9** Use the **commit** or **end** command.

**commit** —Saves the configuration changes and remains within the configuration session.

**end** —Prompts user to take one of these actions:

- **Yes** — Saves configuration changes and exits the configuration session.
- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

**Step 10**    **show call-home****Example:**

```
RP/0/RP0/CPU0:router# show call-home
```

Displays information about the system contacts.

---

## Destination Profiles

A destination profile includes the following information:

- One or more alert groups—The group of alerts that trigger a specific Call Home message if the alert occurs.
- One or more e-mail or HTTPS destinations—The list of recipients for the Call Home messages generated by alert groups assigned to this destination profile.
- Message format—The format for the Call Home message (short text, full text, or XML).
- Message severity level—The Call Home severity level that the alert must meet before a Call Home message is sent to all e-mail and HTTPS URL addresses in the destination profile. An alert is not generated if the Call Home severity level of the alert is lower than the message severity level set for the destination profile. The inventory and configuration alert groups do not have concept of severity level. They are generated directly.

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.

The following predefined destination profiles are supported:

- CiscoTAC-1—Supports the Cisco-TAC alert group in XML message format.

## Configuring and Activating Destination Profiles

You must have at least one activated destination profile for Call Home messages to be sent. The CiscoTAC-1 profile exists by default but is not active.

---

**Step 1**    **configure****Example:**

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

**Step 2**      **call-home****Example:**

```
RP/0/RP0/CPU0:router(config)# call-home
RP/0/RP0/CPU0:router(config-call-home)#
```

Enters call home configuration mode.

**Step 3**      **profile** *profile-name***Example:**

```
RP/0/RP0/CPU0:router(config-call-home)# profile my_profile
RP/0/RP0/CPU0:router(config-call-home-profile)#
```

Enters call home profile configuration mode to configure a new or existing profile.

**Step 4**      **destination address email** *email-address***Example:**

```
RP/0/RP0/CPU0:router(config-call-home-profile)# destination
address email support_me@cisco.com
```

Configures an email address to which Call Home messages are sent for this profile.

**Step 5**      **destination message-size-limit** *max-size***Example:**

```
RP/0/RP0/CPU0:router(config-call-home-profile)# destination
message-size-limit 1000
```

Configures the maximum size of Call Home messages for this profile. Values can be between 50 and 3145728 characters.

**Step 6**      **destination preferred-msg-format** {**short-text** | **long-text** | **xml**}**Example:**

```
RP/0/RP0/CPU0:router(config-call-home-profile)# destination
preferred-msg-format xml
```

Configures the message format for this profile. The default is xml.

**Step 7**      **destination transport-method** [ **email** | **hhttp** ]**Example:**

```
RP/0/RP0/CPU0:router(config-call-home-profile)# destination
transport-method email
```

Configures the transport method for this profile.

**Step 8**      **active****Example:**

```
RP/0/RP0/CPU0:router(config-call-home-profile)# active
```

Activates the destination profile.

**Note** At least one destination profile must be active for Call Home messages to be sent.

**Step 9** Use the **commit** or **end** command.

**commit** —Saves the configuration changes and remains within the configuration session.

**end** —Prompts user to take one of these actions:

- **Yes** — Saves configuration changes and exits the configuration session.
- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

**Step 10** **show call-home profile** {all | *profile-name*}

**Example:**

```
RP/0/RP0/CPU0:router# show call-home profile all
```

Displays information about the destination profile.

## Call Home Alert Groups

An alert group is a predefined subset of alerts or events that Call Home detects and reports to one or more destinations. Alert groups allow you to select the set of alerts that you want to send to a predefined or custom destination profile. Alerts are sent to e-mail destinations in a destination profile only if that alert belongs to one of the alert groups associated with that destination profile and if the alert has a Call Home message severity at or above the message severity set in the destination profile.

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

**Table 8: Alert Groups and Executed Commands**

Alert Group	Description	Executed Commands
Environmental	Events related to power, fan, and environment-sensing elements such as temperature alarms.	<ul style="list-style-type: none"> <li>• <b>show environment</b></li> <li>• <b>show logging</b></li> <li>• <b>show inventory</b></li> <li>• <b>show environment trace</b></li> <li>• <b>show diag</b></li> </ul>
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.	<ul style="list-style-type: none"> <li>• <b>show platform</b></li> <li>• <b>show version</b></li> <li>• <b>show diag</b></li> <li>• <b>show inventory oid</b></li> </ul>

Alert Group	Description	Executed Commands
Syslog	Events generated by specific interesting syslog messages	<ul style="list-style-type: none"> <li>• <b>show version</b></li> <li>• <b>show logging</b></li> <li>• <b>show inventory</b></li> </ul>
Configuration	User-generated request for configuration or configuration change event.	<ul style="list-style-type: none"> <li>• <b>show version</b></li> <li>• <b>show running config all</b></li> <li>• <b>show inventory</b></li> <li>• <b>show configuration history last 30</b></li> <li>• <b>show configuration commit changes last 1</b></li> </ul>
Snapshot	This alert group can be configured for periodic notifications	By default, this alert group has no commands to be run. You can add the required commands that need to be run.

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

## Call Home Message Levels

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 Call Home message level threshold. The Call Home message level ranges from 0 (lowest level of urgency) to 9 (highest level of urgency). Call Home messages are generated if they have a severity level equal to or greater than the Call Home message level threshold for the destination profile.

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



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

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

**Table 9: Severity and syslog Level Mapping**

Call Home Level	Keyword	syslog Level	Description
9	Catastrophic	Not-Applicable	Network-wide catastrophic failure.
8	Disaster	Not-Applicable	Significant network impact.

Call Home Level	Keyword	syslog Level	Description
7	Fatal	Emergency (0)	System is unusable.
6	Critical	Alert (1)	Critical conditions that indicate that immediate attention is needed.
5	Major	Critical (2)	Major conditions.
4	Minor	Error (3)	Minor conditions.
3	Warning	Warning (4)	Warning conditions.
2	Notification	Notice (5)	Basic notification and informational messages. Possibly independently insignificant.
1	Normal	Information (6)	Normal event signifying return to normal state.
	Debugging	Debug (7)	Debugging messages.

## Associating an Alert Group with a Destination Profile

An alert is sent only to destination profiles that have subscribed to the Call Home alert group.

### Before you begin

Use the **show call-home alert-group** command to view available alert groups.

#### Step 1 **configure**

##### Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

#### Step 2 **call-home**

##### Example:

```
RP/0/RP0/CPU0:router(config)# call-home
RP/0/RP0/CPU0:router(config-call-home)#
```

Enters call home configuration mode.

#### Step 3 **profile profile-name**

##### Example:

```
RP/0/RP0/CPU0:router(config-call-home)# profile my_profile
RP/0/RP0/CPU0:router(config-call-home-profile)#
```

Enters call home profile configuration mode to configure a new or existing profile.

**Step 4** `subscribe-to-alert-group inventory [periodic {daily | monthly day-of-month | weekly day-of-week} hh:mm`

**Example:**

```
RP/0/RP0/CPU0:router(config-call-home-profile)# subscribe-to-alert-group
inventory periodic monthly 1 10:00
```

Configures a destination profile to receive messages for the inventory alert group. Either alerts are sent periodically, or any non-normal event triggers an alert.

**Step 5** `subscribe-to-alert-group syslog severity severity-level pattern string`

**Example:**

```
RP/0/RP0/CPU0:router(config-call-home-profile)# subscribe-to-alert-group
syslog severity major pattern
```

Configures a destination profile to receive messages for the syslog alert group. Alerts with a severity the same or greater than the specified severity level are sent.

- **catastrophic**—Includes network-wide catastrophic events in the alert. This is the highest severity.
- **critical**—Includes events requiring immediate attention (system log level 1).
- **disaster**—Includes events with significant network impact.
- **fatal**—Includes events where the system is unusable (system log level 0).
- **major**—Includes events classified as major conditions (system log level 2).
- **minor**—Includes events classified as minor conditions (system log level 3)
- **normal**—Specifies the normal state and includes events classified as informational (system log level 6). This is the default.
- **notification**—Includes events informational message events (system log level 5).
- **warning**—Includes events classified as warning conditions (system log level 4).

You can specify a pattern to be matched in the syslog message. If the pattern contains spaces, you must enclose it in quotes ("").

**Step 6** `subscribe-to-alert-group snapshot severity severity-level pattern string`

**Example:**

```
RP/0/RP0/CPU0:router(config-call-home-profile)# subscribe-to-alert-group
snapshot severity major pattern
```

Configures a destination profile to receive messages for the snapshot alert group. Alerts with a severity the same or greater than the specified severity level are sent.

You can specify a pattern to be matched in the syslog message. If the pattern contains spaces, you must enclose it in quotes ("").

**Step 7** `subscribe-to-alert-group configuration severity severity-level pattern string`

**Example:**



```
RP/0/RP0/CPU0:router(config-call-home-profile)# subscribe-to-alert-group configuration severity major
pattern
```

Configures a destination profile to receive messages for the configuration alert group. Alerts with a severity the same or greater than the specified severity level are sent.

You can specify a pattern to be matched in the syslog message. If the pattern contains spaces, you must enclose it in quotes ("").

**Step 8** Use the **commit** or **end** command.

**commit** — Saves the configuration changes and remains within the configuration session.

**end** — Prompts user to take one of these actions:

- **Yes** — Saves configuration changes and exits the configuration session.
- **No** — Exits the configuration session without committing the configuration changes.
- **Cancel** — Remains in the configuration session, without committing the configuration changes.

---

#### What to do next

Use the **show call-home profile** command to view the profile configurations.

## Configuring Email

If Call Home messages are sent via email, the you must configure your email server before Call Home messages can be sent.

**Step 1** **configure**

**Example:**

```
RP/0/RP0/CPU0:router# configure
Enters mode.
```

**Step 2** **call-home**

**Example:**

```
RP/0/RP0/CPU0:router(config)# call-home
RP/0/RP0/CPU0:router(config-call-home)#
Enters call home configuration mode.
```

**Step 3** (Optional) **sender from** *email-address*

**Example:**

```
RP/0/RP0/CPU0:router(config-call-home)# sender from
my_email@cisco.com
```

Specifies the email message “from” address.

**Step 4** (Optional) **sender reply-to** *email-address*

**Example:**

```
RP/0/RP0/CPU0:router(config-call-home)# sender reply-to  
my_email@cisco.com
```

Specifies the email message “reply-to” address.

**Step 5** Required: **mail-server address priority** *priority*

**Example:**

```
RP/0/RP0/CPU0:router(config-call-home)# mail-server  
198.61.170.16 priority 1
```

Specifies the mail server to use to send Call Home messages. You can specify an IP address or mail server name. You can specify up to five mail servers to use. The server with the lower priority is tried first.

**Step 6** Use the **commit** or **end** command.

**commit** —Saves the configuration changes and remains within the configuration session.

**end** —Prompts user to take one of these actions:

- **Yes** — Saves configuration changes and exits the configuration session.
- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

**Step 7** **show call-home mail-server status**

**Example:**

```
RP/0/RP0/CPU0:router# show call-home mail-server status
```

Displays the status of the specified mail server.

---

## Configuring a HTTPS Proxy Server

This task enables the user to configure a HTTPS Proxy Server.

---

**Step 1** **configure**

**Example:**

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

**Step 2** **call-home**

**Example:**

```
RP/0/RP0/CPU0:router (config) # call-home
```

Enters Call Home configuration mode.

**Step 3** **http-proxy** *proxy-server-name* **port** *port-number***Example:**

```
RP/0/RP0/CPU0:router (config) # http-proxy pl port 100
```

Configures the port for the specified HTTPS proxy server. Range is 1 to 65535.

## Sending Call-home Data through an Email

This task enables the user to configure sending Call-home data using email as the transport method:

**Step 1** **configure****Example:**

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

**Step 2** **call-home****Example:**

```
RP/0/RP0/CPU0:router (config) # call-home
```

Enters Call Home configuration mode.

**Step 3** **profile** *name***Example:**

```
RP/0/RP0/CPU0:router (config-call-home) # profile user1
```

Enters call home destination profile configuration mode for the specified destination profile name. If the specified destination profile does not exist, it is created.

**Step 4** **active****Example:**

```
RP/0/RP0/CPU0:router (config-call-home-profile) # active
```

Enables the destination profile. By default, a user-defined profile is enabled when it is created.

**Step 5** **destination transport-method** **email****Example:**

```
RP/0/RP0/CPU0:router (config-call-home-profile) # destination transport-method email
```

Configures the message transport method for email. This is the default

**Step 6** **destination address** **email** *email-address*

**Example:**

```
RP/0/RP0/CPU0:router (config-call-home-profile) # destination address email xyz@cisco.com
```

Configures the destination e-mail address to which Call Home messages are sent.

**Step 7 destination-preferred-msg-format {long-text |short-text| xml}****Example:**

```
RP/0/RP0/CPU0:router (config-call-home-profile) # destination-preferred-msg-format xml
```

(Optional) Configures a preferred message format. The default is XML.

**Step 8 subscribe-to-alert-group syslog severity severity-level pattern string****Example:**

```
RP/0/RP0/CPU0:router (config-call-home-profile) # subscribe-to-alert-group syslog severity normal
pattern COUNT
```

Configures a destination profile to receive messages for the syslog alert group. Alerts with a severity the same or greater than the specified severity level are sent.

- **critical**—Includes events requiring immediate attention (system log level 1).
- **disaster**—Includes events with significant network impact.
- **fatal**—Includes events where the system is unusable (system log level 0).
- **major**—Includes events classified as major conditions (system log level 2).
- **minor**—Includes events classified as minor conditions (system log level 3).
- **normal**—Specifies the normal state and includes events classified as informational (system log level 6).  
This is the default.
- **notification**—Includes events informational message events (system log level 5).
- **warning**—Includes events classified as warning conditions (system log level 4).

You can specify a pattern to be matched in the syslog message. If the pattern contains spaces, you must enclose it in quotes ("").

**Step 9** Use the **commit** or **end** command.

**commit** —Saves the configuration changes and remains within the configuration session.

**end** —Prompts user to take one of these actions:

- **Yes** — Saves configuration changes and exits the configuration session.
- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

# Sending Call-home Data through HTTPS

This task enables the user to configure sending Call-home data using HTTPS as the transport method:



**Note** For the HTTPS function to work you should use the **crypto ca trustpoint** command to declare a CA, followed by the **crl option** command. This ensures that the certificates of other peers are accepted without trying to obtain the appropriate CRL. For example:

```
RP/0/RP0/CPU0:ios(config)#crypto ca trustpoint Trustpool
RP/0/RP0/CPU0:ios(config-trustp)#crl optional
```

## Step 1 configure

### Example:

```
RP/0/RP0/CPU0:router# configure
Enters mode.
```

## Step 2 call-home

### Example:

```
RP/0/RP0/CPU0:router (config) # call-home
Enters Call Home configuration mode.
```

## Step 3 profile *name*

### Example:

```
RP/0/RP0/CPU0:router (config-call-home) # profile user1
```

Enters call home destination profile configuration mode for the specified destination profile name. If the specified destination profile does not exist, it is created.

## Step 4 active

### Example:

```
RP/0/RP0/CPU0:router (config-call-home-profile) # active
```

Enables the destination profile. By default, a user-defined profile is enabled when it is created.

## Step 5 destination transport-method http

### Example:

```
RP/0/RP0/CPU0:router (config-call-home-profile) # destination transport-method http
```

Configures the message transport method for HTTPS.

## Step 6 destination address http *url*

### Example:

```
RP/0/RP0/CPU0:router (config-call-home-profile) # destination address http https://example.com
```

Configures the destination URL address to which Call Home messages are sent.

**Step 7** `destination preferred-msg-format {long-text |short-text| xml}`

**Example:**

```
RP/0/RP0/CPU0:router (config-call-home-profile) # destinationpreferred-msg-format xml
```

(Optional) Configures a preferred message format. The default is XML.

**Step 8** `subscribe-to-alert-group syslog severity severity-level pattern string`

**Example:**

```
RP/0/RP0/CPU0:router (config-call-home-profile) # subscribe-to-alert-group syslog severity normal
pattern COUNT
```

Configures a destination profile to receive messages for the syslog alert group. Alerts with a severity the same or greater than the specified severity level are sent.

- **critical**—Includes events requiring immediate attention (system log level 1).
- **disaster**—Includes events with significant network impact.
- **fatal**—Includes events where the system is unusable (system log level 0).
- **major**—Includes events classified as major conditions (system log level 2).
- **minor**—Includes events classified as minor conditions (system log level 3).
- **normal**—Specifies the normal state and includes events classified as informational (system log level 6).  
This is the default.
- **notification**—Includes events informational message events (system log level 5).
- **warning**—Includes events classified as warning conditions (system log level 4).

You can specify a pattern to be matched in the syslog message. If the pattern contains spaces, you must enclose it in quotes ("").

**Step 9** Use the `commit` or `end` command.

**commit** —Saves the configuration changes and remains within the configuration session.

**end** —Prompts user to take one of these actions:

- **Yes** — Saves configuration changes and exits the configuration session.
- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

---

## Configuring Call Home to use VRF

---

**Step 1** `configure`

**Example:**

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

**Step 2** **call-home**

**Example:**

```
RP/0/RP0/CPU0:router (config) # call-home
```

Enters Call Home configuration mode.

**Step 3** **vrf vrf-name**

**Example:**

```
RP/0/RP0/CPU0:router (config-call-home) # vrf v1
```

Configures call home for the specified VRF. VRF works only for the http transport method. It does not work for the email transport method.

---

## Configuring Call Home Data Privacy

---

**Step 1** **configure**

**Example:**

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

**Step 2** **call-home**

**Example:**

```
RP/0/RP0/CPU0:router (config) # call-home
```

Enters the call home configuration submode.

**Step 3** **data-privacy { level { normal | high } | hostname }**

**Example:**

```
RP/0/RP0/CPU0:router (config-call-home) # data-privacy level high
```

Scrubs data from call-home message to protect the privacy of the user. The default data-privacy level is normal.

- **normal** - scrubs all normal level commands , such as , password/ username/ ip/ destination.
- **high** - scrubs all normal level commands plus the IP domain name and IP address commands.
- **hostname** - scrubs all high-level or normal-level commands plus the hostname command. It may cause Smart Call Home processing failure.

**Note** Enabling the data-privacy command can affect CPU utilization when scrubbing a large amount of data.

---

# Sending Smart License Data

This task enables the user to configure sending Smart License data through HTTPS transport method in TAC or user-defined profile:

---

**Step 1**    **configure****Example:**

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

**Step 2**    **call-home****Example:**

```
RP/0/RP0/CPU0:router (config) # call-home
```

Enters Call Home configuration mode.

**Step 3**    **profile *name***

Perform either one of the below actions:

- For sending Smart License data in TAC profile:

```
RP/0/RP0/CPU0:router (config-call-home) # profile CiscoTAC-1
```

- For sending Smart License data in user-defined profile:

```
RP/0/RP0/CPU0:router (config-call-home) # profile user1
```

**Step 4**    **active****Example:**

```
RP/0/RP0/CPU0:router (config-call-home-profile) # active
```

Enables the destination profile. By default, a user-defined profile is enabled when it is created.

**Step 5**    **reporting smart-licensing-data****Example:**

```
RP/0/RP0/CPU0:router (config-call-home-profile) # reporting smart-licensing-data
```

Enables sending Smart Licensing data.

**Step 6**    **destination transport-method http****Example:**

```
RP/0/RP0/CPU0:router (config-call-home-profile) # destination transport-method http
```

Configures the message transport method for HTTPS.

**Step 7**    **destination address http *url*****Example:**

```
RP/0/RP0/CPU0:router (config-call-home-profile) # destination address http https://example.com
```



Configures the destination HTTPS address to which Smart License data is sent.

**Step 8** **destination preferred-msg-format {long-text |short-text| xml}**

**Example:**

```
RP/0/RP0/CPU0:router (config-call-home-profile) # destinationpreferred-msg-format xml
```

(Optional) Configures a preferred message format. The default is XML.

**Step 9** Use the **commit** or **end** command.

**commit** —Saves the configuration changes and remains within the configuration session.

**end** —Prompts user to take one of these actions:

- **Yes** — Saves configuration changes and exits the configuration session.
  - **No** —Exits the configuration session without committing the configuration changes.
  - **Cancel** —Remains in the configuration session, without committing the configuration changes.
-





## CHAPTER 8

# Managing Router Hardware

This chapter describes the concepts and tasks used to manage and configure the hardware components of a router running the Cisco IOS XR software.

This module contains the following topics:

- [RP Redundancy and Switchover, on page 81](#)
- [NPU Power Optimization, on page 86](#)
- [Dynamic Power Management, on page 91](#)
- [Ability to Set Maximum Power Limit for the Router , on page 107](#)
- [Configuring the Compatibility Mode for Various ASIC Types, on page 108](#)
- [Storage Media Sanitization, on page 115](#)
- [Excluding Sensitive Information in Show Running Configurations Output, on page 120](#)
- [Fabric Link Management for Uncorrectable Errors, on page 123](#)
- [Fault Recovery Handling, on page 126](#)

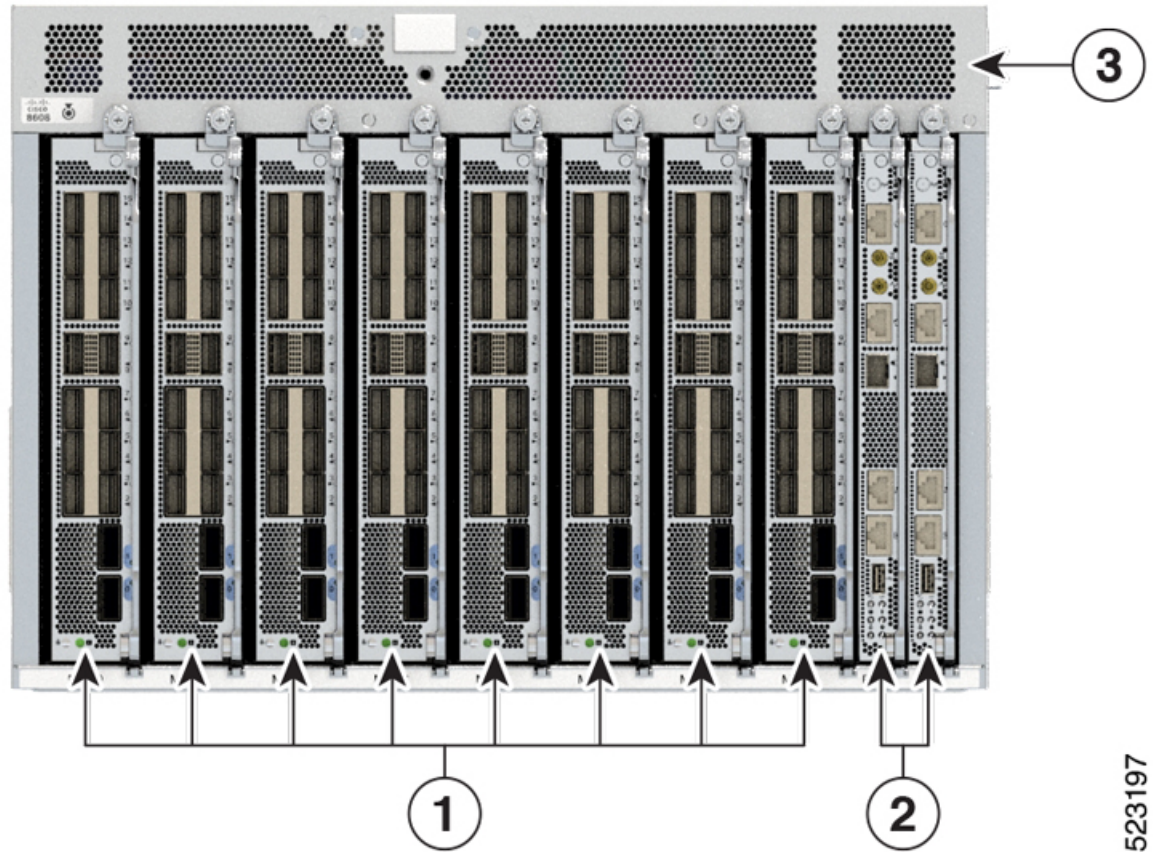
## RP Redundancy and Switchover

This section describes RP redundancy and switchover commands and issues.

### Establishing RP Redundancy

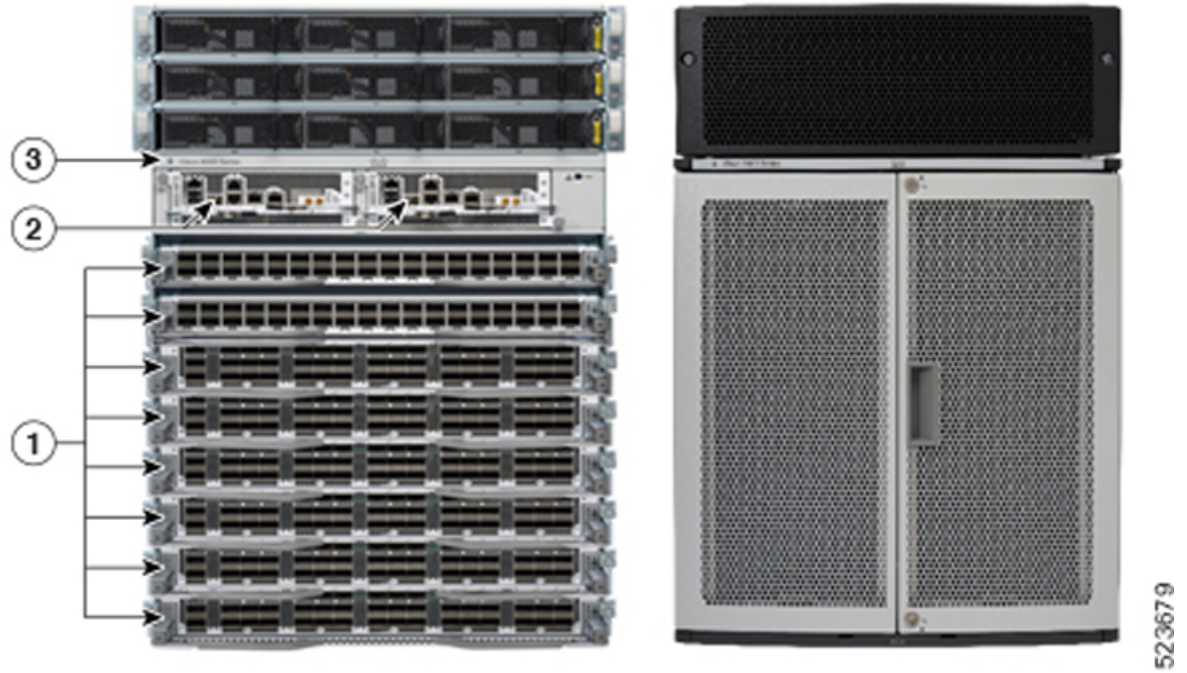
Your router has two slots for RPs: RP0 and RP1 (see [Figure 4: Redundant Set of RP Installed in Slots RP0 and RP1 in an Cisco 8608 8-Slot Centralized Chassis, on page 82](#) and [Figure 5: Redundant Set of RP Installed in Slots RP0 and RP1 in an Cisco 8808 8-Slot Distributed Chassis, on page 83](#)). RP0 is the slot on the left, facing the front of the chassis, and RP1 is the slot on right. These slots are configured for redundancy by default, and the redundancy cannot be eliminated. To establish RP redundancy, install RP into both slots.

Figure 4: Redundant Set of RP Installed in Slots RP0 and RP1 in a Cisco 8608 8-Slot Centralized Chassis



523197

Figure 5: Redundant Set of RP Installed in Slots RP0 and RP1 in an Cisco 8808 8-Slot Distributed Chassis



1	Modular Port Adaptors (MPAs)
2	Route Processors (RPs)
3	Chassis

## Determining the Active RP in a Redundant Pair

During system startup, one RP in each redundant pair becomes the active RP. You can tell which RP is the active RP in the following ways:

- The active RP can be identified by the green Active LED on the faceplate of the card. When the Active LED turns on, it indicates that the RP is active and when it turns off, it indicates that the RP is in standby.
- The slot of the active RP is indicated in the CLI prompt. For example:

```
RP/0/RP1/CPU0:router#
```

In this example, the prompt indicates that you are communicating with the active RP in slot RP1.

- Enter the **show redundancy** command in EXEC mode to display a summary of the active and standby RP status. For example:

```
RP/0/RP0/CPU0:router# show redundancy

This node (0/RP0/CPU0) is in ACTIVE role
Partner node (0/RP1/CPU0) is in STANDBY role
Standby node in 0/RP1/CPU0 is ready
```

```

Reload and boot info
-----
RP reloaded Fri Apr  9 03:44:28 2004: 16 hours, 51 minutes ago
This node booted Fri Apr  9 06:19:05 2004: 14 hours, 16 minutes ago
Last switch-over Fri Apr  9 06:53:18 2004: 13 hours, 42 minutes ago
Standby node boot Fri Apr  9 06:54:25 2004: 13 hours, 41 minutes ago
Standby node last not ready Fri Apr  9 20:35:23 2004: 0 minutes ago
Standby node last ready Fri Apr  9 20:35:23 2004: 0 minutes ago
There have been 2 switch-overs since reload

```

## Role of the Standby RP

The second RP to boot in a redundant pair automatically becomes the standby RP. While the active RP manages the system and communicates with the user interface, the standby RP maintains a complete backup of the software and configurations for all cards in the system. If the active RP fails or goes off line for any reason, the standby RP immediately takes control of the system.

## Summary of Redundancy Commands

RP redundancy is enabled by default in the Cisco IOS XR software, but you can use the commands described in [Table 10: RP Redundancy Commands, on page 84](#) to display the redundancy status of the cards or force a manual switchover.

*Table 10: RP Redundancy Commands*

Command	Description
<b>show redundancy</b>	Displays the redundancy status of the RP. This command also displays the boot and switch-over history for the RP.
<b>redundancy switchover</b>	Forces a manual switchover to the standby RP. This command works only if the standby RP is installed and in the “ready” state.
<b>show platform</b>	Displays the status for node, including the redundancy status of the RP cards. In EXEC mode, this command displays status for the nodes assigned to the SDR. In administration EXEC mode, this command displays status for all nodes in the system.

## Automatic Switchover

Automatic switchover from the active RP to the standby RP occurs only if the active RP encounters a serious system error, such as the loss of a mandatory process or a hardware failure. When an automatic switchover occurs, the RPs respond as follows:

- If a standby RP is installed and “ready” for switchover, the standby RP becomes the active RP. The original active RP attempts to reboot.
- If the standby RP is not in “ready” state, then both RPs reboot. The first RP to boot successfully assumes the role of active RP.

## RP Redundancy During RP Reload

The **reload** command causes the active RP to reload the Cisco IOS XR software. When an RP reload occurs, the RPs respond as follows:

- If a standby RP is installed and “ready” for switchover, the standby RP becomes the active RP. The original active RP reboots and becomes the standby RP.
- If the standby RP is not in the “ready” state, then both RPs reboot. The first RP to boot successfully assumes the role of active RP.

## Manual Switchover

If a standby RP is installed and ready for switchover, you can force a manual switchover using the **redundancy switchover** command or reloading the active RP using the **reload** command.

### Manual Switchover Using the Reload Command

You can force a manual switchover from the active RP to the standby RP by reloading the active RP using the **reload** command. As active RP reboots, the current standby RP becomes active RP, and rebooting RP switches to standby RP.

```
RP/0/RP0/CPU0:router# reload
RP/0/RP1/CPU0:router#
```

### Manual Switchover Using the Redundancy Switchover Command

You can force a manual switchover from the active RP to the standby RP using the **redundancy switchover** command.

If a standby RP is installed and ready for switchover, the standby RP becomes the active RP. The original active RP becomes the standby RP. In the following example, partial output for a successful redundancy switchover operation is shown:

```
RP/0/RP0/CPU0:router# show redundancy

This node (0/RP0/CPU0) is in ACTIVE role
Partner node (0/RP1/CPU0) is in STANDBY role
Standby node in 0/RP1/CPU0 is ready

RP/0/RP0/CPU0:router# redundancy switchover
Updating Commit Database. Please wait...[OK]
Proceed with switchover 0/RP0/CPU0 -> 0/RP1/CPU0? [confirm]
Initiating switch-over.
RP/0/RP0/CPU0:router#

<Your 'TELNET' connection has terminated>
```

In the preceding example, the Telnet connection is lost when the previously active RP resets. To continue management of the router, you must connect to the newly activated RP as shown in the following example:

```
User Access Verification

Username: xxxxx
```

```

Password: xxxxx
Last switch-over Sat Apr 15 12:26:47 2009: 1 minute ago

RP/0/RP1/CPU0:router#

```

If the standby RP is not in “ready” state, the switchover operation is not allowed. In the following example, partial output for a failed redundancy switchover attempt is shown:

```

RP/0/RP0/CPU0:router# show redundancy

Redundancy information for node 0/RP1/CPU0:
=====
Node 0/RP0/CPU0 is in ACTIVE role
Partner node (0/RP1/CPU0) is in UNKNOWN role

Reload and boot info
-----
RP reloaded Wed Mar 29 17:22:08 2009: 2 weeks, 2 days, 19 hours, 14 minutes ago
Active node booted Sat Apr 15 12:27:58 2009: 8 minutes ago
Last switch-over Sat Apr 15 12:35:42 2009: 1 minute ago
There have been 4 switch-overs since reload

RP/0/RP0/CPU0:router# redundancy switchover

Switchover disallowed: Standby node is not ready.

```

## Communicating with a Standby RP

The active RP automatically synchronizes all system software, settings, and configurations with the standby RP.

If you connect to the standby RP through the console port, you can view the status messages for the standby RP. The standby RP does not display a CLI prompt, so you cannot manage the standby card while it is in standby mode.

If you connect to the standby RP through the management Ethernet port, the prompt that appears is for the active RP, and you can manage the router the same as if you had connected through the management Ethernet port on the active RP.

## NPU Power Optimization

*Table 11: Feature History Table*

Feature Name	Release Information	Description
NPU Power Optimization	Release 7.3.15	This feature lets you choose a predefined NPU power mode based on your network's individual requirements, and consequently reducing NPU power consumption.  The <b>hw-module npu-power-profile</b> command is introduced for this feature.



Cisco 8000 series routers are powered by Cisco Silicon One Q200 and Q100 series processors. Cisco Silicon One processors offer high performance, flexible, and power-efficient routing silicon in the market.

NPU Power Optimization feature helps to reduce NPU power consumption by running a processor in a predefined mode. There are three NPU power modes—high, medium, and low. Based on your network traffic and power consumption requirements, you can choose to run the processor in any one of the three NPU power modes.

- High: The router will use the maximum amount of power, resulting in the best possible performance.
- Medium: The router power consumption and performance levels are both average.
- Low: The router operates with optimal energy efficiency while providing a modest level of performance.



**Note** We recommend that you work with your Cisco account representatives before implementing this feature in your network.

On a Q200-based Cisco 8200 series chassis, you can configure an NPU power mode on the entire router.

On a Q200-based Cisco 8800 series chassis, you can configure an NPU power mode only on fabric cards and line cards.

The following table lists the supported hardware, and their default NPU power mode:

**Table 12: Supported Hardware and Default Modes**

Supported Hardware	Default NPU Power Mode
Cisco 8200 32x400 GE 1RU fixed chassis (8201-32FH)	High
88-LC0-36FH without MACSec, based on Q200 Silicon Chip	Medium
88-LC0-36FH-M with MACSec, based on Q200 Silicon Chip	Medium
8808-FC0 Fabric Card, based on Q200 Silicon Chip	Low
8818-FC0 Fabric Card, based on Q200 Silicon Chip	Medium



**Caution** We recommend that you use the default NPU power mode on your router.

## Limitations

The NPU power optimization is not supported on the Q100-based systems.

The NPU Power Profile mode is not supported on the following Q200-based line cards:

Table 13: Limitation on Hardware and Power Profile Modes

Hardware	Power Profile Mode
88-LC0-36FH-M	High
88-LC0-34H14FH	High

## Configuring NPU Power Mode

### Configuring NPU power mode on a fixed chassis:

The following example shows how to configure an NPU power mode on a fixed chassis:

```
RP/0/RP0/CPU0:ios(config)#hw-module npu-power-profile high
RP/0/RP0/CPU0:ios(config)#commit

RP/0/RP0/CPU0:ios(config)#reload
```




---

**Note** Note: Reload the chassis for the configurations changes to take effect.

---

### Verifying NPU power mode configuration on a fixed chassis:

Use the **show controllers npu driver** command to verify the NPU power mode configuration:

```
RP/0/RP0/CPU0:ios#show controllers npu driver location 0/RP0/CPU0
Mon Aug 24 23:29:34.302 UTC
=====
NPU Driver Information
=====
Driver Version: 1
SDK Version: 1.32.0.1
Functional role: Active,      Rack: 8203, Type: lcc, Node: 0
Driver ready      : Yes
NPU first started : Mon Aug 24 23:07:41 2020
Fabric Mode:
NPU Power profile: High
Driver Scope: Node
Respawn count    : 1
Availablity masks :
      card: 0x1,   asic: 0x1,   exp asic: 0x1
...

```

### Configuring NPU power mode on a modular chassis

The following example shows how to configure an NPU power mode on a fabric card and a line card:

```
RP/0/RP0/CPU0:ios(config)#hw-module npu-power-profile card-type FC high
RP/0/RP0/CPU0:ios(config)#hw-module npu-power-profile card-type LC low location 0/1/cpu0
RP/0/RP0/CPU0:ios(config)#commit
```



**Note** For the configurations to take effect, you must:

- Reload a line card if the configuration is applied on the line card.
- Reload a router if the configuration is applied on a fabric card.

**Verifying the NPU power mode configuration on a modular chassis**

Use the **show controllers npu driver location** command to verify the NPU power mode configuration:

```
RP/0/RP0/CPU0:ios#show controllers npu driver location 0/1/CPU0
```

```
Functional role: Active,      Rack: 8808, Type: lcc, Node: 0/RP0/CPU0
Driver ready      : Yes
NPU first started : Mon Apr 12 09:57:27 2021
Fabric Mode: FABRIC/8FC
NPU Power profile: High
Driver Scope: Rack
Respawn count    : 1
Availability masks :
      card: 0xba,      ASIC: 0xcfcc,      exp ASIC: 0xcfcc
Weight distribution:
      Unicast: 80,      Multicast: 20
```

```

+-----+
| Process | Connection | Registration | Connection | DLL |
| /Lib    | status     | status       | requests   | registration|
+-----+
| FSDB    | Active     | Active       | 1          | n/a  |
| FGID    | Active     | Active       | 1          | n/a  |
| AEL     | n/a       | n/a          | n/a       | Yes  |
| SM      | n/a       | n/a          | n/a       | Yes  |
+-----+

```

```
Asics :
HP - HotPlug event, PON - Power On reset
HR - Hard Reset,    WB - Warm Boot
```

```

+-----+
| Asic inst. | fap|HP|Slice|Asic|Admin|Oper | Asic state | Last |PON|HR | FW |
| (R/S/A)   | id | |state|type|state|state|           | init |(#)|(#)| Rev |
+-----+
| 0/FC1/2   | 202|1| UP |s123| UP | UP |NRML      |PON  | 1| 0|0x0000|
| 0/FC1/3   | 203|1| UP |s123| UP | UP |NRML      |PON  | 1| 0|0x0000|
| 0/FC3/6   | 206|1| UP |s123| UP | UP |NRML      |PON  | 1| 0|0x0000|
| 0/FC3/7   | 207|1| UP |s123| UP | UP |NRML      |PON  | 1| 0|0x0000|
| 0/FC4/8   | 208|1| UP |s123| UP | UP |NRML      |PON  | 1| 0|0x0000|
| 0/FC4/9   | 209|1| UP |s123| UP | UP |NRML      |PON  | 1| 0|0x0000|
| 0/FC5/10  | 210|1| UP |s123| UP | UP |NRML      |PON  | 1| 0|0x0000|
| 0/FC5/11  | 211|1| UP |s123| UP | UP |NRML      |PON  | 1| 0|0x0000|
| 0/FC7/14  | 214|1| UP |s123| UP | UP |NRML      |PON  | 1| 0|0x0000|
| 0/FC7/15  | 215|1| UP |s123| UP | UP |NRML      |PON  | 1| 0|0x0000|
+-----+

```

SI Info :

```

+-----+
| Card | Board | SI Board | SI Param | Retimer SI | Retimer SI | Front Panel |
|      | HW Version | Version | Version | Board Version | Param Version | PHY |
+-----+

```

```

| FC1 | 0.22 | 1 | 6 | NA | NA | NA
|
| FC3 | 0.21 | 1 | 6 | NA | NA | NA
|
| FC4 | 0.21 | 1 | 6 | NA | NA | NA
|
| FC5 | 0.21 | 1 | 6 | NA | NA | NA
|
| FC7 | 0.21 | 1 | 6 | NA | NA | NA
|

```

```

-----+
Functional role: Active, Rack: 8808, Type: lcc, Node: 0/1/CPU0
Driver ready : Yes
NPU first started : Mon Apr 12 09:58:10 2021
Fabric Mode: FABRIC/8FC
NPU Power profile: Low
Driver Scope: Node
Respawn count : 1
Availablity masks :
    card: 0x1, asic: 0x7, exp asic: 0x7
Weight distribution:
    Unicast: 80, Multicast: 20
-----+

```

```

-----+
| Process | Connection | Registration | Connection | DLL
| /Lib    | status    | status      | requests   | registration|
-----+
| FSDB    | Active    | Active      | 1          | n/a        |
| FGID    | Inactive  | Inactive    | 0          | n/a        |
| AEL     | n/a       | n/a         | n/a        | Yes        |
| SM      | n/a       | n/a         | n/a        | Yes        |
-----+

```

```

Asics :
HP - HotPlug event, PON - Power On reset
HR - Hard Reset, WB - Warm Boot

```

```

-----+
| Asic inst. | fap|HP|Slice|Asic|Admin|Oper | Asic state | Last |PON|HR | FW |
| (R/S/A)   | id | |state|type|state|state|           | init |(#)|(#)| Rev |
-----+
| 0/2/0     | 8 | 1| UP | npu | UP | UP | NRML      | PON  | 1 | 0|0x0000|
| 0/2/1     | 9 | 1| UP | npu | UP | UP | NRML      | PON  | 1 | 0|0x0000|
| 0/2/2     | 10| 1| UP | npu | UP | UP | NRML      | PON  | 1 | 0|0x0000|
-----+

```

```

SI Info :

```

```

-----+
| Card | Board | SI Board | SI Param | Retimer SI | Retimer SI | Front Panel
|
|      | HW Version | Version | Version | Board Version | Param Version | PHY
|
-----+
| LC2 | 0.41 | 1 | 9 | NA | NA | DEFAULT
|
-----+

```

# Dynamic Power Management

Table 14: Feature History Table

Feature Name	Release Information	Description
Dynamic Power Management	Release 7.3.15	<p>The Dynamic Power Management feature considers certain dynamic factors before allocating power to the fabric and line cards.</p> <p>This feature has the following benefits:</p> <ul style="list-style-type: none"> <li>• Reduces number of PSUs required by accurately representing the maximum power consumption</li> <li>• Improves PSU efficiency by providing more accurate power allocation</li> </ul> <p>This feature thus optimizes power allocation and avoids overprovisioning power to a router.</p>
Dynamic Power Management	Release 7.3.2	<p>Previously available for fabric and line cards, this feature that helps avoid excess power allocation by considering dynamic factors before allocating power to them is now available for optical modules.</p> <p>To view the power allocation on a per port basis, a new command “<b>show environment power allocated [details]</b>” is introduced.</p>
Dynamic Power Management	Release 7.3.3	<p>The Dynamic Power Management feature is now supported on the following Cisco 8100 and 8200 series routers:</p> <ul style="list-style-type: none"> <li>• Cisco 8201</li> <li>• Cisco 8202</li> <li>• Cisco 8201-32-FH</li> <li>• Cisco 8101-32-FH</li> </ul>
Dynamic Power Management	Release 7.5.2	<p>The Cisco 8202-32FH-M router will now consider dynamic factors, such as optical modules, NPU power profile, and MACsec mode to enable improved power allocation and utilization.</p>

Prior to Cisco IOS XR Release 7.3.15, when Cisco 8000 series routers were powered on or reloaded, the power management feature reserved power to fabric cards and allocated maximum power to line cards. The

power management feature wouldn't consider dynamic factors, such as the type of fabric or line cards in the chassis, or whether a fabric or line card was present in a slot.

The Dynamic Power Management feature considers such dynamic factors before allocating power to the fabric and line cards.

This feature has the following benefits:

- Reduces number of PSUs required by accurately representing the maximum power consumption
- Improves PSU efficiency by providing more accurate power allocation

This feature thus optimizes power allocation and avoids overprovisioning power to a router.

This feature is supported on the following Cisco 8000 series routers:

- Cisco 8804, 8808, 8812, and 8818 routers
- Cisco 8201, 8202, 8201-32-FH, and 8202-32FH-M routers
- Cisco 8101-32-FH

By default, this feature is enabled on the router.

The Dynamic Power Management feature allocates the total power to a router and its fabric card or line card based on the following parameters:

- Number and type of fabric cards installed on the router
- Fabric cards operating modes (5FC or 8FC)
- Number and type of line cards installed on the router
- Combination of line card and fabric card types installed
- NPU power mode configured on a fabric card
- Number and type of optics installed (supported in Cisco IOS XR Software Release 7.3.2 and later)
- MACSec-enabled ports (supported from Cisco IOS XR Software Release 7.3.3 and later)

For details, see *Dynamic Power Management for MACSec-Enabled Ports* section in the *Configuring MACSec* chapter in the *System Security Configuration Guide for Cisco 8000 Series Routers*.

On 8202-32FH-M router, the Dynamic Power Management feature allocates the total power to a router based on the following parameters:

- Optical modules installed.
- NPU power profile. To identify the mode on which the router is operating, use the `hw-module npu-power-profile` command.
- MACSec mode. By default, MACSec mode is disabled on 8202-32FH-M router.



---

**Note** We recommend you work with your Cisco account representatives to calculate power requirements for the Cisco 8000 series router.

---

### Power Allocation to Empty Card Slot

This feature allocates a minimum required power for all empty LC or FC slots. This minimum power is required to boot the CPU and FPGAs immediately when a card is inserted. The feature doesn't control booting up the CPU and FPGAs. Also, the minimum power is required to detect the card type before the feature decides if there's enough power to power up the data path.

For example, the following **show environment power** command output displays various LC or FC card statuses, and also shows allocated and used power.



**Note** The allocated power capacity shown in the following **show** command output isn't standard capacity. The allocated power capacity varies depending on various other factors.

```
Router# show environment power
Thu Apr 22 12:03:06.754 UTC
=====
CHASSIS LEVEL POWER INFO: 0
=====
Total output power capacity (N + 1)      : 9600W + 6300W
Total output power required              : 9241W
Total power input                        : 6146W
Total power output                       : 5826W
=====
```

Power Module	Supply Type	-----Input-----		-----Output---		Status
		Volts A/B	Amps A/B	Volts	Amps	
0/PT0-PM0	PSU6.3KW-HV	245.5/245.7	5.1/5.0	54.7	43.1	OK
0/PT0-PM1	PSU6.3KW-HV	0.0/245.2	0.0/7.4	54.3	31.7	OK
0/PT0-PM2	PSU6.3KW-HV	0.0/246.9	0.0/7.5	54.1	32.3	OK

```
Total of Power Modules:          6146W/25.0A          5826W/107.1A
=====
```

Location	Card Type	Power Allocated Watts	Power Used Watts	Status
0/RP0/CPU0	8800-RP	95	69	ON
0/RP1/CPU0	-	95	-	RESERVED
<b>0/0/CPU0</b>	<b>88-LC0-36FH</b>	<b>796</b>	<b>430</b>	<b>ON</b>
0/1/CPU0	-	102	-	RESERVED
0/2/CPU0	88-LC0-36FH	796	430	ON
<b>0/3/CPU0</b>	<b>-</b>	<b>102</b>	<b>-</b>	<b>RESERVED</b>
0/4/CPU0	-	102	-	RESERVED
0/5/CPU0	-	102	-	RESERVED
0/6/CPU0	-	102	-	RESERVED
0/7/CPU0	-	102	-	RESERVED
0/8/CPU0	-	102	-	RESERVED
<b>0/9/CPU0</b>	<b>88-LC0-36FH</b>	<b>102</b>	<b>-</b>	<b>OFF</b>
0/10/CPU0	-	102	-	RESERVED
0/11/CPU0	-	102	-	RESERVED
0/FC0	-	26	-	RESERVED
0/FC1	-	26	-	RESERVED
0/FC2	-	26	-	RESERVED
0/FC3	8812-FC	784	509	ON
0/FC4	8812-FC	784	503	ON
0/FC5	8812-FC	26	-	OFF
0/FC6	8812-FC	26	-	OFF
<b>0/FC7</b>	<b>8812-FC</b>	<b>26</b>	<b>-</b>	<b>OFF</b>

0/FT0	8812-FAN	1072	1000	ON
0/FT1	8812-FAN	1072	1012	ON
0/FT2	8812-FAN	1072	861	ON
0/FT3	8812-FAN	1072	1033	ON

This table describes the card slot statuses:

**Table 15: Router Card Slot Status**

Status	Description
RESERVED	When a slot is empty
OFF	When a card is inserted in a slot but power isn't allocated to the card
ON	When a card is allocated power and the card is in operational state

### Low-Power Condition

When you insert an LC or FC in a card slot at the time when the router doesn't have enough power available to allocate to the new card, the dynamic power management feature doesn't provision power to the card. It raises the `ev_power_budget_not_ok` alarm, and gracefully shuts down the card.

In the following `show` command output, an FC inserted in the card slot location 0/FC6 is gracefully shut down due to lack of power:

```
Router# show shelfmgr history events location 0/FC6
Thu Apr 22 12:03:11.763 UTC
NODE NAME      : 0/FC6
CURRENT STATE  : CARD_SHUT_POWERED_OFF
TIME STAMP     : Apr 20 2021 16:49:52
-----
```

DATE	TIME (UTC)	EVENT	STATE
Apr 20 2021	16:49:52	ev_powered_off	CARD_SHUT_POWERED_OFF
Apr 20 2021	16:49:52	ev_device_offline	STATE_NOT_CHANGED
Apr 20 2021	16:49:52	ev_unmapped_event	STATE_NOT_CHANGED
Apr 20 2021	16:49:48	transient_condition	CARD_SHUTDOWN
Apr 20 2021	16:49:48	ev_check_card_down_reaso	CHECKING_DOWN_REASON
Apr 20 2021	16:49:48	ev_timer_expiry	CARD_SHUTDOWN_IN_PROGRESS
<b>Apr 20 2021</b>	<b>16:48:46</b>	<b>ev_power_budget_not_ok</b>	<b>CARD_SHUTDOWN_IN_PROGRESS</b>
Apr 20 2021	16:48:45	transient_condition	POWER_BUDGET_CHECK
Apr 20 2021	16:48:45	ev_fpd_upgrade_not_reqd	CARD_STATUS_CHECK_COMPLETE
Apr 20 2021	16:47:45	ev_card_status_check	CARD_STATUS_CHECK
Apr 20 2021	16:47:45	ev_card_info_rcvd	CARD_INFO_RCVD
Apr 20 2021	16:47:44	ev_device_online	DEVICE_ONLINE
Apr 20 2021	16:47:43	ev_timer_expiry	CARD_POWERED_ON
Apr 20 2021	16:47:33	ev_powered_on	CARD_POWERED_ON
Apr 20 2021	16:47:33	init	CARD_DISCOVERED

However, after an LC, FC, or chassis reload, the dynamic power management feature can't ensure that the same LCs, FCs, optics, or interfaces, which were operational earlier (before the reload), would become active again.





**Note** During a low-power condition, this feature doesn't borrow power from a redundant power supply.

### Power Allocation to Optics

From Cisco IOS XR Release 7.3.2 onwards, power requirement for optics is also considered before allocating power to them.

To identify the power allocated for a particular interface, use the **show environment power allocated [details] location location** command.

When the optical modules are inserted, power is automatically allocated for that interface. If power has been allocated to the interface, then use the “**no shut**” command to enable the interface.

```
Router# show environment power allocated location 0/3/CPU0
Thu Oct 7 22:27:35.732 UTC
```

Location	Components	Power Allocated Watts
0/3/CPU0	Data-path	772
	<b>OPTICS</b>	<b>138</b>
	Total	910

```
Router# show environment power allocated details location 0/3/CPU0
Thu Oct 7 22:27:42.221 UTC
```

Location	Components	Power Allocated Watts
0/3/CPU0	Data-path	772
	0/3/0/0	3
	0/3/0/1	3
	0/3/0/2	3
	0/3/0/3	3
	0/3/0/4	3
	0/3/0/5	3
	0/3/0/6	3
	0/3/0/7	3
	0/3/0/8	3
	0/3/0/9	3
	0/3/0/10	3
	0/3/0/11	3
	0/3/0/12	3
	0/3/0/13	3
	0/3/0/14	3
	0/3/0/15	3
	0/3/0/16	3
	0/3/0/17	3
	0/3/0/18	3
	0/3/0/19	3
	0/3/0/20	3
	0/3/0/21	3
	0/3/0/22	3
	0/3/0/23	3
	0/3/0/24	3

```

0/3/0/25          3
0/3/0/26          3
0/3/0/27          3
0/3/0/28          3
0/3/0/29          3
0/3/0/30          3
0/3/0/31          3
0/3/0/32          3
0/3/0/33          3
0/3/0/34          3
0/3/0/35          3
0/3/0/36          3
0/3/0/37          3
0/3/0/38          3
0/3/0/39          3
0/3/0/40          3
0/3/0/41          3
0/3/0/42          3
0/3/0/43          3
0/3/0/44          3
0/3/0/46          3

```

```

=====
Total                910

```

When the power is not allocated to the interface, the following syslog error and alarms are displayed

```

!<--Syslog Error-->!
#LC/0/3/CPU0:Oct  7 22:46:48.114 UTC: optics_driver[165]: %PKT_INFRA-FM-3-FAULT_MAJOR :
ALARM_MAJOR :POWER ALLOCATION FAIL :DECLARE :0/3/CPU0: Optics0/3/0/44
LC/0/3/CPU0:Oct  7 22:46:48.114 UTC: optics_driver[165]:
%L2-OPTICS-2-QSFP_POWER_ALLOCATION_FAILURE : Not enough power available to enable Optics
0/3/0/44

```

```

!<--Alarm-->!
Router#show alarms brief system active
Thu Oct  7 22:47:19.569 UTC

```

```

-----
Active Alarms
-----

```

Location	Severity	Group	Set Time	Description
----------	----------	-------	----------	-------------

0/3/CPU0	Major	Software	10/07/2021 22:46:48 UTC	Optics0/3/0/44 - hw_optics: Lack of available power to enable the optical module
----------	-------	----------	-------------------------	--

0/3/CPU0	Major	Software	10/07/2021 22:47:06 UTC	Optics0/3/0/46 - hw_optics: Lack of available power to enable the optical module
----------	-------	----------	-------------------------	--

If power is not allocated to an interface and you attempt to enable that interface using the “**no shut**” command, the following syslog error is displayed:

```

LC/0/2/CPU0:Aug 30 18:01:14.930 UTC: eth_intf_ea[262]: %PLATFORM-VEEA-1-PORT_NOT_ENABLED :
Power not allocated to enable the interface HundredGigE0_2_0_6.

```

### Power Allocation to Fixed-Port Routers

The following **show environment power** command output displays power information for fixed-port routers and components.

```

Router# show environment power
Wed Feb 16 21:05:10.001 UTC
=====
CHASSIS LEVEL POWER INFO: 0
=====
Total output power capacity (Group 0 + Group 1) :    1400W +    1400W
Total output power required                      :    1033W
Total power input                               :    390W
Total power output                              :    255W

Power Group 0:
=====
Power      Supply      -----Input-----  -----Output---  Status
Module    Type                Volts      Amps      Volts      Amps
=====
0/PM0     PSU1.4KW-ACPE      244.5      0.8      12.0      11.1      OK

Total of Group 0:                195W/0.8A                133W/11.1A

Power Group 1:
=====
Power      Supply      -----Input-----  -----Output---  Status
Module    Type                Volts      Amps      Volts      Amps
=====
0/PM1     PSU1.4KW-ACPE      244.2      0.8      12.0      10.2      OK

Total of Group 1:                195W/0.8A                122W/10.2A

=====
Location    Card Type                Power      Power      Status
                Allocated      Used
                Watts      Watts
=====
0/RP0/CPU0  8201                    893        -          ON
0/FT0       FAN-1RU-PE              28         -          ON
0/FT1       FAN-1RU-PE              28         -          ON
0/FT2       FAN-1RU-PE              28         -          ON
0/FT3       FAN-1RU-PE              28         -          ON
0/FT4       FAN-1RU-PE              28         -          ON

```

To identify the power allocated for a particular interface, use the **show environment power allocated [details] location location** command.

```

Router# show environment power allocated location 0/RP0/CPU0
Wed Feb 16 21:05:21.360 UTC
=====
Location    Components                Power
                Allocated
                Watts
=====
0/RP0/CPU0  Data-path                  858
                OPTICS                    35
=====
                Total                  893

```

```

Router# show environment power allocated details location 0/RP0/CPU0
Wed Feb 16 21:05:36.142 UTC
=====
Location    Components                Power
                Allocated
                Watts
=====
0/RP0/CPU0  Data-path                  858

```

0/0/0/19	21
0/0/0/18	14
=====	
Total	893

## Disabling Dynamic Power Management

By default, the dynamic power management is enabled on a router. The following example shows how to disable dynamic power management:

```
RP/0/RP0/CPU0:ios(config)#power-mgmt action disable
RP/0/RP0/CPU0:ios(config)#commit
```



**Caution** After disabling the dynamic power management feature, you must manage the router power on your own. So, use this command with caution.



**Note** To reenable dynamic power management, use the **no power-mgmt action disable** command.

## On-demand transfer of Redundant Power Modules to Power Reservation Pool

Table 16: Feature History Table

Feature Name	Release Information	Feature Description
On-demand transfer of Redundant Power Modules to Power Reservation Pool	Release 7.11.1	The Cisco 8800 Series Modular Routers now have a functionality that allows them to transfer their redundant Power Supply Units (PSUs) to the power reservation pool when there is inadequate power supply. This capability helps prevent the router from shutting down hardware components due to a lack of power in the reservation pool, which used to occur due to the router prioritizing redundancy over power availability in the power reservation pool. Consequently, the router now raises an alarm indicating redundancy loss when it transfers PSUs to the power reservation pool. This feature ensures that the router components reserve the necessary power, even when redundancy is enabled.

The Cisco 8000 Series Modular Routers offer redundancy while managing Power Supply Units (PSUs), providing continuous operation if there is PSU failure. By default, the router operates in N+1 redundancy, where N represents the number of PSUs allotted to the power reservation pool for powering the router components, and 1 indicates the backup PSU. You can use the `power-mgmt redundancy-num-pms number` command in XR Config mode to configure the PSU redundancy from N+1 to N+x, where x is the number of redundant PSUs required. The total number of functioning PSUs must be at least x more than the number of PSUs required to support the power demanded by all the components in the system for optimal router functionality. The range of values assigned to x is 0–11, where 0 implies no power redundancy. The router uses the redundant PSUs only when there is a PSU failure. But, if the power requirement of the router increases than the available power offered by PSUs, the router prioritizes maintaining PSU redundancy overpowering the components.

Starting from Cisco IOS XR Release 7.11.1, the Cisco 8800 Modular Routers prioritize powering the router components over preserving redundancy. The router transfers the redundant PSUs to a power reservation pool to power the router components on demand. The router utilizes the redundant PSUs to increase the power capacity in the power reservation pool rather than maintaining redundancy. For example, consider a scenario with 18900W (3 6300W PSUs) available power. Initially, the router reserves 12600W (using 2 PSUs) in the power reservation pool and retains 6300W (one PSU) as a backup to maintain N+1 redundancy. Suppose the router needs to reserve power for any components to power up and needs more power than is available in the reservation pool. In that case, the router uses the entire 18900W with all three PSUs to power the components by transferring the redundant PSU to the power reservation pool. The router then triggers a redundancy loss alarm with such an assignment. However, if any further actions result in reduced power consumption in the router, the system automatically restores redundancy and clears the redundancy lost alarm.

On redundancy loss, the router raises a **Critical** severity **Power Module redundancy lost** alarm. You can use the `show alarms brief` command to view the redundancy lost alarm.

Syslog messages for transforming redundant PSU into borrowable resource:

Syslog message created while redundancy loss (transforming redundant PSU to functional PSU):

```
RP/0/RP0/CPU0:Jul 24 11:49:01.316 UTC: envmon[214]: %PKT_INFRA-FM-3-FAULT_MAJOR : ALARM_MAJOR
:Power Module redundancy lost :DECLARE :0:
```

Syslog message created while restoring redundancy:

```
RP/0/RP0/CPU0:Jul 24 11:49:11.375 UTC: envmon[214]: %PKT_INFRA-FM-3-FAULT_MAJOR : ALARM_MAJOR
:Power Module redundancy lost :CLEAR :0:
```

You can also use the `show environment` view the redundancy status of the PSUs in the router.

The following section details the commands to verify the redundancy status in the router:

#### Router with N+1 redundancy:

```
Router:ios# show environment power
```

```
=====
CHASSIS LEVEL POWER INFO: 0
=====

Total output power capacity (N + 1)      : 12600W + 6300W
Total output power required              : 11545W
Total power input                        : 3302W
Total power output                       : 3004W

=====

Power      Supply      -----Input-----  ----Output---  Status
Module    Type                Volts A/B  Amps A/B  Volts  Amps
=====
```

```

0/PT5-PM0  PSU6.3KW-HV  240.5/241.3  2.2/2.4  55.1  18.3  OK
0/PT5-PM1  PSU6.3KW-HV  240.5/240.8  2.1/2.3  54.8  17.3  OK
0/PT5-PM2  PSU6.3KW-HV  242.2/241.1  2.3/2.4  54.9  19.1  OK
    
```

```

Total of Power Modules:      3302W/13.7A      3004W/54.7A
    
```

```

=====
Location      Card Type      Power      Power      Status
Allocated    Used
Watts        Watts
=====
0/RP0/CPU0    8800-RP        105         78         ON
0/RP1/CPU0    -              105         -          RESERVED
0/0/CPU0      8800-LC-36FH   1097        513        ON
0/1/CPU0      -              102         -          RESERVED
0/2/CPU0      88-LC0-36FH   102         0          OFF
0/3/CPU0      -              102         -          RESERVED
0/4/CPU0      -              102         -          RESERVED
0/5/CPU0      -              102         -          RESERVED
0/6/CPU0      -              102         -          RESERVED
0/7/CPU0      -              102         -          RESERVED
0/8/CPU0      -              102         -          RESERVED
0/9/CPU0      -              102         -          RESERVED
0/10/CPU0     -              102         -          RESERVED
0/11/CPU0     -              102         -          RESERVED
0/12/CPU0     -              102         -          RESERVED
0/13/CPU0     -              102         -          RESERVED
0/14/CPU0     -              102         -          RESERVED
0/15/CPU0     -              102         -          RESERVED
0/16/CPU0     -              102         -          RESERVED
0/17/CPU0     -              102         -          RESERVED
0/FC0         -              32          -          RESERVED
0/FC1         -              32          -          RESERVED
0/FC2         8818-FC0      584         475        ON
0/FC3         -              32          -          RESERVED
0/FC4         8818-FC0      584         472        ON
0/FC5         -              32          -          RESERVED
0/FC6         -              32          -          RESERVED
0/FC7         -              32          -          RESERVED
0/FT0         8818-FAN      1786        237        ON
0/FT1         8818-FAN      1786        228        ON
0/FT2         8818-FAN      1786        234        ON
0/FT3         8818-FAN      1786        228        ON
    
```

**Router with redundancy loss:**

Router:ios# **sh env power**

```

=====
CHASSIS LEVEL POWER INFO: 0
=====
Total output power capacity (N + 1)      : 18900W +      0W
Total output power required              : 12689W
Total power input                         : 3302W
Total power output                        : 3004W
=====

Power      Supply      -----Input-----  -----Output-----  Status
Module     Type          Volts A/B  Amps A/B  Volts      Amps
=====
    
```

```

0/PT5-PM0 PSU6.3KW-HV 240.5/241.3 2.2/2.4 55.1 18.3 OK
0/PT5-PM1 PSU6.3KW-HV 240.5/240.8 2.1/2.3 54.8 17.3 OK
0/PT5-PM2 PSU6.3KW-HV 242.2/241.1 2.3/2.4 54.9 19.1 OK
  
```

```

Total of Power Modules:          3302W/13.7A          3004W/54.7A
  
```

```

=====
Location      Card Type          Power      Power      Status
Allocated    Used
Watts        Watts
=====
0/RP0/CPU0    8800-RP            105        78          ON
0/RP1/CPU0    -                  105        -           RESERVED
0/0/CPU0      8800-LC-36FH      1097       513         ON
0/1/CPU0      -                  102        -           RESERVED
0/2/CPU0      88-LC0-36FH      916        510         ON
0/3/CPU0      -                  102        -           RESERVED
0/4/CPU0      -                  102        -           RESERVED
0/5/CPU0      -                  102        -           RESERVED
0/6/CPU0      -                  102        -           RESERVED
0/7/CPU0      -                  102        -           RESERVED
0/8/CPU0      -                  102        -           RESERVED
0/9/CPU0      -                  102        -           RESERVED
0/10/CPU0     -                  102        -           RESERVED
0/11/CPU0     -                  102        -           RESERVED
0/12/CPU0     -                  102        -           RESERVED
0/13/CPU0     -                  102        -           RESERVED
0/14/CPU0     -                  102        -           RESERVED
0/15/CPU0     -                  102        -           RESERVED
0/16/CPU0     -                  102        -           RESERVED
0/17/CPU0     -                  102        -           RESERVED
0/FC0         -                  32         -           RESERVED
0/FC1         -                  32         -           RESERVED
0/FC2         8818-FC0          749        475         ON
0/FC3         -                  32         -           RESERVED
0/FC4         8818-FC0          749        472         ON
0/FC5         -                  32         -           RESERVED
0/FC6         -                  32         -           RESERVED
0/FC7         -                  32         -           RESERVED
0/FT0         8818-FAN          1786       237         ON
0/FT1         8818-FAN          1786       225         ON
0/FT2         8818-FAN          1786       234         ON
0/FT3         8818-FAN          1786       228         ON
  
```

```

Router:ios# sh alarms brief system active
  
```

```

-----
Active Alarms
  
```

```

-----
Location      Severity  Group      Set Time
Description
-----
0/RP0/CPU0    Critical  Software   10/27/2023 00:22:08 UTC
Redundancy Partner Not Present

0             Major    Environ    10/27/2023 00:23:48 UTC    Power
Module redundancy lost
  
```

## On-demand transfer of Redundant Power Modules to Power Reservation Pool

Plane-0 status	0/RP0/CPU0	Minor	Fabric	10/27/2023 00:22:39 UTC	Fabric
Plane-1 status	0/RP0/CPU0	Minor	Fabric	10/27/2023 00:22:39 UTC	Fabric
Plane-3 status	0/RP0/CPU0	Minor	Fabric	10/27/2023 00:22:39 UTC	Fabric
Plane-5 status	0/RP0/CPU0	Minor	Fabric	10/27/2023 00:22:39 UTC	Fabric
Plane-6 status	0/RP0/CPU0	Minor	Fabric	10/27/2023 00:22:39 UTC	Fabric
Plane-7 status	0/RP0/CPU0	Minor	Fabric	10/27/2023 00:22:39 UTC	Fabric
Communications Failure With Cisco Licensing Cloud	0/RP0/CPU0	Major	Software	10/27/2023 00:22:59 UTC	
<b>Module redundancy lost</b>	0	<b>Major</b>	<b>Environ</b>	<b>10/27/2023 00:23:48 UTC</b>	<b>Power</b>



# Power Redundancy Protection

Table 17: Feature History Table

Feature Name	Release Information	Feature Description
Power Redundancy Protection	Release 24.1.1	<p>You can now prevent power module exhaustion or failure due to power redundancy issues in the power feeds with the help of alarms that warn that the total output power required by the router exceeds the total feed redundancy capacity. You can configure either single-fault protection or dual fault protection, depending on whether you want to trigger alarms during redundancy failures in the power supply feed, PSU redundancy, or both.</p> <p>The feature introduces these changes:</p> <p><b>CLI:</b></p> <ul style="list-style-type: none"> <li>• <code>power-mgmt feed-redundancy</code></li> <li>• The <code>Total feed redundancy capacity</code> field is added to the <code>show environment</code> command.</li> </ul>

The Cisco 8000 Series Modular Routers have two redundancy mechanisms to ensure the router continues functioning even during power supply failures:

- The PSU redundancy involves having extra power supplies that can take over if one fails, ensuring continuous operation.
- The power feed redundancy divides the input power into A and B feeds. When both feeds are functioning normally, they share the power load equally. However, if one of the feeds fails, the other feed scales up to its maximum capacity or the power supply unit (PSU) will operate with reduced input to ensure that the power supply to the router is uninterrupted.

These power redundancy options provide a high level of reliability and minimize the risk of network downtime due to power supply failures.

The routers now have power redundancy protection that triggers alarms for PSU and feed redundancy failures when the total output power required by the router exceeds its total feed redundancy capacity. You can configure the total feed redundancy capacity in two modes- single fault protection and dual fault protection.

The **single fault protection** mode monitors the router against a **power supply feed or PSU** redundancy failure. Meanwhile, the **dual fault protection** monitors the router against a **power supply feed and PSU** redundancy failure simultaneously. You can also customize the PSU single feed capacity in the router. Each

PSU has a default power range for the single feed; you can configure a value within the range to meet your specific infrastructure requirements.

The feed redundancy alarm is triggered when the total output power required exceeds the total feed redundancy capacity. The router's total feed capacity is determined by the least of two factors: feed redundancy capacity and PSU redundancy capacity. The PSU redundancy capacity is the number of power supply units minus the redundant ones (N) multiplied by a dual feed capacity. On the other hand, the feed redundancy capacity is the total number of PSUs multiplied by a single feed capacity. In single-fault protection, the PSU refers to the router's total number of power supply units (N+1). In dual-fault protection, the PSU refers to the number of power supply units minus the redundant ones (N).

For example, consider a router that has a total of 9 PSUs with a default N + 1 power redundancy configuration. The PSU feed capacity with dual feed is 4800 W and the single feed capacity value is set 3200 W, then the total feed redundancy capacity would be:

Power Redundancy Protection	Total Number of PSUs	PSU redundancy	Number of PSUs minus the redundant ones (N)	Dual Feed Capacity	Single Feed Capacity	Feed Redundancy Capacity	PSU Redundancy Capacity	Total Feed Redundancy Capacity
Single fault protection	9	N+1	8	4800 W	3200 W	28800 W	38400 W	28800 W
Dual fault protection	9	N+1	8	4800 W	3200 W	25600 W	38400 W	25600 W

## Guidelines and Restrictions for Power Redundancy Protection

- By default, the router doesn't enable Power Redundancy Protection.
- The Power Redundancy Protection feature doesn't impact the power budgeting in the routers.
- For maximum power redundancy protection, use the dual fault protection.
- For total feed redundancy capacity calculations, the router considers only the PSUs with A and B inputs. Both A and B inputs must be within the operating range in healthy conditions. If either feed is unavailable, the router excludes such PSUs from the calculations.
- The router considers all PSUs, including redundant PSUs with two feeds (within the operating range in healthy condition) for feed redundancy capacity in single fault protection. However, the router excludes the redundant PSUs for feed redundancy capacity in dual fault protection. If the router has 8 PSUs and N+3 redundancy, single fault protection calculation considers all eight PSUs, whereas dual fault protection considers just 5 PSUs.

## Configure Power Redundancy Protection

To configure the power redundancy protection mode and PSU single feed capacity, you can use the [power-mgmt feed-redundancy](#) command.

### Single fault protection with PSU single feed capacity set to 2400 Watts

**Configuration:**

```
Router# config
Router(config)# power-mgmt feed-redundancy single-fault-protection capacity 2400
Router(config)# commit
```

### Running Configuration:

```
Router# show run power
...
power-mgmt feed-redundancy single-fault-protection capacity 2400
...
```

### Verification:

```
Router# show env power
=====
CHASSIS LEVEL POWER INFO: 0
=====
Total output power capacity (N + 1)      : 28800W + 4800W
Total output power required          : 6679W >>>>> 1
Total power input                        : 2394W
Total power output                       : 2066W
Total feed redundancy capacity (Single Fault) : 16800W >>>>> 2
/*The router triggers feed redundancy loss alarm when 1 > 2.**//
=====
Power      Supply      -----Input-----  -----Output---  Status
Module     Type                Volts A/B    Amps A/B    Volts    Amps
=====
0/PT0-PM0  PSU4.8KW-DC100    62.8/62.7  2.6/2.5    55.2     5.3    OK
0/PT0-PM1  PSU4.8KW-DC100    62.7/62.7  2.7/2.6    55.3     5.3    OK
0/PT0-PM3  PSU4.8KW-DC100    61.0/62.7  2.6/2.5    55.2     4.8    OK
0/PT1-PM0  PSU4.8KW-DC100    67.3/67.3  2.7/2.5    55.3     5.2    OK
0/PT1-PM1  PSU4.8KW-DC100    67.3/67.2  2.8/2.7    55.3     5.7    OK
0/PT1-PM2  PSU4.8KW-DC100    67.3/67.4  2.7/2.7    55.2     5.6    OK
0/PT1-PM3  PSU4.8KW-DC100    67.3/67.3  2.6/2.5    55.3     5.5    OK
Total of Power Modules:                2394W/36.7A                2066W/37.4A
```

### Dual fault protection with PSU single feed capacity set to 2400 Watts

#### Configuration:

```
Router# config
Router(config)# power-mgmt feed-redundancy dual-fault-protection capacity 2400
Router(config)# commit
```

#### Running Configuration:

```
Router# show run power
...
power-mgmt feed-redundancy dual-fault-protection capacity 2400
...
```

#### Verification:

```
Router# show env power
=====
CHASSIS LEVEL POWER INFO: 0
=====
Total output power capacity (N + 1)      : 28800W + 4800W
Total output power required          : 6679W >>>>> 1
Total power input                        : 2394W
Total power output                       : 2066W
Total feed redundancy capacity (Dual Fault) : 14400W >>>>> 2
/*The router triggers feed redundancy loss alarm when 1 > 2.**//
=====
```

Power Module	Supply Type	-----Input-----		-----Output---		Status
		Volts A/B	Amps A/B	Volts	Amps	
0/PT0-PM0	PSU4.8KW-DC100	62.8/62.7	2.6/2.5	55.2	5.3	OK
0/PT0-PM1	PSU4.8KW-DC100	62.7/62.7	2.7/2.6	55.3	5.3	OK
0/PT0-PM3	PSU4.8KW-DC100	61.0/62.7	2.6/2.5	55.2	4.8	OK
0/PT1-PM0	PSU4.8KW-DC100	67.3/67.3	2.7/2.5	55.3	5.2	OK
0/PT1-PM1	PSU4.8KW-DC100	67.3/67.2	2.8/2.7	55.3	5.7	OK
0/PT1-PM2	PSU4.8KW-DC100	67.3/67.4	2.7/2.7	55.2	5.6	OK
0/PT1-PM3	PSU4.8KW-DC100	67.3/67.3	2.6/2.5	55.3	5.5	OK
Total of Power Modules:		2394W/36.7A		2066W/37.4A		

### Alarms for power redundancy loss

You can use the [show alarms brief](#) command to view the power redundancy alarm:



**Note** The router triggers the Power Module redundancy feed mode lost alarm only when **Total output power required** exceeds **Total feed redundancy capacity**.

```
Router# show alarms brief system active
```

```
-----
Active Alarms
```

```
-----
Location          Severity      Group          Set Time          Description
```

```
-----
0                Major        Environ        11/27/2023 12:55:08 UTC    Power Module redundancy
feed mode lost
```

### System Log messages for power redundancy loss

Syslog message created while power redundancy loss (total output power exceeds total feed redundancy capacity):

```
RP/0/RP0/CPU0:Dec 15 10:24:29.489 UTC: envmon [123]: %PKT_INFRA-FM-3-FAULT_MAJOR : ALARM_MAJOR
:Power Feed redundancy lost :DECLARE :0
```

# Ability to Set Maximum Power Limit for the Router

Table 18: Feature History Table

Feature Name	Release Information	Feature Description
Ability to Set Maximum Power Limit for the Router	Release 7.11.1	<p>We are introducing functionality to set the maximum power limit for a router to improve power management and distribution in the PSUs. It prevents a router from using more than the configured power and also gives the ability to limit the reservation pool regardless of how many power supplies are present. In the previous releases, the ability to prevent a router from using more than a configured amount of power was unavailable.</p> <p>This feature introduces the following change:</p> <p><b>CLI</b></p> <ul style="list-style-type: none"> <li><b>power-mgmt configured-power-capacity</b></li> </ul>

In the earlier releases, there was no mechanism to limit the power a router consumed. Routers could draw more than the infrastructure could handle. Over power consumption could result in system brownout.

With the Cisco IOS XR Software Release 7.11.1, you can allocate system power based on max power capacity configuration. This prevents the router from allocating more power than the infrastructure can handle. It also gives you the ability to limit power to a router according to your infrastructure requirements. The max power capacity parameter doesn't allow power consumed by the hardware to cross the configured amount.

The criteria to set maximum power limit is that the value must be set between the current allocated power and the available maximum power at time of configuration.

This feature is not applicable for fixed routers.

A new command **power-mgmt configured-power-capacity** has been introduced with this feature.

A new alarm **PKT\_INFRA-FM-3-FAULT\_MAJOR : ALARM\_MAJOR :Power reservation exceeds configured power** is introduced to be raised when the max power capacity is crossed.



**Note** This alarm is extremely rare and is raised only when the power reservation exceeds configured power. This can only happen when hardware is inserted, it is granted power without a request, such as a fan tray.

# Configuring the Compatibility Mode for Various ASIC Types

Table 19: Feature History Table

Feature Name	Release Information	Description
Configure Compatibility Mode for Q100 and Q200-based Line Cards	Release 7.7.1	<p>You can now configure the compatibility behavior of line cards to operate in Q100 mode (default behavior) or in Q200 mode when you have a mix of Q100-based line cards and Q200-based line cards that are installed in a router.</p> <p>In earlier releases, in a mixed mode combination, where multiple generations of line cards were installed on a distributed chassis, the behavior was to make the second-generation line cards interoperate with the first-generation line cards. However, this led the NPUs to set lower resource limits for the newer generation line cards to ensure backward compatibility. Also, the router didn't fully utilize the improved scale, higher capacity, and feature-rich capabilities of the newer generation line cards.</p> <p>This compatibility feature now enables you to select if you want the line cards to operate in Q100 or Q200 mode.</p> <p>The <b>hw-module profile npu-compatibility</b> command is introduced for this feature.</p>

In earlier releases, if you install a mix of Q100-based line cards and Q200-based line cards, the Q200-based line cards operate in a scaled-down (Q100) mode by default.

The compatibility feature, applicable to Cisco 8800 Series modular/distributed chassis, now allows you to choose if you want line cards to operate in Q100 (default behavior), Q200, or P100 mode. In Q200 mode, the router boots only the Q200-based line cards and gracefully shuts down the Q100-based line cards.

For example, if a router has a Q100 ASIC family line card and you try to add a line card from the Q200 ASIC family, the Q200 ASIC line card operates in a scaled down mode to be able to work with the older generation-Q100 line cards. With the new implementation, you can choose if you want the router to work in the Q100 mode or shutdown the Q100-based linecards, and use the Q200 ASIC line cards in the Q200 mode.

## FAQs About the New Implementation

- **Can the line cards still be used in scaled down mode, like in the previous scenario?**

Yes, you can still switch to the previous implementation, if you may, to the scaled down mode.

- **What all ASICs can participate in the new implementation?**

P100, Q200, and Q100

- **Is there any default ASIC set by the system?**

The ASIC default is based on the Fabric Cards (FCs) and route processor cards used in a distributed chassis. However, you can choose to change the ASIC mode to Q200, Q100, or P100.

• **Do I need to reboot the router after implementing a new ASIC line card?**

**Yes**, reboot the router for the new ASIC line cards to take effect.

### Usage Guidelines and Limitations

The following guidelines and limitations apply when you configure the line cards from different ASIC families:

- By default, a mix of Q100, Q200, and P100 line cards results in the Q200 and P100 line cards operating in Q100 (scaled-down) mode. Configuring Q100 mode results in the same (default) behavior.
- To be able to use the improved scale, higher capacity, and feature-rich capabilities of the Q200-based line cards, use the `hw-module profile npu-compatibility` command and set it to operate in the Q200 mode. Else, the Q200-based line cards scale down to the Q100 mode, which is the default behavior. The same behavior applies to the P100-based line cards.
- Reboot the router for the compatibility mode to take effect. If the system detects a noncompatible line card, it shuts down that line card. For example, in Q200 mode, the router boots only the Q200-based line cards and gracefully shuts down the Q100-based line cards.
- The `hw-module profile npu-compatibility` command isn't configurable on the Cisco 8100 and 8200 Series fixed chassis.
- For 8800-RP, the default ASIC mode is Q100. For 8800-RP2, the default ASIC mode is Q200.

This table lists the Q100, Q200, and P100-based line cards that support the compatibility mode:

ASIC Family	Line Card
Q100-based line cards	8800-LC-48H
	8800-LC-36FH
Q200-based line cards	88-LC0-34H14FH
	88-LC0-36FH
	88-LC0-36FH-M
P100-based line cards	88-LC1-36EH

### Route Processor Card Behavior with ASICs

A newer generation Route Processor (RP) card takes precedence over an older generation RP card when installed from different ASIC families. The precedence followed by the system is: P100 > Q200 > Q100.

If you have Q200-based line cards and an older generation RP card (8800-RP) installed on your router, the router sets the ASIC mode for the line cards to Q100. However, you can change the ASIC mode from Q100 to Q200 by using the `hw-module profile npu-compatibility` command. Setting the ASIC mode to a newer generation ASIC allows you to utilize their improved scale, higher capacity, and feature-rich capabilities when you replace your RPs with a newer generation RP.

For instance, if your chassis is equipped with an 8800-RP route processor card set to ASIC mode as Q200, upgrading to an 8800-RP2 RP card won't require changing the ASIC mode from Q100 to Q200.



**Note** For 8800-RP, the default ASIC mode is Q100. For 8800-RP2, the default ASIC mode is Q200.

### Line Card Behavior with ASICs

The following table explains how the various line cards take precedence when installed from different ASIC families. The precedence followed by the system is: P100 > Q200 > Q100, where the newer generation line cards take precedence over an older generation line card.

ASIC Family of Installed Line Cards	Compatibility Mode Configured?	Compatibility Mode	Router Behavior during Bootup for the Line Cards
Q200 and Q100	N	Default (Q100)	Q200 line cards boot up and operate in Q100 mode, Q100 up.
	Y	Q200	Q200 line cards boot up, Q100 line cards shut down.
	Y	Q100	All line cards boot up, Q200 line cards operate in Q100 mode.
Q200 and Q200	N	Default (Q100)	Both the Q200 line cards boot up and operate in Q100 mode.
	Y	Q200	Both the Q200 line cards boot up

### Supported Compatibility Modes on Fabric Cards, RP Cards, and Line Cards

The following table details the fabric cards (FCs), the supported RP card (8800-RP) on the FCs, the default ASIC mode, the line cards, and if you can configure the **hw-module profile npu-compatibility** command on those line cards:



Table 20: Compatibility Mode between Fabric Cards and 8800-RP

Fabric Card	Fabric Card ASIC	RP(8800-RP)	Default ASIC	Supported Line Cards	Configure NPU Compatibility?
8808-FC 8812-FC 8818-FC	Q100	Supported	Q100	Q100-based and Q200-based	Yes  You can configure the ASIC mode to Q200 if you have only Q200-based line cards installed on your chassis.
8804-FC0	Q200	Supported	Q100	Q100-based and Q200-based	Yes  You can configure the ASIC mode to Q200 if you have only Q200-based line cards installed on your chassis.
8808-FC1	F100	Not Supported	NA	NA	NA

The following table details the fabric cards (FCs), the supported RP card (8800-RP2) on the FCs, the default ASIC mode, the line cards, and if you can configure the **hw-module profile npu-compatibility** command on those line cards:

Table 21: Compatibility Mode between Fabric Cards and 8800-RP2

Fabric Card	Fabric Card ASIC	RP(8800-RP2)	Default ASIC	Supported Line Cards	Configure NPU Compatibility?
8808-FC 8812-FC 8818-FC	Q100	Not Supported	NA	NA	NA
8812-FC0	Q200	Supported	Q200	Q200-based	NA
8808-FC1	F100	Supported	P100	P100-based	Yes  You can configure the ASIC mode to Q200 if you have both Q200-based and P100-based line cards installed on your chassis.

### IP Forwarding Feature Support for Different ASICs and RPs

The following tables list the IP forwarding features supported on various ASIC combinations, such as Q100, Q200, P100, when you have 8800-RP or 8800-RP1 route processor cards installed on your chassis.

- This [Table 22: IP Forwarding Feature Support on Q100 and Q200 ASICs with 8800-RP Card](#) lists the supported and unsupported features when you have Q100 and Q200 ASICs, and 8800-RP route processor card installed on your chassis.
- This [Table 23: IP Forwarding Feature Support on Q200 and P100 ASICs with 8800-RP1 Card](#) lists the supported and unsupported features when you have Q200 and P100 ASICs, and 8800-RP1 route processor card installed on your chassis.

**Table 22: IP Forwarding Feature Support on Q100 and Q200 ASICs with 8800-RP Card**

Feature	Q100	Q200
<a href="#">EVPN</a>	Not Supported	Single-homing Support
<a href="#">SRv6</a>	Not Supported	Supported
<a href="#">Class-based Forwarding</a>	Not Supported	Supported
<a href="#">Policy-Based Tunnel Selection</a>	Not Supported	Supported
<a href="#">VRF Redirect</a>	Not Supported	Supported
<a href="#">VLAN</a>	Not Supported	Supported
<a href="#">First Hop Redundancy Protocol</a>	Not Supported	Supported
<a href="#">BGP Labeled Unicast over RSVP-TE</a>	Not Supported	Supported
<a href="#">SRv6-H-Insert</a>	Not Supported	Not Supported
<a href="#">IP-in-IP</a>	Not Supported	Supported
<a href="#">Pseudowire Headend (PWHE)</a>	Not Supported	Not Supported
<a href="#">Lawful Intercept</a>	Not Supported	Not Supported

Table 23: IP Forwarding Feature Support on Q200 and P100 ASICs with 8800-RP1 Card

Feature	Q200	P100
EVPN	Single-homing Supported	Multi-homing Supported
SRv6	Supported	Supported
Class-based Forwarding	Supported	Supported
Policy-Based Tunnel Selection	Supported	Supported
VRF Redirect	Supported	Supported
VLAN	Supported	Supported
First Hop Redundancy Protocol	Supported	Supported
BGP Labeled Unicast over RSVP-TE	Supported	Supported
SRv6-H-Insert	Not Supported	Supported
IP-in-IP	Supported	Not Supported
Pseudowire Headend (PWHE)	Not Supported	Supported
Lawful Intercept	Not Supported	Supported

### Configuring Line Cards from Different ASICs

To configure a router for handling line cards of different ASIC families, use the `hw-module profile npu-compatibility` command. To go back to the default mode, use the `no` form of this command.

The following are the options available in command and their descriptions:

<code>npu-compatibility</code>	Allows you to make a router compatible with an ASIC family.
<code>mode-name</code>	Allows you to set the mode, such as Q100, Q200, or P100.

The following is a configuration example:

```
Router:ios(config)#hw-module profile npu-compatibility q200
Tue Dec 7 15:06:53.697 UTC
Chassis mode will be activated after a manual reload of chassis/all line cards
Router:ios(config)#commit
Tue Dec 7 15:06:54.646 UTC
LC/0/1/CPU0:Dec 7 15:06:54.796 UTC: npu_drvr292:
%FABRIC-NPU_DRV-3-HW_MODULE_PROFILE_NPU_COMPATIBILITY_CHASSIS_CFG_CHANGED : Please reload
chassis for the configuration to take effect
end
Router:ios(config)#end
Router:ios#
```

### Running Configuration

```
RP/0/RP0/CPU0:ios# show ver
```

```
Mon Jun 27 19:25:52.947 UTC
Cisco IOS XR Software, Version 7.7.1.27I LNT
Copyright (c) 2013-2022 by Cisco Systems, Inc.
```

## Build Information:

```
Built By      : ingunawa
Built On     : Wed Jun 01 23:50:09 UTC 2022
Build Host   : iox-ucs-060
Workspace    : /auto/iox-ucs-060-san1/prod/7.7.1.27I.SIT_IMAGE/8000/ws
Version     : 7.7.1.27I
Label       : 7.7.1.27I
```

```
cisco 8000 (VXR)
cisco 8808 (VXR) processor with 32GB of memory
ios uptime is 3 minutes
Cisco 8808 8-slot Chassis
```

```
RP/0/RP0/CPU0:ios#
```

```
RP/0/RP0/CPU0:ios# conf
```

```
Mon Jun 27 19:24:40.621 UTC
```

```
RP/0/RP0/CPU0:ios(config)# hw-module profile npu-compatibility ?
```

```
  Q100 Use Q100 for Chassis mode
```

```
  Q200 Use Q200 for Chassis mode
```

**Verification**

```
RP/0/RP0/CPU0:ios# show hw-module profile npu-compatibility matrix
```

```
Mon Jun 27 19:41:47.560 UTC
```

```
Node           Card Type           NPU Type
-----
0/0/CPU0       8800-LC-48H           Q100
```

```

NPU Type      Compatibility      Compatibility
Mode Q100     Mode Q200
-----
Q100          Compatible        Not Compatible
Q200          Compatible        Compatible
```

```
Default mode: Q100
```

```
RP/0/RP0/CPU0:ios# show hw-module profile npu-compatibility
```

```
Mon Jun 27 19:41:59.318 UTC
```

```

-----
Knob           Status           Applied   Action
-----
npu_compatibility  Unconfigured    N/A      None
```

```
RP/0/RP0/CPU0:ios#
```

# Storage Media Sanitization

Table 24: Feature History Table

Feature Name	Release Information	Feature Description
Storage Media Sanitization	Release 7.5.1Release 7.3.4	<p>To comply with NIST SP 800-88 guidelines for Media Sanitization, it is important that your organization ensures that no easily reconstructible data is stored in the router and associated devices after it has left the control of your organization or is no longer protected by confidentiality categorization.</p> <p>With this feature, you can erase and overwrite any sensitive data, configuration, or keys present in the route processor or line card, ensuring media sanitization and preventing unauthorized data retrieval.</p>

When you identify an RP or line card for RMA, or you require to ship it outside your organization, a service personnel may not be available on-site to remove the card immediately. However, you can reset your RP or line card to erase customer-sensitive data and let the RP or line card remain in the slot. The RP or line card shuts down automatically after the factory reset is complete.

## Guidelines

- We recommend using **factory-reset** without performing **commit replace** for securely removing the files in the misc/config folder.
- The RP or line card shuts down automatically if the factory reset takes more than 30 minutes, you can perform the factory reset again. The console displays the following log message during automatic shutdown:
 

```
[ TIME ] Timed out starting Power-Off.
[ !! ] Forcibly powering off as result of failure.
```
- If your router has dual RPs, and to perform the factory reset on both the RPs, first reset the standby RP from the active RP. After the reset is complete, the standby RP automatically shuts down, you can then reset the active RP.

## Prerequisites

The RP or line card must be operational to perform factory reset.

## Commands

Use the **factory-reset** command for erasing the following folders of RP or line card:

- /misc/disk1
- /misc/scratch
- /var/log
- /misc/config

Run the following command through the console port of the router to erase customer-sensitive data in the RP or line card:

**factory-reset location** <location-id> - erases customer-sensitive data in the specified location




---

**Note** Factory-reset logs are displayed on the console port of the node where the reset is performed.

---

The following steps explain how to reset your RP or line card to factory settings:

1. Erasing the RP or line card folder contents: Run the **factory-reset location** command to delete the encryption keys and erase the customer-sensitive data from the RP or line card.

The following example shows how to perform the factory-reset command on an RP:

```
Router#factory-reset location 0/RP1/CPU0
Factory reset requested
Started punching watchdog
Started cleaning up mount point: /misc/scratch
Started syncing folder: /misc/scratch
Finished syncing folder: /misc/scratch
Finished cleaning up mount point: /misc/scratch
factory_reset_stop.sh
+++++
Started cleaning up mount point: /var/log
Started syncing folder: /var/log
Finished syncing folder: /var/log
Finished cleaning up mount point: /var/log
factory_reset_stop.sh
+++++
Started cleaning up mount point: /misc/disk1
Started syncing folder: /misc/disk1
Finished syncing folder: /misc/disk1
Finished cleaning up mount point: /misc/disk1
factory_reset_stop.sh
+++++
Started cleaning up folder: /misc/config
UTC 2022 Started syncing folder: /misc/config
Finished syncing folder: /misc/config
Finished cleaning up folder: /misc/config
factory_reset_stop.sh
+++++
Started cleaning up folder: /var/xr/enc/misc/config
/var/xr/enc/misc/config not present
Finished cleaning up folder: /var/xr/enc/misc/config
factory_reset_stop.sh
+++++
Started cleaning up folder: /mnt/rootfs/misc/config
/mnt/rootfs/misc/config not present
```

```

Finished cleaning up folder: /mnt/rootfs/misc/config
factory_reset_stop.sh
+++++
Encrypted logical volume does not exist. Nothing to remove.
/usr/local/etc/fpga-functions: line 797: 10912 Terminated
/usr/local/etc/punch-wd.sh
Stopped punching watchdog

```

2. Verifying factory reset: Use the **show shelfmgr history events location** command to verify the successful completion of the factory-reset in the standby RP or line card.

The following example shows how to verify the factory-reset command:

```

RP/0/RP0/CPU0:Router#show shelfmgr history events location 0/RP1/CPU0
Tue Mar 15 01:45:56.402 UTC
NODE NAME      : 0/RP1/CPU0
CURRENT STATE  : CARD_SHUT_POWERED_OFF
TIME STAMP     : Mar 15 2022 01:44:47
-----
DATE           TIME (UTC)  EVENT                               STATE
-----
Mar 15 2022 01:44:47  ev_powered_off      CARD_SHUT_POWERED_OFF
Mar 15 2022 01:44:47  transient_condition CARD_SHUTDOWN
Mar 15 2022 01:44:47  ev_check_card_down_reaso CHECKING_DOWN_REASON
Mar 15 2022 01:44:47  ev_os_halted        OS_HALTED
Mar 15 2022 01:44:43  ev_factory_reset_done  FACTORY_RESET_DONE
Mar 15 2022 01:33:16  ev_factory_reset_started FACTORY_RESET_IN_PROGRESS
Mar 15 2022 01:33:11  ev_os_halting        OS_HALT_IN_PROGRESS
Mar 15 2022 01:33:10  ev_xr_shut           START_OS_HALT
Mar 15 2022 01:33:09  ev_ack_ok            STATE_NOT_CHANGED
Mar 15 2022 01:33:09  ev_graceful_shut     CARD_SHUTDOWN_IN_PROGRESS
Mar 15 2022 00:55:31  ev_xr_ready          XR_RUN

```

## Commands

Use the **factory-reset** command for erasing the following folders of RP or line card:

- /misc/disk1
- /misc/scratch
- /var/log
- /misc/config

Run the following command through the console port of the router to erase customer-sensitive data in the RP or line card:

**factory-reset { reload | shutdown } location <location-id>** - erases customer-sensitive data in the specified location. Use the reload option in the command to reload the RP or line card after the factory reset and use the shutdown option to shut down the RP or line card after the factory reset.



**Note** Factory-reset logs are displayed on the console port of the node where the reset is performed.

The following steps explain how to reset your RP or line card to factory settings:

1. Erasing the RP or line card folder contents: Run the **factory-reset { reload | shutdown } location** command to delete the encryption keys and erase the customer-sensitive data from the RP or line card.

The following example shows how to perform the factory-reset shutdown command on an RP:

```
Router#factory-reset shutdown location 0/RP1/CPU0
Factory reset requested
Started punching watchdog
Started cleaning up mount point: /misc/scratch
Started syncing folder: /misc/scratch
Finished syncing folder: /misc/scratch
Finished cleaning up mount point: /misc/scratch
factory_reset_stop.sh
+++++
Started cleaning up mount point: /var/log
Started syncing folder: /var/log
Finished syncing folder: /var/log
Finished cleaning up mount point: /var/log
factory_reset_stop.sh
+++++
Started cleaning up mount point: /misc/disk1
Started syncing folder: /misc/disk1
Finished syncing folder: /misc/disk1
Finished cleaning up mount point: /misc/disk1
factory_reset_stop.sh
+++++
Started cleaning up folder: /misc/config
UTC 2022 Started syncing folder: /misc/config
Finished syncing folder: /misc/config
Finished cleaning up folder: /misc/config
factory_reset_stop.sh
+++++
Started cleaning up folder: /var/xr/enc/misc/config
/var/xr/enc/misc/config not present
Finished cleaning up folder: /var/xr/enc/misc/config
factory_reset_stop.sh
+++++
Started cleaning up folder: /mnt/rootfs/misc/config
/mnt/rootfs/misc/config not present
Finished cleaning up folder: /mnt/rootfs/misc/config
factory_reset_stop.sh
+++++
Encrypted logical volume does not exist. Nothing to remove.
/usr/local/etc/fpga-functions: line 797: 10912 Terminated
/usr/local/etc/punch-wd.sh
Stopped punching watchdog
```

The following example shows how to perform the factory-reset reload command on an RP:

```
Router#factory-reset reload location 0/RP1/CPU0
Factory reset requested
Started punching watchdog
Started cleaning up mount point: /misc/scratch
Started syncing folder: /misc/scratch
Finished syncing folder: /misc/scratch
Finished cleaning up mount point: /misc/scratch
+++++
Started cleaning up mount point: /var/log
Started syncing folder: /var/log
Finished syncing folder: /var/log
Finished cleaning up mount point: /var/log
+++++
Started cleaning up mount point: /misc/disk1
Started syncing folder: /misc/disk1
Finished syncing folder: /misc/disk1
Finished cleaning up mount point: /misc/disk1
```



```

+++++
Started cleaning up folder: /misc/config
Started syncing folder: /misc/config
Finished syncing folder: /misc/config
Finished cleaning up folder: /misc/config
+++++
Started cleaning up folder: /var/xr/enc/misc/config
/var/xr/enc/misc/config not present
Finished cleaning up folder: /var/xr/enc/misc/config
+++++
Started cleaning up folder: /mnt/rootfs/misc/config
/mnt/rootfs/misc/config not present
Finished cleaning up folder: /mnt/rootfs/misc/config
+++++
Encrypted logical volume does not exist. Nothing to remove.
/usr/local/etc/fpga-functions: line 790: 4137 Terminated
/usr/local/etc/punch-wd.sh
Stopped punching watchdog
    
```

2. Verifying factory reset: Use the **show shelfmgr history events location** command to verify the successful completion of the factory-reset in the standby RP or line card.

The following example shows how to verify the factory-reset shutdown command:

```

RP/0/RP0/CPU0:Router#show shelfmgr history events location 0/RP1/CPU0
Tue Mar 15 01:45:56.402 UTC
NODE NAME      : 0/RP1/CPU0
CURRENT STATE  : CARD_SHUT_POWERED_OFF
TIME STAMP     : Mar 15 2022 01:44:47
-----
DATE           TIME (UTC)  EVENT                               STATE
-----
Mar 15 2022 01:44:47  ev_powered_off          CARD_SHUT_POWERED_OFF
Mar 15 2022 01:44:47  transient_condition     CARD_SHUTDOWN
Mar 15 2022 01:44:47  ev_check_card_down_reaso CHECKING_DOWN_REASON
Mar 15 2022 01:44:47  ev_os_halted            OS_HALTED
Mar 15 2022 01:44:43  ev_factory_reset_done   FACTORY_RESET_DONE
Mar 15 2022 01:33:16  ev_factory_reset_started FACTORY_RESET_IN_PROGRESS
Mar 15 2022 01:33:11  ev_os_halting           OS_HALT_IN_PROGRESS
Mar 15 2022 01:33:10  ev_xr_shut              START_OS_HALT
Mar 15 2022 01:33:09  ev_ack_ok               STATE_NOT_CHANGED
Mar 15 2022 01:33:09  ev_graceful_shut        CARD_SHUTDOWN_IN_PROGRESS
Mar 15 2022 00:55:31  ev_xr_ready             XR_RUN
    
```

The following example shows how to verify the factory-reset reload command:

```

RP/0/RP0/CPU0:Router#show shelfmgr history events location 0/RP0/CPU0
Tue Mar 15 01:45:56.402 UTC
NODE NAME      : 0/RP0/CPU0
CURRENT STATE  : CARD_SHUT_POWERED_OFF
TIME STAMP     : Mar 15 2022 01:44:47
-----
DATE           TIME (UTC)  EVENT                               STATE
-----
Jun 29 2022 13:48:34  ev_xr_ready             XR_RUN
Jun 29 2022 13:48:10  ev_card_info_rcvd       CARD_INFO_RCVD
Jun 29 2022 13:47:52  ev_xr_init              XR_INITIALIZING
Jun 29 2022 13:47:44  ev_kernel_booting       STATE_NOT_CHANGED
Jun 29 2022 13:47:14  ev_kernel_booting       KERNEL_BOOTING
Jun 29 2022 13:46:53  ev_unmapped_event       STATE_NOT_CHANGED
Jun 29 2022 13:46:53  ev_bios_started         BIOS_STARTED
Jun 29 2022 13:46:51  ev_bios_ready           BIOS_READY
Jun 29 2022 13:46:10  ev_unmapped_event       STATE_NOT_CHANGED
    
```

```

Jun 29 2022 13:46:10 ev_powered_on CARD_POWERED_ON
Jun 29 2022 13:46:05 ev_card_reset_done CARD_RESET
Jun 29 2022 13:46:05 transient_condition CARD_RESETTING
Jun 29 2022 13:46:05 ev_check_card_down_reaso CHECKING_DOWN_REASON
Jun 29 2022 13:46:05 ev_os_halted OS_HALTED
Jun 29 2022 13:45:50 ev_factory_reset_done FACTORY_RESET_DONE
Jun 29 2022 13:34:09 ev_factory_reset_started FACTORY_RESET_IN_PROGRESS
Jun 29 2022 13:33:59 ev_os_halting OS_HALT_IN_PROGRESS
Jun 29 2022 13:33:58 ev_xr_shut START_OS_HALT
Jun 29 2022 13:33:56 ev_graceful_reload CARD_SHUTDOWN_IN_PROGRESS
Jun 29 2022 09:18:43 ev_xr_ready XR_RUN
Jun 29 2022 09:17:37 ev_card_info_rcvd CARD_INFO_RCVD
Jun 29 2022 09:17:32 ev_powered_on CARD_POWERED_ON
Jun 29 2022 09:17:31 init CARD_DISCOVERED

```

## Excluding Sensitive Information in Show Running Configurations Output

Table 25: Feature History Table

Feature Name	Release Information	Feature Description
Excluding Sensitive Information in Show Running Configurations Command Output	Release 7.5.4	<p>You can now exclude sensitive information such as strings, usernames, passwords, comments, or IP addresses within the <b>show running-configuration</b> command output by enabling sanitization on the nonvolatile generation (NVGEN) process.</p> <p>With this feature, you can achieve better data protection to prevent cybersecurity risks compared to regular router algorithms.</p> <p>This feature introduces the <b>nvgen default-sanitize</b> command.</p>

The **show running configuration** command uses the nonvolatile generation (NVGEN) process in IOS-XR software to collect configuration information from every system component and construct a running configuration file to create its output. However, this file may contain sensitive information, including usernames, passwords, and IP addresses, which could pose a security threat when obfuscation algorithms in the router are weak compared to modern cryptographic standards.

In this feature, you can mask the following types of sensitive information in the show running configurations:

- Strings
- Usernames
- Passwords
- Comments

- IP Addresses

On enabling the sanitization in show running configurations, the NVGEN process replaces the corresponding information with **<removed>** string. For example, if you enable sanitization for IP Addresses, the show running configuration includes the **<removed>** string in place of all the IP Addresses in the output.

### Sanitizing Strings

#### Configuration

```
Router# config
Router:(config)# nvgen default-sanitize strings
Router:(config)# commit
```

#### Running Configuration

```
Router# show run nvgen
nvgen
  default-sanitize strings
!
```

#### Verification

```
Router# show run int Hu0/2/0/4
interface HundredGigE0/2/0/4
  ! This is comment 1
  description <removed>
!
```

### Sanitizing Usernames

#### Configuration

```
Router# config
Router:(config)# nvgen default-sanitize usernames
Router:(config)# commit
```

#### Running Configuration

```
Router# show run nvgen
nvgen
  default-sanitize usernames
!
```

#### Verification

```
Router# show run username test
username <removed>
  group root-lr
  password 7 172864HJWBHBCWH
!
```

### Sanitizing Passwords

#### Configuration

```
Router# config
Router:(config)# nvgen default-sanitize passwords
Router:(config)# commit
```

#### Running Configuration

```
Router# show run nvgen
nvgen
```

```
default-sanitize passwords
!
```

### Verification

```
Router# show run username test
username test
  group root-lr
  password 7 <removed>
!
```

### Sanitizing Comments

#### Configuration

```
Router# config
Router:(config)# nvgen default-sanitize comments
Router:(config)# commit
```

#### Running Configuration

```
Router# show run nvgen
nvgen
  default-sanitize comments
!
```

### Verification

```
Router# show run int Hu0/2/0/4
interface HundredGigE0/2/0/4
  ! <comments removed>
  description This is bundle member
!
```

### Sanitizing IP Addresses

#### Configuration

```
Router# config
Router:(config)# nvgen default-sanitize ipaddrs
Router:(config)# commit
```

#### Verification

```
Router# show run int Hu0/2/0/4
interface HundredGigE0/2/0/4
  ! This is comment 1
  description This is bundle member
  ipv4 address <removed> <removed>
!
```

# Fabric Link Management for Uncorrectable Errors

**Table 26: Feature History Table**

Feature Name	Release Information	Feature Description
Fabric Link Management for Uncorrectable Errors	Release 24.2.11	<p>You can now run your fabric links error-free using the forward error correction (FEC) technique.</p> <p>The feature allows you to determine the link quality by monitoring the noisy fabric links during and post bring-up.</p> <p>This feature introduces the <b>hw-module fabric-fec-monitor disable</b> command.</p>

Forward error correction (FEC) is a method for obtaining error control in data transmission in which the transmitter sends redundant data and the receiver recognizes only the portion of the data that contains no apparent errors. When FEC is used in data transmissions, the receiver can detect and correct a limited number of errors.

The Cisco IOS XR router will not bring the link to the data plane if the link is noisy at inception (during bring up). If the link becomes noisy post bring up, fabric link will be re-set and re-tuned. If this event continues for five times with in an hour then fabric link will be shutdown permanently. Post link up, polling interval for link error is 10 minutes.

Fabric link management feature uses FEC as the criteria to determine if a link is good. The router receives a notification for every bad FEC on each fabric port. FEC can correct up to 15 bits beyond which the error is considered as uncorrectable error. This feature allows you to make fabric links run error-free.



**Note** In Cisco IOS XR Release 24.2.11, this feature is enabled only for Q200 based line cards and Fabric cards.

## FEC bin index

FEC bin index indicates the number of bit errors.

For example, FEC bin index of a good link is {12642341946,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,} and for a bad link it is {221568554,532162,414888,101277,144171,3317,61681,4065,741,1977586,138,62,17,6,1,}.

If the FEC bin index more or equal to ten non-zero bits (bin >= 10 non-zero) before the link is up, then the FEC will show the link as bad FEC. When the link is up and the FEC bin index more or equal to 13 non-zero bits (bin >= 13 non-zero), the FEC will show the link as bad FEC.

## Link States for a Noisy Fabric Link

When there is a noisy fabric link, any one of the following link states can be possible:

- Link does not come up at all.

- Link comes up, but fluctuates.
- Link comes up, but generates uncorrectable errors.

The network traffic flow is not impacted if the link never comes up and there will be packet drops observed for the other two states of the link mentioned above.

## Monitor FEC Fabric Links

For every FEC report, the router performs the following process to monitor the fabric links:

1. If the fabric link is faulty before the link is up, the router will retune and checks again for the FEC improvement.
2. If the link quality does not improve after retuning, the router displays the syslog message after tuning for 100 times and will not bring the link to the data plane.
3. Post link up, when the fabric link becomes noisy, the router will collect a snapshot and retune after the first failure.
4. From second failure to fifth failure, the MAC port will be stopped and re-activated (retune will be done as part of this process).
5. If the fabric link fails for the sixth time within an hour, the router will permanently shut down the link.

### Verify the FEC Links

Verify the FEC link information using **show controllers npu link-info** command.

```
router#show controllers npu link-info rx 254 255 fsm instance 0 location 0/2/CPU0 detail
Sat Jan 13 00:39:49.448 UTC
```

```
Node ID: 0/2/CPU0
Link ID: 0/2/0/254    255    Oper State: DOWN    0/FC1/2/158
Event                State                Timestamp
-----+-----+-----+
LINK_UP_INTR         UP                   Sat Jan 13 00:18:44 2018
BAD_FEC_BELOW_THR   PORT_ACTIVATE_DELAYED Sat Jan 13 00:19:16 2018
LINK_MON             FAB_PORT_CREATED    Sat Jan 13 00:19:17 2018
LINK_MON             ACTIVATED           Sat Jan 13 00:19:19 2018
LINK_UP_INTR         MAC_UP              Sat Jan 13 00:19:24 2018
LINK_UP_INTR         PEER_DISCOVERY     Sat Jan 13 00:19:24 2018
LINK_UP_INTR         PEER_DETECTED      Sat Jan 13 00:19:24 2018
LINK_UP_INTR         TOPOLOGY_CHECK     Sat Jan 13 00:19:24 2018
LINK_UP_INTR         SYNC_WAIT          Sat Jan 13 00:19:24 2018
LINK_UP_INTR         KEEPALIVE_START    Sat Jan 13 00:19:24 2018
LINK_UP_INTR         CHECK_REACH        Sat Jan 13 00:19:24 2018
LINK_UP_INTR         UP                  Sat Jan 13 00:19:24 2018
BAD_FEC              UP                  Sat Jan 13 00:20:16 2018
DIS_PERM_SHUT        MAC_UP              Sat Jan 13 00:20:16 2018
DIS_PERM_SHUT        STOPPED             Sat Jan 13 00:20:16 2018
-----+-----+-----+
```

This table describes the significant fields shown in the above example.

Table 27: show controllers npu link-info Field Descriptions

Field	Description
BAD_FEC_BELOW_THR	There are FEC failures, but the number of failures has not exceeded the predefined threshold (in this case, 5 per hour). The router retunes and checks for FEC improvement.
BAD_FEC	This part of the log entry indicates that FEC detected failures, and the number of these failures surpassed a predefined threshold. As a result, the decision was made to permanently shut down the affected interface or port as a protective measure.
DIS_PERM_SHUT	The link or port has been intentionally disabled and is in a shutdown state after FEC fails for the threshold limit (After fifth failure).

### System Log messages

The router displays the following syslog messages after retuning:

- If the link is noisy at inception (during bring up), the router displays the following syslog message after tuning for 100 times:

```
LC/0/2/CPU0:Jan 13 00:56:03.939 UTC: npu_drvr[128]:
%FABRIC-NPU_DRVVR-3-NPU_CPA_GEN_ERR_INFO : Link 0/254 has tuned 100 times and failed to
come up. FEC bin is filled to 11
```

- If the link is noisy post bring up, the router permanently shuts down the link and displays the following syslog message:

```
LC/0/2/CPU0:Jan 13 00:20:16.251 UTC: npu_drvr[128]:
%FABRIC-NPU_DRVVR-3-NPU_CPA_GEN_ERR_INFO : FEC check failures on link 0/254. FEC bin is
filled to 14
```

### Disable Fabric Link Management for Uncorrectable Errors

Fabric link management for uncorrectable errors is enabled by default. To disable this feature, use the **hw-module fabric-fec-monitor disable** command in XR Config mode mode.

The following example shows how to disable the fabric FEC monitor:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# hw-module fabric-fec-monitor disable
RP/0/RP0/CPU0:router(config)# commit
```

# Fault Recovery Handling

Table 28: Feature History Table

Feature Name	Release Information	Feature Description
Fault Recovery Handling	Release 24.2.11	<p>You can now configure the number of fault recovery attempts by a line card, fabric card or a route processor before it permanently shuts down, thus preventing a faulty card from entering into a cycle of automatic recovery.</p> <p>This feature introduces the following change:</p> <p><b>CLI:</b></p> <ul style="list-style-type: none"> <li>• <a href="#">hw-module fault-recovery</a></li> </ul> <p><b>YANG DATA Model:</b></p> <ul style="list-style-type: none"> <li>• New XPath for Cisco-IOS-XR-hw-module-cfg.yang (see <a href="#">Github, YANG Data Models Navigator</a>)</li> </ul>

In the previous releases, if a line card, fabric card or a route processor experienced a fault, they used to trigger fault recovery and reboot themselves to be operational. Fault recovery mechanism was time based as the fault recovery count used to reset to zero if the card remained operational for more than hour. After the fault recovery count exceeded five, then the faulty card was shut down. As power related faults triggered were not frequent, and fault recovery count used to reset to zero, the card never entered the shut down mode. As a result the card always attempted for fault recovery.

With the Cisco IOS XR Software Release 24.2.11, we have introduced the **hw-module fault-recovery** command with which you can set the number of times a fault recovery can take place before permanently shutting down a faulty card.



**Note** This configuration is not applicable for BMC instance

## How to Configure the Fault Recovery Attempts

### Configuration Examples

The configuration example shows how to configure the fault recovery attempts on the fabric card FC0.



```
Router#configure
Router (config)#hw-module fault-recovery location 0/FC0 count 1
Router(config)#commit
```

## Verification

Use **show running-config formal | include hw-module** command to display the number of times a card can initiate recovery attempts before shutting down .

```
Router#show running-config formal | include hw-module
Building configuration...
hw-module fault-recovery location 0/FC0 count 1
```

The following system logs are generated when the number of fault recovery attempts on the card exceeds the configured count:

```
RP/0/RP0/CPU0:Dec 4 15:44:22.950 PST: shelfmgr[121]:
%PLATFORM-SHELFMGR-2-FAULT_ACTION_CARD_SHUTDOWN : Forced shutdown requested for card 0/FC0.
Reason Fault retry attempts exceeded configured count(1)

RP/0/RP0/CPU0:Dec 4 15:44:25.247 PST: shelfmgr[121]: %PLATFORM-SHELFMGR-4-CARD_SHUTDOWN :
Shutting down 0/FC0: Fault retry attempts exceeded configured count(1)
```

Use the **show reboot history** command to get the reason of card shutting down. In the following example, it shows that the card was shut down due to **Fault retry attempts exceeded configured count(1)**.

```
RP/0/RP0/CPU0:ios#show reboot history location 0/FC0 detail
Mon Dec 4 15:44:55.827 PST
-----
No   Attribute      Value
-----
1    Time (PST)     Dec 04 2023 15:44:22
     Cause Code    0x0800000d
     Cause String  REBOOT_CAUSE_FM
     Graceful Reload No
     Kdump Requested No
     Reason       Fault retry attempts exceeded configured count(1)
```

Use the **show platform** command to see the current state of the card that was shut down because of Fault recovery handling feature.

```
RP/0/RP0/CPU0:ios#show platform
Mon Oct 2 21:08:03.383 UTC

Node                               Type                               State                               Config state
-----
0/RP0/CPU0                          8800-RP (Active)                   IOS XR RUN                          NSHUT
0/RP0/BMC0                          8800-RP                            OPERATIONAL                          NSHUT
0/RP1/CPU0                          8800-RP (Standby)                 IOS XR RUN                          NSHUT
0/RP1/BMC0                          8800-RP                            OPERATIONAL                          NSHUT
0/3/CPU0                            8800-LC-48H                       IOS XR RUN                          NSHUT
0/FC0                               8812-FC                          SHUT DOWN                          NSHUT
0/FC3                               8812-FC                            OPERATIONAL                          NSHUT
0/FT0                               SF-D-12-FAN                       OPERATIONAL                          NSHUT
0/FT1                               SF-D-12-FAN                       OPERATIONAL                          NSHUT
0/FT2                               SF-D-12-FAN                       OPERATIONAL                          NSHUT
0/FT3                               SF-D-12-FAN                       OPERATIONAL                          NSHUT
0/PT0                               FAM7000-ACHV-TRAY                 OPERATIONAL                          NSHUT
0/PT1                               FAM7000-ACHV-TRAY                 OPERATIONAL                          NSHUT
0/PT2                               FAM7000-ACHV-TRAY                 OPERATIONAL                          NSHUT
Router#
```





## CHAPTER 9

# The Network Configuration Protocol

The Network Configuration Protocol (Netconf) provides mechanisms to install, manipulate, and delete the configuration of network devices. It uses an Extensible Markup Language (XML)-based data encoding for the configuration data as well as the protocol messages, as defined in RFC6241. Yang is a data modeling language used with Netconf, as defined in RFC6020.

Netconf uses a simple RPC-based (Remote Procedure Call) mechanism to facilitate communication between a client and a server. The client can be a script or application typically running as part of a network manager. The server is typically a network device.

Netconf runs within a Secure Shell (SSH) session as an SSH subsystem, as defined in RFC6242.

The configuration of features need not be done the traditional way (using CLIs), the client application (controller) reads the Yang model and communicates with the Netconf server (IOS XR) accordingly.

- [Netconf Sessions and Operations, on page 129](#)
- [The Yang data model , on page 130](#)
- [Netconf and Yang, on page 131](#)
- [Supported Yang Models , on page 132](#)
- [Denial of Services Defense for Netconf-Yang, on page 132](#)
- [Enabling NETCONF over SSH, on page 133](#)

## Netconf Sessions and Operations

A Netconf session is the logical connection between a network configuration application and a network device. A device should be capable of supporting multiple sessions and atleast one Netconf session.

Characteristics of a netconf session:

- Netconf is connection-oriented - SSH is the underlying transport.
- The netconf client establishes session with the server.
- Netconf sessions are established with the *hello* message. Features and capabilities are announced.
- Sessions can be terminated using the *close* or *kill* messages.

Basic Netconf operations:

- Get configuration <get-config>
- Get all information <get>

- Edit configuration <edit-config>
- Copy configuration <copy-config>



**Note** <copy-config> does not support source attribute with “data store” at present.

- <lock>, <unlock>
- <kill-session>
- <close-session>
- Commit configuration <commit>

## The Yang data model

Each feature has a defined Yang Model which is synthesized from the schemas. A model is published in a tree format and includes:

- Top level nodes and their subtrees
- Subtrees that augment nodes in other yang models

```
Example: The aaa Yang model
(exec-19.42.10) bash-4.2$ pyang -f tree Cisco-IOS-XR-aaa-lib-cfg.yang
module: Cisco-IOS-XR-aaa-lib-cfg
  +--rw aaa
    +--rw accountings
      | +--rw accounting* [type listname]
      |   +--rw type          xr:Cisco-ios-xr-string
      |   +--rw listname     xr:Cisco-ios-xr-string
      |   +--rw rp-failover? dt1:Aaa-accounting-rp-failover
      |   +--rw broadcast?  dt1:Aaa-accounting-broadcast
      |   +--rw type-xr?    dt1:Aaa-accounting
      |   +--rw method1?   dt1:Aaa-method-accounting
      |   +--rw method2?   dt1:Aaa-method-accounting
      |   +--rw method3?   dt1:Aaa-method-accounting
      |   +--rw method4?   dt1:Aaa-method-accounting
      |   +--rw server-group-name1? string
      |   +--rw server-group-name2? string
      |   +--rw server-group-name3? string
      |   +--rw server-group-name4? string
    +--rw authorizations
      | +--rw authorization* [type listname]
      |   +--rw type          xr:Cisco-ios-xr-string
      |   +--rw listname     xr:Cisco-ios-xr-string
      |   +--rw method1?    dt1:Aaa-method
      |   +--rw method2?    dt1:Aaa-method
      |   +--rw method3?    dt1:Aaa-method
      |   +--rw method4?    dt1:Aaa-method
      |   +--rw server-group-name1? string
      |   +--rw server-group-name2? string
      |   +--rw server-group-name3? string
      |   +--rw server-group-name4? string
    +--rw accounting-update!
      | +--rw type          dt1:Aaa-accounting-update
      | +--rw periodic-interval? uint32
```

```

+--rw banner
| +--rw login?  string
+--rw authentications
  +--rw authentication* [type listname]
    +--rw type                xr:Cisco-ios-xr-string
    +--rw listname            xr:Cisco-ios-xr-string
    +--rw method1?           dt1:Aaa-method
    +--rw method2?           dt1:Aaa-method
    +--rw method3?           dt1:Aaa-method
    +--rw method4?           dt1:Aaa-method
    +--rw server-group-name1? string
    +--rw server-group-name2? string
    +--rw server-group-name3? string
    +--rw server-group-name4? string

```

Advantages of using the Yang model are:

- Yang supports programmatic interfaces.
- Yang supports simplified network management applications.
- Yang supports interoperability that provides a standard way to model management data.

## Netconf and Yang

The workflow displayed here, will help the user to understand how Netconf-Yang can configure and control the network with minimal user intervention. The required components:

- Cisco 8000 Series Router with Netconf capability
- Netconf Client Application with connection to the router

S. No.	Device / component	Action
1	Cisco router	Login/ access the router.
2	Cisco router	Prerequisites for enabling Netconf: <ul style="list-style-type: none"> <li>• Crypto keys must be generated.</li> </ul>
3	Cisco router	Enable Netconf agent. Use the <b>netconf-yang agent ssh</b> and <b>ssh server netconf</b> command. The port can be selected. By default, it is set as 830.
4	Cisco router	Yang models are a part of the software image. The models can be retrieved from the router , using the <get-schema> operation.

S. No.	Device / component	Action
5	Netconf client (application) The application can be on any standalone application or a SDN controller supporting Netconf	Installs and processes the Yang models.  The client can offer a list of supported yang models; else the user will have to browse and locate the required yang file.  There is a yang model file for each configuration module; for instance if the user wants to configure CDP , the relevant yang model is Cisco-IOS-XR-cdp-cfg  <b>Note</b> Refer the table which lists all the supported yang models <a href="#">Supported Yang Models</a> , on page 132
5	Netconf client	Sends Netconf operation request over SSH to the router. A configuration request could include Yang-based XML data to the router. Currently, SSH is the only supported transport method.
6	Cisco router	Understands the Yang-based XML data and the network is configured accordingly (in case of configuration request from the client).
		The interactions between the client and the router happens until the network is configured as desired.

## Supported Yang Models

The Yang models can be downloaded from a prescribed location (ftp server) or can also be retrieved directly from the router using the get-schema operation.

For a feature, separate Yang models are available for configuring the feature and to get operational statistics (show commands). The **-cfg.yang** suffix denotes configuration and **-oper\*.yang** is for operational data statistics. In some cases, **-oper** is followed by **-sub**, indicating that a submodule(s) is available.

For a list of supported Yang models, see <https://github.com/YangModels/yang/tree/master/vendor/cisco/xr>

## Denial of Services Defense for Netconf-Yang

In case of a DoS (Denial of Service) attack on Netconf, wherein, Netconf receives numerous requests in a short span of time, the router may become irresponsive if Netconf consumes most of the bandwidth or CPU processing time. This can be prevented, by limiting the traffic directed at the Netconf agent. This is achieved using the **netconf-yang agent rate-limit** and **netconf-yang agent session** commands.

If rate-limit is set, the Netconf processor measures the incoming traffic from the SSH server. If the incoming traffic exceeds the set rate-limit, the packets are dropped.

If session-limit is set, the Netconf processor checks for the number of open sessions. If the number of current sessions is greater than or equal to, the set limit, no new sessions are opened.

Session idle-timeout and absolute-timeout also prevent DoS attacks. The Netconf processor closes the sessions, even without user input or intervention, as soon as the time out session is greater than or equal to the set time limit.

The relevant commands are discussed in detail, in the *System Security Command Reference for Cisco 8000 Series Routers*

## Enabling NETCONF over SSH

This task enables NETCONF over SSH. SSH is currently the only supported transport method .

If the client supports, Netconf over ssh can utilize the multi-channeling capabilities of IOS XR ssh server.

### Prerequisite:

- Crypto keys must be generated prior to this configuration.

---

### Step 1 **configure**

#### Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters mode.

### Step 2 **netconf-yang agent ssh**

#### Example:

```
RP/0/RP0/CPU0:router (config) # netconf agent ssh
```

Enables NETCONF agent over SSH connection. After NETCONF is enabled, the Yang model in the controller, can configure the relevant models.

**Note** The Yang models can be retrieved from the router via NETCONF <get-schema> operation.

### Step 3 **ssh server netconf [vrf vrf-name [ipv4 access-list ipv4 access list name] [ipv6 access-list ipv6 access list name] ]**

#### Example:

```
RP/0/RP0/CPU0:router (config) # ssh server netconf vrf netconfvrf ipv4 access-list InternetFilter
```

Brings up the netconf subsystem support with SSH server using a specified VRF of up to 32 characters. If no VRF is specified, the default VRF is used. To stop the SSH server from receiving any further connections for the specified VRF, use the **no** form of this command.

Optionally ACLs for IPv4 and IPv6 can be used to restrict access to the netconf subsystem of the ssh server before the port is opened.

**Note** The netconf subsystem support with SSH server can be configured for use with multiple VRFs .

### Step 4 **ssh server netconf port port-number**

#### Example:

```
RP/0/RP0/CPU0:router (config) # ssh server netconf port 830
```

Configures a port for the netconf ssh server. This command is optional. If no port is specified, port 830 is used by default.

**Note** 830 is the IANA-assigned TCP port for NETCONF over SSH, but it can be changed using this command.

### What to do next

The **show netconf-yang statistics** command and **show netconf-yang clients** command can be used to verify the configuration details of the netconf agent.

The **clear netconf-yang agent session** command clears the specified Netconf session (on the Netconf server side).

## Examples: Netconf over SSH

This section illustrates some examples relevant to Netconf:

### Enabling netconf-yang for ssh transport and netconf subsystem for default vrf with default port (830)

```
config
netconf-yang agent ssh
ssh server netconf vrf default
!
!
```

### Enabling netconf-yang for ssh transport and netconf subsystem for vrf *green* and vrf *red* with netconf port (831)

```
config
netconf-yang agent ssh
!
ssh server netconf vrf green
ssh server netconf vrf red
ssh server netconf port 831
!
!
```

### Show command outputs

```
show netconf-yang statistics
Summary statistics          requests|          total time|   min time per request|   max
time per request|   avg time per request|
other                    0|   0h 0m 0s 0ms|   0h 0m 0s 0ms|
  0h 0m 0s 0ms|   0h 0m 0s 0ms|
close-session            4|   0h 0m 0s 3ms|   0h 0m 0s 0ms|
  0h 0m 0s 1ms|   0h 0m 0s 0ms|
kill-session             0|   0h 0m 0s 0ms|   0h 0m 0s 0ms|
  0h 0m 0s 0ms|   0h 0m 0s 0ms|
get-schema               0|   0h 0m 0s 0ms|   0h 0m 0s 0ms|
  0h 0m 0s 0ms|   0h 0m 0s 0ms|
get                       0|   0h 0m 0s 0ms|   0h 0m 0s 0ms|
  0h 0m 0s 0ms|   0h 0m 0s 0ms|
get-config               1|   0h 0m 0s 1ms|   0h 0m 0s 1ms|
  0h 0m 0s 1ms|   0h 0m 0s 1ms|
edit-config              3|   0h 0m 0s 2ms|   0h 0m 0s 0ms|
  0h 0m 0s 1ms|   0h 0m 0s 0ms|
commit                   0|   0h 0m 0s 0ms|   0h 0m 0s 0ms|
  0h 0m 0s 0ms|   0h 0m 0s 0ms|
cancel-commit            0|   0h 0m 0s 0ms|   0h 0m 0s 0ms|
  0h 0m 0s 0ms|   0h 0m 0s 0ms|
```



```

lock
  0h 0m 0s 0ms|      0h 0m 0s 0ms|      0h 0m 0s 0ms|
unlock
  0h 0m 0s 0ms|      0h 0m 0s 0ms|      0h 0m 0s 0ms|
discard-changes
  0h 0m 0s 0ms|      0h 0m 0s 0ms|      0h 0m 0s 0ms|
validate
  0h 0m 0s 0ms|      0h 0m 0s 0ms|      0h 0m 0s 0ms|

show netconf-yang clients
client session ID|  NC version|  client connect time|  last OP time|  last
OP type|  <lock>|
22969|  1.1|  0d 0h 0m 2s|  11:11:24|
close-session|  No|
15389|  1.1|  0d 0h 0m 1s|  11:11:25|  get-config|
      No|

```





# CHAPTER 10

## Configuring PTP

**Table 29: Feature History Table**

Feature name	Release Information	Feature Description
Precision Time Protocol (PTP) Support on 88-LC0-36FH-M Line card and 8202-32FH-M Router	Release 7.5.2	<p>With this release, support for Precision Time Protocol (PTP) telecom profiles 8273.2 and 8275.1 is extended to the following:</p> <ul style="list-style-type: none"> <li>• 88-LC0-36FH-M line card</li> <li>• 8202-32FH-M router</li> </ul> <p>Support of PTP profile 8273.2 allows distribution of time and phase synchronization for packet-based network. Support of PTP profile 8275.1 enables network element interoperability for the delivery of accurate phase and time synchronization.</p>
Support for Precision Time Protocol (PTP)	Release 7.3.1	<p>Precision Time Protocol (PTP) is based on the IEEE 1588-2008 clock synchronization standard and enables clocks in a distributed system to be synched with highly precise clocks. The precision in time synchronization is achieved through packets that are transmitted and received in a session between the primary clock and secondary clock. PTP also ensures that the best clock is selected as a timing source (the primary clock) and all other clocks are synchronized with the primary clock.</p>

Precision Time Protocol (PTP) is a protocol that defines a method to distribute time around a network. PTP support is based on the IEEE 1588-2008 standard.

This module describes the concepts around this protocol and details the various configurations involved.

- [PTP Overview, on page 138](#)
- [Performance Monitoring for PTP Networks, on page 145](#)
- [ITU-T Telecom Profiles for PTP, on page 150](#)
- [ITU-T Telecom Profile Examples, on page 160](#)

## PTP Overview

The Precision Time Protocol (PTP), as defined in the IEEE 1588 standard, synchronizes with nanosecond accuracy the real-time clocks of the devices in a network. The clocks are organized into a master-slave hierarchy. PTP identifies the port that is connected to a device with the most precise clock. This clock is referred to as the master clock. All the other devices on the network synchronize their clocks with the master and are referred to as members. Constantly exchanged timing messages ensure continued synchronization. PTP ensures that the best available clock is selected as the source of time (the grandmaster clock) for the network and that other clocks in the network are synchronized to the grandmaster.

**Table 30: PTP Clocks**

Network Element	Description
Grandmaster (GM)	A network device physically attached to the primary time source. All clocks are synchronized to the grandmaster clock.
Ordinary Clock (OC)	An ordinary clock is a 1588 clock with a single PTP port that can operate in one of the following modes: <ul style="list-style-type: none"> <li>• Master mode—Distributes timing information over the network to one or more slave clocks, thus allowing the slave to synchronize its clock to the master.</li> <li>• Slave mode—Synchronizes its clock to a master clock. You can enable the slave mode on up to two interfaces simultaneously in order to connect to two different master clocks.</li> </ul>
Boundary Clock (BC)	The device participates in selecting the best master clock and can act as the master clock if no better clocks are detected.  Boundary clock starts its own PTP session with a number of downstream slaves. The boundary clock mitigates the number of network hops and packet delay variations in the packet network between the Grand Master and Slave.

Network Element	Description
Transparent Clock (TC)	A transparent clock is a device or a switch that calculates the time it requires to forward traffic and updates the PTP time correction field to account for the delay, making the device transparent in terms of time calculations.

PTP consists of two parts:

- The port State machine and Best Master Clock Algorithm: This provides a method to determine state of the ports in the network that will remain passive (neither master nor slave), run as a master (providing time to other clocks in the network), or run as slaves (receiving time from other clocks in the network).
- Delay-Request/Response mechanism and a Peer-delay mechanism: This provides a mechanisms for slave ports to calculate the difference between the time of their own clocks and the time of their master clock.




---

**Note** Transparent Clock (TC) is not supported.

---

### Frequency and Time Selection

The selection of the source to synchronize the device clock frequency is made by frequency synchronization, and is outside of the scope of PTP. The Announce, Sync, and Delay-request frequencies must be the same on the master and slave.

### Delay Request-Response Mechanism

The Delay Request-response mechanism (defined in section 11.3 of IEEE Std 1588-2008) lets a slave port estimate the difference between its own clock-time and the clock-time of its master. The following options are supported:

- One-step mechanism - The timestamp for a Sync message is sent in the Sync message itself.
- Two-step mechanism - The timestamp for a Sync message is sent later in a Follow-up message.

When running a port in Slave state, a router can send Delay-request messages and handle incoming Sync, Follow-up, and Delay-response messages. The timeout periods for both Sync and Delay-response messages are individually configurable.

### Hybrid Mode

Your router allows the ability to select separate sources for frequency and time-of-day (ToD). Frequency selection can be between any source of frequency available to the router, such as: BITS, GPS, SyncE or IEEE 1588 PTP. The ToD selection is between the source selected for frequency and PTP, if available (ToD selection is from GPS, or PTP). This is known as hybrid mode, where a physical frequency source (BITS or SyncE) is used to provide frequency synchronization, while PTP is used to provide ToD synchronization.

Frequency selection uses the algorithm described in ITU-T recommendation G.781. The ToD selection is controlled using the time-of-day priority configuration. This configuration is found under the clock interface frequency synchronization configuration mode and under the global PTP configuration mode. It controls the

order for which sources are selected for ToD. Values in the range of 1 to 254 are allowed, with lower numbers indicating higher priority.

The steps involved in section *Configuring PTP Hybrid Mode* of the topic [G.8275.1, on page 154](#).

### Time of Day (ToD) Support

The router receives GPS ToD messages in serial ASCII stream through the RS422 interface in one of the following configurable formats:

- NTP Type 4
- Cisco

### Port States for PTP

State machine indicates the behavior of each port. The possible states are:

State	Description
INIT	Port is not ready to participate in PTP.
LISTENING	First state when a port becomes ready to participate in PTP: In this state, the port listens to PTP masters for a (configurable) period of time.
PRE-MASTER	Port is ready to enter the MASTER state.
MASTER	Port provides timestamps for any Slave or boundary clocks that are listening.
UNCALIBRATED	Port receives timestamps from a Master clock but, the router's clock is not yet synchronized to the Master.
SLAVE	Port receives timestamps from a Master clock and the router's clock is synchronized to the Master.
PASSIVE	Port is aware of a better clock than the one it would advertise if it was in MASTER state and is not a Slave clock to that Master clock.

## Restrictions for PTP

The following PTP restrictions apply to the Cisco 8000 Series Router:

- Sync2 interface is supported only if 10 MHz, 1 Pulse per Second (PPS) and time-of-day (ToD) ports are configured.
- PTP is not supported with global MACSec.
- PTP is not supported with MACSec on the same interface.  
However, PTP is supported if MACSec is not configured on interface.
- PTP is not supported with global MACSec-FIPS-Post.

MACSec-FIPS-Post is not available per interface.

- Transparent Clock is not supported. One-step clock is supported. It can receive follow-up PTP packets, that is, it can support a two-step peer primary but it cannot send follow-up PTP packets.
- When a subinterface is configured with encapsulation default or untag configuration, you must configure PTP on that subinterface, instead of the main interface.
- PTP is configurable on Gigabit Ethernet interfaces (1G, 10G, 40G, and 100G), Bundle Ethernet interfaces, and sub-interfaces. PTP is not configurable on LAG Ethernet sub-interfaces.
- PTP is supported over individual bundle member links and not supported on Bundle-Ether interfaces.

## PTP Support Information

This table lists different types of support information related to PTP:

Transport Media	<ul style="list-style-type: none"> <li>• UDP over IPv4</li> <li>• UDP over IPv6</li> <li>• Ethernet</li> </ul>
Messages	<ul style="list-style-type: none"> <li>• Signaling</li> <li>• Announce</li> <li>• Sync</li> <li>• Follow-up</li> <li>• Delay-request</li> <li>• Delay-response</li> <li>• Management</li> </ul>
Transport Modes	<ul style="list-style-type: none"> <li>• Unicast: This is the default mode. All packets are sent as unicast messages. Unicast is applicable only for PTP over IP profiles.</li> <li>• Multicast: All packets are sent as multicast messages. Multicast is the only mode for PTP over ethernet profiles.</li> </ul>

## Timing Profile and Class Support Matrix

This table provides a detailed information on the timing features that are supported on the Cisco 8000 series routers and line cards.

Table 31: Timing Profile and Class Support Matrix

Hardware Module	Supported Profile	Supported G.8273.2 Class	Cisco IOS XR Release
<ul style="list-style-type: none"> <li>8000-RP2 Route Processor</li> <li>88-LC0-36FH-M and 8800-LC-36FH line cards</li> </ul>	G8275.1	NA	Release 7.11.1
	G8273.2	Class C	
<ul style="list-style-type: none"> <li>88-LC0-36FH-M line card</li> <li>8202-32FH-M router</li> </ul>	G.8273.2	Class C	Release 7.5.2
	G.8275.1	NA	
<ul style="list-style-type: none"> <li>88-LC0-36FH line card</li> <li>88-LC0-34H14FH line card</li> <li>8201-32FH router</li> </ul>	G.8273.2	Class C	Release 7.3.3
	G.8275.1	NA	
<ul style="list-style-type: none"> <li>8201 router</li> <li>8202 router</li> <li>8800-LC-48H line card</li> <li>8800-LC-36FH line card</li> </ul>	G.8273.2	Class C	Release 7.3.1
	G.8275.1	NA	
	G.8265.1	NA	
	G.8263	NA	

## Configuring PTP Delay Asymmetry

Table 32: Feature History Table

Feature Name	Release Information	Description
PTP Delay Asymmetry	Release 7.3.2	<p>Any delays on Precision Time Protocol (PTP) paths can impact PTP accuracy and in turn impact clock settings for all devices in a network. This feature allows you to configure the static asymmetry such that the delay is accounted for and the PTP synchronization remains accurate.</p> <p>The <b>delay-symmetry</b> command is introduced for this feature.</p>



Configure PTP delay asymmetry to offset the static delays on a PTP path that occur due to different route selection for forward and reverse PTP traffic. Delays can also be due to any node having different delay for ingress or egress path. These delays can impact PTP accuracy due to the asymmetry in PTP. With this feature, you can enable a higher degree of accuracy in the PTP server performance leading to better synchronization between real-time clocks of the devices in a network.

Configuration of this delay asymmetry provides an option to configure static delays on a client clock for every server clock. You can configure this delay value in microseconds and nanoseconds. Configured PTP delay asymmetry is also synchronized with the Servo algorithm.

**Note**

- If you configure multiple PTP delay asymmetries for the same PTP profile, the latest PTP delay asymmetry that you configure is applied to the PTP profile.
- For G8275.1 and G8275.2 PTP profiles, PTP delay asymmetry is supported for both, client port and dynamic port that act as a client.
- Fixed delay can be measured by using any test and measurement tool. Fixed delay can be compensated by using the positive or negative values. For example, if the fixed delay is +10 nanoseconds, configure -10 nanoseconds to compensate the fixed delay.

A positive value indicates that the server-to-client propagation time is longer than the client-to-server propagation time, and conversely for negative values.

**Supported PTP Profiles**

The following PTP profiles support the configuration of PTP delay asymmetry:

- PTP over IP (G8275.2 or default profile)
- PTP over L2 (G8275.1)

**Restrictions**

- PTP delay asymmetry can be configured only on the PTP port of the grandmaster clock, which can either be a boundary clock or an ordinary clock.
- PTP delay asymmetry is supported for delay compensation of fixed cables and not for variable delay in the network.
- PTP delay asymmetry can be configured within the range of 3 microseconds and -3 microseconds or 3000 nanoseconds and -3000 nanoseconds.

**Configuration**

To configure PTP delay asymmetry:

1. Configure an interface with PTP.
2. Configure PTP delay asymmetry on the client side.

### Configuration Example

```

/* Configure an interface with PTP. */
Router# configure
Router(config)# interface HundredGigE 0/1/0/0
Router(config-if)# ptp

/* Configure PTP delay asymmetry on the client side. */
Router(config-if-ptp)# delay-asymmetry 3 microseconds
Router(config-if-ptp)# commit

```

### Running Configuration

```

interface preconfigure HundredGigE 0/1/0/0
  ptp
  delay-asymmetry 3 microseconds

```

### Verification

To verify if PTP delay asymmetry is applied, use the **show ptp foreign-masters** command:

```

Router# show ptp foreign-masters
Sun Nov 1 10:19:21.874 UTC
Interface HundredGigE0/1/0/0 (PTP port number 1)
IPv4, Address 209.165.200.225, Unicast
Configured priority: 1
Configured clock class: None
Configured delay asymmetry: 3 microseconds <- configured variable delay asymmetry value
Announce granted: every 2 seconds, 300 seconds
Sync granted: 16 per-second, 300 seconds
Delay-resp granted: 16 per-second, 300 seconds
Qualified for 2 minutes, 45 seconds
Clock ID: 80e01dffffe8ab73f
Received clock properties:
Domain: 0, Priority1: 128, Priority2: 128, Class: 6
Accuracy: 0x22, Offset scaled log variance: 0xcd70
Steps-removed: 1, Time source: GPS, Timescale: PTP
Frequency-traceable, Time-traceable
Current UTC offset: 37 seconds (valid)
Parent properties:
Clock ID: 80e01dffffe8ab73f
Port number: 1

```

To validate the approximate compensated delay value, use the **show ptp platform servo** command:

```

Router# show ptp platform servo
Mon Jun 27 22:32:44.912 UTC
Servo status: Running
Servo stat_index: 2
Device status: PHASE_LOCKED
Servo Mode: Hybrid
Servo log level: 0
Phase Alignment Accuracy: -2 ns
Sync timestamp updated: 18838
Sync timestamp discarded: 0
Delay timestamp updated: 18837
Delay timestamp discarded: 0
Previous Received Timestamp T1: 1657002314.031435081 T2: 1657002314.031436686 T3:
1657002314.026815770 T4: 1657002314.026814372
Last Received Timestamp T1: 1657002314.031435081 T2: 1657002314.031436686 T3:
1657002314.088857790 T4: 1657002314.088856392
Offset from master: 0 secs, 1502 nsecs <<--compensated value shows 1.5 microseconds
because the asymmetry configured under the interface is
3 microseconds.->>

```

```

Mean path delay      : 0 secs, 103 nsecs
setTime():0  stepTime():0  adjustFreq():2
Last setTime: 0.000000000  flag:0  Last stepTime:0  Last adjustFreq:-5093

```

## Performance Monitoring for PTP Networks

**Table 33: Feature History Table**

Feature Name	Release Information	Feature Description
Performance Monitoring	Release 24.3.1	<p>Introduced in this release on: Fixed Systems (8200); Centralized Systems (8600); and Modular Systems (8800[LC ASIC: Q100, Q200, P100])</p> <p>You can now get statistical information with Performance Monitoring in PTP networks, such as clock accuracy, synchronization status, and network delays by defining Performance Monitoring Parameters and Port Specific Parameters.</p> <p>This feature empowers operators with comprehensive performance monitoring and precise time-stamp analysis, offering enhanced granularity for time synchronization in telecommunication networks. By providing detailed insights, it enables operators to make well-informed decisions and take proactive actions to ensure optimal network performance.</p> <p>The feature introduces these changes:</p> <p><b>CLI:</b></p> <ul style="list-style-type: none"> <li>• <a href="#">performance-monitoring</a></li> <li>• <a href="#">show ptp platform performance-counters</a></li> <li>• <a href="#">show ptp dataset performance</a></li> </ul> <p><b>YANG Data Models:</b></p> <ul style="list-style-type: none"> <li>• <a href="#">Cisco-IOS-XR-ptp-cfg.yang</a></li> <li>• <a href="#">Cisco-IOS-XR-ptp-oper.yang</a></li> <li>• <a href="#">Cisco-IOS-XR-um-ptp-cfg.yang</a></li> </ul> <p>(see <a href="#">GitHub</a>, <a href="#">YANG Data Models Navigator</a>)</p>

Performance Monitoring in PTP involves tracking and analyzing the performance of PTP networks to ensure accurate time synchronization across devices. This includes monitoring various metrics such as clock accuracy, synchronization status, and network delays. The goal is to identify and address any issues that may affect the precision and reliability of time synchronization in the network.

Performance Monitoring now has the ability to provide performance monitoring and time-stamp analysis information in a PTP network as per Annex J IEEE 1588-2019. This feature also includes additional monitoring

granularity for time synchronization in telecommunication networks as per Annex F from the G8275 standard. For more information on PTP, Best TimeTransmitter Clock Algorithm (BTCA), see [PTP Overview](#).

You can use the following parameters to define the Performance Monitoring in a PTP Network:

- Performance Monitoring Parameters
- Port Specific Parameters

### Performance Monitoring Parameters

In addition to using the timestamps received from the grandmaster to sync to the grandmaster's clock, the timestamps can now be used to calculate parameters that are of your interest in performance monitoring:

- **TimeTransmitter - TimeReceiver Delay**: corrected propagation time from TimeTransmitter to TimeReceiver.
- **TimeReceiver - TimeTransmitter Delay**: corrected propagation time from TimeReceiver to TimeTransmitter.
- **Mean Path Delay**: mean propagation time over the PTP Communication Path.
- **Offset from TimeTransmitter**: time difference between a TimeTransmitter PTP instance and a TimeReceiver PTP instance as computed by the TimeReceiver PTP instance.

For each of these parameters, you can measure the average, minimum, maximum, and standard deviation for each measurement. These values are calculated and maintained for the following time intervals over the specified time periods:

- **3-minute**: maintained for the current 1-hour period.
- **15-minute**: maintained for the current 24-hour period.
- **1-hour**: maintained for the current 2-hour period.
- **24-hour**: maintained for the current 48-hour period.

The platform actively calculates the end-to-end latency between the TimeTransmitter and TimeReceiver through the Delay-Request-Response-Mechanism (DRRM), allowing Precision Time Protocol (PTP) to seamlessly operate across networks equipped with Transparent clocks, non-PTP aware switches, or a mix of both. Upon a request, PTP dynamically extracts these calculated values from the servo using a platform specific API, allowing you to make proactive changes to the network to ensure precise time synchronization essential for applications that depend on accurate timing.

### Additional Port Specific Parameters

The parameters mentioned above apply to the entire Precision Time Protocol (PTP) instance, and there is an additional set of parameters specific to individual ports. These parameters include the counters for various packet types,

- received (rx) and
- transmitted (tx)

It is important to collect and maintain these counters for performance monitoring purposes, which follows the same time intervals and periods as those used for monitoring clock performance.

Port-specific parameters play a crucial role in ensuring accurate time synchronization. These packet types are essential for maintaining the accuracy and reliability of time synchronization in a PTP network:

- **Sync Packets:** These packets are sent by the master clock to the slave clocks to synchronize their time. They contain the precise time at which the packet was sent.
- **Delay Request Packets:** Sent by the slave clock to the master clock, these packets measure the delay between the master and slave clocks. The master clock responds with a Delay Response packet.
- **Follow-Up Packets:** These packets are sent by the master clock immediately after the Sync packet. They contain the exact time the Sync packet was sent, which helps in correcting any delays introduced by the network.
- **Announce Packets:** These packets are used by the master clock to announce its presence and capabilities to the slave clocks. They help in the selection of the best master clock in the network.
- **Management Packets:** These packets are used for configuration and management purposes within the PTP network. They allow for the adjustment of various parameters and settings.

### Record format

Record format refers to the structure or layout of data within a record, which is used to store information about time synchronization events and measurements. This format can include various fields such as timestamps, event types, and other relevant data that PTP uses to maintain accurate time synchronization across a network. It is a single buffer for both annexes.

The format is the same for both clock and port performance monitoring parameters that is presented in the operational data. The data is stored over a 48-hour period, resulting in a list of records as per Annex J 1588-2019, composed of the following:

- 1 record for the current 15-minute set of statistics (stored at position 0 in the buffer).
- 96 records for the 15-minute sets of statistics over the last 24-hour period (stored between positions 1-96 in the buffer).
- 1 record for the current 24-hour set of statistics (stored at position 97 in the buffer).
- 1 record for the previous 24-hour set of statistics (stored at position 98 in the buffer).

The data buffer records data at 3-minute intervals over the most recent 1-hour period, creating a list of records that includes:

- 1 record for the current 3-minute set of statistics (stored at position 100 in the buffer).
- 20 records for the 3-minute set of statistics over the last 1-hour period (stored between positions 101-120).
- 1 record for the current 1-hour set of statistics (stored at position 121 in the buffer).
- 1 record for the previous 1-hour set of statistics (stored at position 122 in the buffer).

## Configure PTP Performance Monitoring

The purpose of this task is to configure and verify PTP performance monitoring.

**Step 1** Configure the **performance-monitoring** command to enable collection of performance monitoring statistics and for the users to make performance monitoring requests.

**Example:**

```
Router(config)# ptp
Router(config-ptp)# performance-monitoring
Router(config-ptp)# commit
```

**Step 2** Run the **sh ptp platform performance-counters** command to display the details of all 123 records.

The existing command **show ptp platform** is extended to include the performance monitoring data for the local clock. The detail mode of the command displays all 123 records while the brief mode displays only the current windows for 15 minutes, 24 hours, 3minutes, and 1hour.

**Example:**

```
Router#sh ptp platform performance-counters detail

PTP Current record index 15 min: 96
PTP Current record index 3 min: 119

PTP performance monitoring statistics:
=====
15 min stats
[0] 12 August 2024 07:08:59 UTC 15 min statistics
-----
```

Stat Samples	Min(sec.nsec)	Max(sec.nsec)	Mean(sec.nsec)	Std deviation
Master-slave-delay 154	-000000000.15937	000000000.333	-000000000.1780	000000000.71191
Slave-master-delay 154	000000000.319	000000000.16593	000000000.2437	000000000.74103
mean-path-delay 154	000000000.322	000000000.334	000000000.327	000000000.4057
offset-from-master 154	-000000000.16263	000000000.6	-000000000.2108	000000000.72546

```
-----
Complete      Valid      PmRef      ServoAtStart  ServoAtEnd      LastServoFlapTime
-----
07:09:09 UTC  FALSE     FALSE     TRUE         PHASE_LOCKED   HOLDOVER         12 Apr 2024
=====
....
```

**Step 3** Run the **show ptp dataset performance clock** command to display the performance monitoring data-set details in 15 minutes intervals.

**Example:**

```
Router#show ptp dataset performance clock
```

```

performanceMonitoringDS for the current 15-minute window:
Clock ID cccfffecccc00, steps removed 1, receiving-port 2:
  Start of time window: Thursday, April 11, 2024 14:18:59
  Measurement is valid
  Period is complete
  Measurement has been taken with reference to system clock
  Master slave delay:
    Average: 50ns
    Min: 50ns
    Max: 70ns
    Std: 1ns
  Slave master delay:
    Average: 51ns
    Min: 51ns
    Max: 71ns
    Std: 2ns
  Mean path delay:
    Average: 52ns
    Min: 52ns
    Max: 72ns
    Std: 3ns
  Offset from master:
    Average: 53ns
    Min: 53ns
    Max: 73ns
    Std: 4ns

```

```

Clock ID aaaabbbbecccc00, steps removed 1, receiving-port 2:
  Start of time window: Thursday, April 11, 2024 14:18:59
  Measurement is not valid
  Period is not complete
  Measurement has been taken with reference to system clock
  Master slave delay:
    Average: 50ns
    Min: 50ns
    Max: 70ns
    Std: 1ns
  Slave master delay:
    Average: 51ns
    Min: 51ns
    Max: 71ns
    Std: 2ns
  Mean path delay:
    Average: 52ns
    Min: 52ns
    Max: 72ns
    Std: 3ns
  Offset from master:
    Average: 53ns
    Min: 53ns
    Max: 73ns
    Std: 4ns

```

**Step 4** Run the **show ptp dataset performance port** to display the Performance Monitoring Port Data-set in 15 minutes intervals.

**Example:**

```
Router#show ptp dataset performance port GigabitEthernet 0/0/0/1
```

```

performanceMonitoringPortDS for the current 15-minute window:
Interface GigabitEthernet 0/0/0/1
  Start of time window: Thursday, April 11, 2024 14:18:59
  Measurement is valid
  Period is not complete
  Measurement has been taken with reference to system clock

```

Packets	Sent	Received	Dropped
Announce	3	83	11
Sync	0	32	5
Follow-Up	0	31	0
Delay-Req	22	0	0
Delay-Resp	0	21	7
Pdelay-Req	0	7	0
Pdelay-Resp	0	0	0
Pdelay-Resp-Follow-Up	0	0	0
Signaling	2	1	0
Management	0	0	0
Other	0	3	12
TOTAL	27	178	35

## ITU-T Telecom Profiles for PTP

Cisco IOS XR software supports ITU-T Telecom Profiles for PTP as defined in the ITU-T recommendations. A profile is a specific selection of PTP configuration options that are selected to meet the requirements of a particular application.

PTP lets you define separate profiles to adapt itself for use in different scenarios. A telecom profile differs in several ways from the default behavior defined in the IEEE 1588-2008 standard and the key differences are mentioned in the subsequent sections.

The following sections describe the ITU-T Telecom Profiles that are supported for PTP.

### G.8265.1

G.8265.1 profile fulfills specific frequency-distribution requirements in telecom networks. Features of G.8265.1 profile are:

- Clock advertisement: G.8265.1 profile specifies changes to values used in Announce messages for advertising PTP clocks. The clock class value is used to advertise the quality level of the clock, while the other values are not used.
- Clock Selection: G.8265.1 profile also defines an alternate Best Master Clock Algorithm (BMCA) to select port states and clocks is defined for the profile. This profile also requires to receive Sync messages (and optionally, Delay-Response messages) to qualify a clock for selection.
- Port State Decision: The ports are statically configured to be Master or Slave instead of using state machines to dynamically set port states.
- Packet Rates: The packet rates higher than rates specified in the IEEE 1588-2008 standard are used. They are:
  - Sync/Follow-Up Packets: Rates from 128 packets-per-second to 16 seconds-per-packet.
  - Delay-Request/Delay-Response Packets: Rates from 128 packets-per-second to 16 seconds-per-packet.
  - Announce Packets: Rates from 8 packets-per-second to 64 packets-per-second.



- Transport Mechanism: G.8265.1 profile only supports IPv4 PTP transport mechanism.
- Mode: G.8265.1 profile supports transport of data packets only in unicast mode.
- Clock Type: G.8265.1 profile only supports Ordinary Clock-type (a clock with only one PTP port).
- Domain Numbers: The domain numbers that can be used in a G.8265.1 profile network ranges from 4 to 23.
- Port Numbers: All PTP port numbers can only be one (1) because all clocks in this profile network are Ordinary Clocks.
- G.8261 class-specification standard is supported.

G.8265.1 profile defines an alternate algorithm to select between different master clocks based on the local priority given to each master clock and their quality levels (QL). This profile also defines Packet Timing Signal Fail (PTSF) conditions to identify the master clocks that do not qualify for selection. They are:

- PTSF-lossSync condition: Raised for master clocks that do not receive a reliable stream of Sync and Delay-Resp messages. Cisco IOS XR software requests Sync and Delay-Resp grants for each configured master clock to track the master clock with this condition.
- PTSF-lossAnnounce condition: Raised for master clocks that do not receive a reliable stream of Announce messages.
- PTSF-unusable condition: Raised for master clocks that receives a reliable stream of Announce, Sync, and Delay-Resp messages, but not usable by slave clocks. Cisco IOS XR software does not use this condition.

### Configuring Global G.8265.1 Master Profile

The following configuration describes the steps involved to create a global configuration profile for a PTP interface that can then be assigned to any interface as required. It uses G.8265.1 profile as an example:

```
Router# config terminal
Router(config)# ptp
Router(config-ptp)# clock
Router(config-ptp-clock)# domain 4
Router(config-ptp-clock)# profile g.8265.1 clock-type master
Router(config-ptp-clock)# exit
Router(config-ptp)# profile master
Router(config-ptp-profile)# transport ipv4
Router(config-ptp-profile)# sync frequency 32
Router(config-ptp-profile)# announce frequency 1
Router(config-ptp-profile)# delay-request frequency 32
Router(config-ptp-profile)# exit
```

### Verification

To display the configured PTP profile details, use **show run ptp** command.

```
Router# show run ptp

Wed Feb 28 11:16:05.943 UTC
ptp
clock domain 4
profile g.8265.1 clock-type master
!
profile master
```

```

transport ipv4
sync frequency 32
announce frequency 1
delay-request frequency 32
!
```

### Configuring Global G.8265.1 Slave Profile

The following configuration describes the steps involved to create a global configuration profile for a PTP interface that can then be assigned to any interface as required. It uses G.8265.1 profile as an example:

```

Router/# config terminal
Router(config)# ptp
Router(config-ptp)# clock
Router(config-ptp-clock)# domain 4
Router(config-ptp-clock)# profile g.8265.1 clock-type slave
Router(config-ptp-clock)# exit
Router(config-ptp)# profile slave
Router(config-ptp-profile)# transport ipv4
Router(config-ptp-profile)# sync frequency 32
Router(config-ptp-profile)# announce frequency 1
Router(config-ptp-profile)# delay-request frequency 32
Router(config-ptp-profile)# exit
```

### Verification

To display the configured PTP profile details, use **show run ptp** command.

```

Router# show run ptp

Wed Feb 28 11:16:05.943 UTC
ptp
clock domain 4
profile g.8265.1 clock-type slave
!
profile slave
transport ipv4
sync frequency 32
announce frequency 1
delay-request frequency 32
!
```

### Configuring PTP Master Interface

The following configuration describes the steps involved to configure a PTP interface to be a Master.

```

Router# configure terminal
Router(config)# interface HundredGigE 0/0/0/0
Router(config-if)# ipv4 address 18.1.1.1/24
Router(config-if)# ptp
Router(config-if-ptp)# profile master
Router(config-if-ptp)# port state master-only
Router(config-if-ptp)# commit
```

## G.8263 Standard

**Table 34: Feature History Table**

Feature Name	Release Information	Feature Description
Support for ITU-T G.8263 standard for secondary clock with ITU-T G.8265.1 profile	Release 7.3.1	ITU-T G.8263 is a performance compliance standard for secondary clocks configured with ITU-T G.8265.1 profiles. These clocks drive frequency synchronization based on the PTP packets received at the secondary devices, from traceable primary devices.

G.8263 is the performance compliance standard for the clocks with G.8265.1 profile configured. These clocks drive frequency synchronization based on the PTP packets that are received at the secondary devices from traceable primary device. To handle excess PDV in the network, special servo mode is enabled by configuring the **network-type high-pdv** command in the PTP configuration.

### Configuration

To configure high PDV mode on the slave clock, use the following steps:

1. Configure telecom profile G.8265.1 and clock-type as slave.
2. Configure network type as high PDV.

### Configuration Example

```

/* Configure telecom profile G.8265.1 and clock-type as slave. */
Router# configure
Router(config)# ptp
Router(config-ptp)# clock
Router(config-ptp-clock)# domain 4
Router(config-ptp-clock)# profile g.8265.1 clock-type slave
Router(config-ptp-clock)# commit
Router(config-ptp-clock)# exit

/* Configure network type as high PDV. */
Router(config-ptp)# network-type high-pdv
Router(config-ptp)# commit

```

### Running Configuration

```

ptp
 clock
  domain 4
  profile g.8265.1 clock-type slave
 !
 network-type high-pdv
 !
 !

```

## G.8275.1

*Table 35: Feature History Table*

Feature Name	Release Information	Feature Description
ITU-T G.8275.1 profile support	Release 7.3.1	This feature supports the architecture defined in ITU-T G.8275 for systems requiring accurate phase and time synchronisation, phase or time-of-day synchronization is required, and where each network device participates in the PTP protocol. Support of this capability is extended on the Cisco 8000 Series router, in this release.

G.8275.1 profile fulfills the time-of-day and phase synchronization requirements in telecom networks with all network devices participating in the PTP protocol. G.8275.1 profile provides better frequency stability for the time-of-day and phase synchronization.

Features of G.8275.1 profile are:

- Synchronization Model: G.8275.1 profile adopts hop-by-hop synchronization model. Each network device in the path from master to slave synchronizes its local clock to upstream devices and provides synchronization to downstream devices.
- Clock Selection: G.8275.1 profile also defines an alternate BMCA that selects a clock for synchronization and port state for the local ports of all devices in the network is defined for the profile. The parameters defined as a part of the BMCA are:
  - Clock Class
  - Clock Accuracy
  - Offset Scaled Log Variance
  - Priority 2
  - Clock Identity
  - Steps Removed
  - Port Identity
  - notSlave flag
  - Local Priority
- Port State Decision: The port states are selected based on the alternate BMCA algorithm. A port is configured to a master-only port state to enforce the port to be a master for multicast transport mode.
- Packet Rates: The nominal packet rate for Announce packets is 8 packets-per-second and 16 packets-per-second for Sync/Follow-Up and Delay-Request/Delay-Response packets.
- Transport Mechanism: G.8275.1 profile only supports Ethernet PTP transport mechanism.

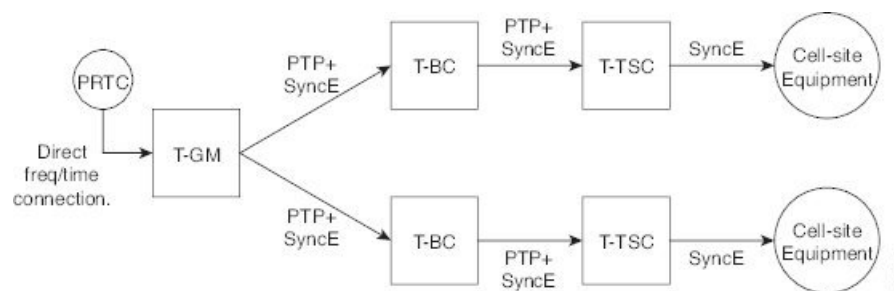
- Mode: G.8275.1 profile supports transport of data packets only in multicast mode. The forwarding is done based on forwardable or non-forwardable multicast MAC address.
- Clock Type: G.8275.1 profile supports the following clock types:
  - Telecom Grandmaster (T-GM): Provides timing for other network devices and does not synchronize its local clock to other network devices.
  - Telecom Time Slave Clock (T-TSC): A slave clock synchronizes its local clock to another PTP clock, but does not provide PTP synchronization to any other network devices.
  - Telecom Boundary Clock (T-BC): Synchronizes its local clock to a T-GM or an upstream T-BC clock and provides timing information to downstream T-BC or T-TSC clocks.
- Domain Numbers: The domain numbers that can be used in a G.8275.1 profile network ranges from 24 to 43. The default domain number is 24.

The G.8275.1 supports the following:

- T-GM: The telecom grandmaster (T-GM) provides timing to all other devices on the network. It does not synchronize its local clock with any other network element other than the Primary Reference Time Clock (PRTC).
- T-BC: The telecom boundary clock (T-BC) synchronizes its local clock to a T-GM or an upstream T-BC, and provides timing information to downstream T-BCs or T-TSCs. If at a given point in time there are no higher-quality clocks available to a T-BC to synchronize to, it may act as a grandmaster.
- T-TSC: The telecom time slave clock (T-TSC) synchronizes its local clock to another PTP clock (in most cases, the T-BC), and does not provide synchronization through PTP to any other device.

The following figure describes a sample G.8275.1 topology.

**Figure 6: A Sample G.8275.1 Topology**



### Configuring Global G.8275.1 Profile

The following configuration describes the steps involved to create a global PTP configuration profile that can be applied at an interface level. It uses G.8275.1 profile as an example:

```
Router# config terminal
Router(config)# ptp
Router(config-ptp)# clock
Router(config-ptp-clock)# domain 24
Router(config-ptp-clock)# profile g.8275.1 clock-type T-BC
Router(config-ptp-clock)# exit
Router(config-ptp)# profile slave
```

```

Router(config-ptp-profile)# multicast target-address ethernet 01-1B-19-00-00-00
Router(config-ptp-profile)# transport ethernet
Router(config-ptp-profile)# sync frequency 16
Router(config-ptp-profile)# announce frequency 8
Router(config-ptp-profile)# delay-request frequency 16
Router(config-ptp-profile)# exit
Router(config-ptp)# profile master
Router(config-ptp-profile)# multicast target-address ethernet 01-1B-19-00-00-00
Router(config-ptp-profile)# transport ethernet
Router(config-ptp-profile)# sync frequency 16
Router(config-ptp-profile)# announce frequency 8
Router(config-ptp-profile)# delay-request frequency 16
Router(config-ptp-profile)# exit
Router(config-ptp)# physical-layer-frequency
Router(config-ptp)# log
Router(config-ptp-log)# servo events
Router(config-ptp-log)# commit

```

### Verification

To display the configured PTP profile details, use **show run ptp** command.

```

Router# show run ptp

Wed Feb 28 11:16:05.943 UTC
ptp
 clock
  domain 24
  profile g.8275.1 clock-type T-BC
 !
profile slave
 multicast target-address ethernet 01-1B-19-00-00-00
 transport ethernet
 sync frequency 16
 announce frequency 8
 delay-request frequency 16
 !
profile master
 multicast target-address ethernet 01-1B-19-00-00-00
 transport ethernet
 sync frequency 16
 announce frequency 8
 delay-request frequency 16
 !
physical-layer-frequency
log
 servo events
 !

```

### Configuring PTP Master Interface

The below configuration describes the steps involved to configure a PTP interface to be a Master.

```

Router# configure terminal
Router(config)# interface HundredGigE0/0/0/0
Router(config-if)# ptp
Router(config-if-ptp)# profile master
Router(config-if-ptp)# port state master-only
Router(config-if-ptp)# commit

```

### Verification

To verify the port state details, use **show run interface** *interface-name* command.

```
Router# show run interface HundredGigE0/0/0/0
interface HundredGigE0/0/0/0
  ptp
  profile master
  port state master-only
!
```

### Configuring PTP Slave Interface

This procedure describes the steps involved to configure a PTP interface to be a Slave.

```
Router# configure terminal
Router(config)# interface HundredGigE0/0/0/1
Router(config-if)# ptp
Router(config-if-ptp)# profile slave
Router(config-if-ptp)# port state slave-only
Router(config-if-ptp)# commit
```

### Verification

To verify the port state details, use **show run interface** *interface-name* command.

```
Router# show run interface HundredGigE0/0/0/1
interface HundredGigE0/0/0/1
  ptp
  profile slave
  port state slave-only
!
```

### Configuring PTP Hybrid Mode

This procedure describes the steps involved to configure router in a hybrid mode. You configure hybrid mode by selecting PTP for phase and time-of-day (ToD) and another source for the frequency.



- 
- Note**
- G.8275.1 PTP profile supports only the hybrid mode. It is mandatory to have hybrid mode for G8275.1 profile for T-BC and T-TSC clock types. By default, the hybrid mode is used, regardless of the physical-layer-frequency configuration.
- 

To configure PTP Hybrid mode:

#### 1. Configure Global Frequency Synchronization

```
Router(config)# frequency synchronization
Router(config)# commit
```

#### 2. Configure Frequency Synchronization for an Interface. The time-of-day-priority setting specifies that SyncE to be used as a ToD source if there is no source available with a lower priority.

```
Router(config)# interface HundredGigE 0/0/0/0
Router(config-if)# frequency synchronization
Router(config-if-freqsync)# selection input
Router(config-if-freqsync)# time-of-day-priority 100
Router(config-if-freqsync)# commit
```

## Verification

```
Router # show frequency synchronization selection location 0/RP0/CP0
```

```
Tue Feb  6 06:34:17.627 UTC
Node 0/RP0/CPU0:
=====
Selection point: T0-SEL-B (3 inputs, 1 selected)
Last programmed 00:01:04 ago, and selection made 00:00:24 ago
Next selection points
SPA scoped : None
Node scoped : CHASSIS-TOD-SEL
Chassis scoped: LC_TX_SELECT
Router scoped : None
Uses frequency selection
Used for local line interface output
S Input Last Selection Point QL Pri Status
== =====
1 HundredGigE 0/0/0/0 0/2/CPU0 ETH_RXMUX 1 ePRTC 1 Locked
PTP [0/RP0/CPU0] n/a PRS 254 Available
Internal0 [0/RP0/CPU0] n/a ST3E 255 Available

Selection point: T4-SEL (3 inputs, 1 selected)
Last programmed 00:01:04 ago, and selection made 00:00:24 ago
Next selection points
SPA scoped : None
Node scoped : None
Chassis scoped: None
Router scoped : None
Uses frequency selection
Used for local clock interface output
S Input Last Selection Point QL Pri Status
== =====
1 HundredGigE 0/0/0/0 0/2/CPU0 ETH_RXMUX 1 ePRTC 1 Locked
PTP [0/RP0/CPU0] n/a PRS 254 Available
Internal0 [0/RP0/CPU0] n/a ST3E 255 Available

Selection point: 1588-SEL (2 inputs, 1 selected)
Last programmed 00:01:04 ago, and selection made 00:00:24 ago
Next selection points
SPA scoped : None
Node scoped : None
Chassis scoped: None
Router scoped : None
Uses frequency selection
S Input Last Selection Point QL Pri Status
== =====
1 HundredGigE 0/0/0/0 0/2/CPU0 ETH_RXMUX 1 ePRTC 1 Locked
Internal0 [0/RP0/CPU0] n/a ST3E 255 Available

Selection point: CHASSIS-TOD-SEL (2 inputs, 1 selected)
Last programmed 00:00:53 ago, and selection made 00:00:51 ago
Next selection points
SPA scoped : None
Node scoped : None
Chassis scoped: None
Router scoped : None
Uses time-of-day selection
S Input Last Selection Point Pri Time Status
== =====
1 PTP [0/RP0/CPU0] n/a 100 Yes Available
HundredGigE 0/0/0/0 0/RP0/CPU0 T0-SEL-B 1 100 No Available

RP0/RP0/CPU0:SF-D#
```



```

RP/0/RP0/CPU0:SF-D#
RP/0/RP0/CPU0:SF-D#show frequency synchronization selection location 0/RP0/CP$
Thu Jan 1 00:16:56.105 UTC
Node 0/RP0/CPU0:
=====
Selection point: T0-SEL-B (3 inputs, 1 selected)
Last programmed 00:01:09 ago, and selection made 00:00:29 ago
Next selection points
SPA scoped : None
Node scoped : CHASSIS-TOD-SEL
Chassis scoped: LC_TX_SELECT
Router scoped : None
Uses frequency selection
Used for local line interface output
S Input Last Selection Point QL Pri Status
== =====
1 HundredGigE 0/0/0/0 0/2/CPU0 ETH_RXMUX 1 ePRTC 1 Locked
PTP [0/RP0/CPU0] n/a PRS 254 Available
Internal0 [0/RP0/CPU0] n/a ST3E 255 Available

Selection point: T4-SEL (3 inputs, 1 selected)
Last programmed 00:01:09 ago, and selection made 00:00:29 ago
Next selection points
SPA scoped : None
Node scoped : None
Chassis scoped: None
Router scoped : None
Uses frequency selection
Used for local clock interface output
S Input Last Selection Point QL Pri Status
== =====
1 HundredGigE 0/0/0/0 0/2/CPU0 ETH_RXMUX 1 ePRTC 1 Locked
PTP [0/RP0/CPU0] n/a PRS 254 Available
Internal0 [0/RP0/CPU0] n/a ST3E 255 Available

Selection point: 1588-SEL (2 inputs, 1 selected)
Last programmed 00:01:09 ago, and selection made 00:00:29 ago
Next selection points
SPA scoped : None
Node scoped : None
Chassis scoped: None
Router scoped : None
Uses frequency selection
S Input Last Selection Point QL Pri Status
== =====
1 HundredGigE 0/0/0/0 0/2/CPU0 ETH_RXMUX 1 ePRTC 1 Locked
Internal0 [0/RP0/CPU0] n/a ST3E 255 Available

Selection point: CHASSIS-TOD-SEL (2 inputs, 1 selected)
Last programmed 00:00:57 ago, and selection made 00:00:56 ago
Next selection points
SPA scoped : None
Node scoped : None
Chassis scoped: None
Router scoped : None
Uses time-of-day selection
S Input Last Selection Point Pri Time Status
== =====
1 PTP [0/RP0/CPU0] n/a 100 Yes Available
HundredGigE 0/0/0/0 0/RP0/CPU0 T0-SEL-B 1 100 No Available

```

# ITU-T Telecom Profile Examples:

## G.8265.1 Profile Configuration Examples

### Master Global Configuration:

```

frequency synchronization
  quality itu-t option 1
  log selection changes
!
ptp
  clock
  domain 4
  profile g.8265.1 clock-type master
!
  profile master
  transport ipv4
  sync frequency 64
  announce frequency 1
  delay-request frequency 64
interface HundredGigE 0/2/0/4
  ptp
  profile master
  port state master-only
!
  ipv4 address 18.1.1.1 255.255.255.0
!

```




---

**Note** For G.8265.1 PTP master clock, either SyncE or BITS reference source configuration is recommended. Otherwise, the device uses its own internal clock.

---

### Slave Global Configuration:

```

frequency synchronization
  quality itu-t option 1
  log selection changes
!
ptp
  clock
  domain 4
  profile g.8265.1 clock-type slave
!
  profile slave
  transport ipv4
  sync frequency 64
  announce interval 1
  delay-request frequency 64
interface HundredGigE 0/1/0/0
  ptp
  profile slave
  Master ipv4 18.1.1.1
  port state slave-only
!
  ipv4 address 18.1.1.2/24

```

## G.8275.1

**Table 36: Feature History Table**

Feature Name	Release Information	Feature Description
ITU-T G.8275.1 profile support	Release 7.3.1	This feature supports the architecture defined in ITU-T G.8275 for systems requiring accurate phase and time synchronisation, phase or time-of-day synchronization is required, and where each network device participates in the PTP protocol. Support of this capability is extended on the Cisco 8000 Series router, in this release.

G.8275.1 profile fulfills the time-of-day and phase synchronization requirements in telecom networks with all network devices participating in the PTP protocol. G.8275.1 profile provides better frequency stability for the time-of-day and phase synchronization.

Features of G.8275.1 profile are:

- Synchronization Model: G.8275.1 profile adopts hop-by-hop synchronization model. Each network device in the path from master to slave synchronizes its local clock to upstream devices and provides synchronization to downstream devices.
- Clock Selection: G.8275.1 profile also defines an alternate BMCA that selects a clock for synchronization and port state for the local ports of all devices in the network is defined for the profile. The parameters defined as a part of the BMCA are:
  - Clock Class
  - Clock Accuracy
  - Offset Scaled Log Variance
  - Priority 2
  - Clock Identity
  - Steps Removed
  - Port Identity
  - notSlave flag
  - Local Priority
- Port State Decision: The port states are selected based on the alternate BMCA algorithm. A port is configured to a master-only port state to enforce the port to be a master for multicast transport mode.
- Packet Rates: The nominal packet rate for Announce packets is 8 packets-per-second and 16 packets-per-second for Sync/Follow-Up and Delay-Request/Delay-Response packets.
- Transport Mechanism: G.8275.1 profile only supports Ethernet PTP transport mechanism.

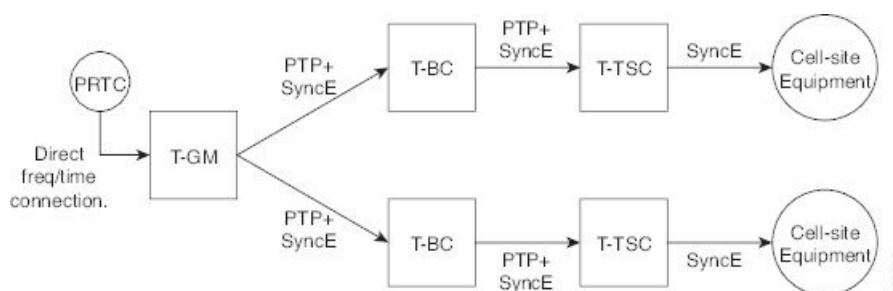
- Mode: G.8275.1 profile supports transport of data packets only in multicast mode. The forwarding is done based on forwardable or non-forwardable multicast MAC address.
- Clock Type: G.8275.1 profile supports the following clock types:
  - Telecom Grandmaster (T-GM): Provides timing for other network devices and does not synchronize its local clock to other network devices.
  - Telecom Time Slave Clock (T-TSC): A slave clock synchronizes its local clock to another PTP clock, but does not provide PTP synchronization to any other network devices.
  - Telecom Boundary Clock (T-BC): Synchronizes its local clock to a T-GM or an upstream T-BC clock and provides timing information to downstream T-BC or T-TSC clocks.
- Domain Numbers: The domain numbers that can be used in a G.8275.1 profile network ranges from 24 to 43. The default domain number is 24.

The G.8275.1 supports the following:

- T-GM: The telecom grandmaster (T-GM) provides timing to all other devices on the network. It does not synchronize its local clock with any other network element other than the Primary Reference Time Clock (PRTC).
- T-BC: The telecom boundary clock (T-BC) synchronizes its local clock to a T-GM or an upstream T-BC, and provides timing information to downstream T-BCs or T-TSCs. If at a given point in time there are no higher-quality clocks available to a T-BC to synchronize to, it may act as a grandmaster.
- T-TSC: The telecom time slave clock (T-TSC) synchronizes its local clock to another PTP clock (in most cases, the T-BC), and does not provide synchronization through PTP to any other device.

The following figure describes a sample G.8275.1 topology.

**Figure 7: A Sample G.8275.1 Topology**



### Configuring Global G.8275.1 Profile

The following configuration describes the steps involved to create a global PTP configuration profile that can be applied at an interface level. It uses G.8275.1 profile as an example:

```
Router# config terminal
Router(config)# ptp
Router(config-ptp)# clock
Router(config-ptp-clock)# domain 24
Router(config-ptp-clock)# profile g.8275.1 clock-type T-BC
Router(config-ptp-clock)# exit
Router(config-ptp)# profile slave
```

```

Router(config-ptp-profile)# multicast target-address ethernet 01-1B-19-00-00-00
Router(config-ptp-profile)# transport ethernet
Router(config-ptp-profile)# sync frequency 16
Router(config-ptp-profile)# announce frequency 8
Router(config-ptp-profile)# delay-request frequency 16
Router(config-ptp-profile)# exit
Router(config-ptp)# profile master
Router(config-ptp-profile)# multicast target-address ethernet 01-1B-19-00-00-00
Router(config-ptp-profile)# transport ethernet
Router(config-ptp-profile)# sync frequency 16
Router(config-ptp-profile)# announce frequency 8
Router(config-ptp-profile)# delay-request frequency 16
Router(config-ptp-profile)# exit
Router(config-ptp)# physical-layer-frequency
Router(config-ptp)# log
Router(config-ptp-log)# servo events
Router(config-ptp-log)# commit

```

### Verification

To display the configured PTP profile details, use **show run ptp** command.

```

Router# show run ptp

Wed Feb 28 11:16:05.943 UTC
ptp
 clock
  domain 24
  profile g.8275.1 clock-type T-BC
 !
profile slave
 multicast target-address ethernet 01-1B-19-00-00-00
 transport ethernet
 sync frequency 16
 announce frequency 8
 delay-request frequency 16
 !
profile master
 multicast target-address ethernet 01-1B-19-00-00-00
 transport ethernet
 sync frequency 16
 announce frequency 8
 delay-request frequency 16
 !
physical-layer-frequency
log
 servo events
 !

```

### Configuring PTP Master Interface

The below configuration describes the steps involved to configure a PTP interface to be a Master.

```

Router# configure terminal
Router(config)# interface HundredGigE0/0/0/0
Router(config-if)# ptp
Router(config-if-ptp)# profile master
Router(config-if-ptp)# port state master-only
Router(config-if-ptp)# commit

```

### Verification

To verify the port state details, use **show run interface** *interface-name* command.

```
Router# show run interface HundredGigE0/0/0/0
interface HundredGigE0/0/0/0
  ptp
  profile master
  port state master-only
!
```

### Configuring PTP Slave Interface

This procedure describes the steps involved to configure a PTP interface to be a Slave.

```
Router# configure terminal
Router(config)# interface HundredGigE0/0/0/1
Router(config-if)# ptp
Router(config-if-ptp)# profile slave
Router(config-if-ptp)# port state slave-only
Router(config-if-ptp)# commit
```

### Verification

To verify the port state details, use **show run interface** *interface-name* command.

```
Router# show run interface HundredGigE0/0/0/1
interface HundredGigE0/0/0/1
  ptp
  profile slave
  port state slave-only
!
```

### Configuring PTP Hybrid Mode

This procedure describes the steps involved to configure router in a hybrid mode. You configure hybrid mode by selecting PTP for phase and time-of-day (ToD) and another source for the frequency.



- 
- Note**
- G.8275.1 PTP profile supports only the hybrid mode. It is mandatory to have hybrid mode for G8275.1 profile for T-BC and T-TSC clock types. By default, the hybrid mode is used, regardless of the physical-layer-frequency configuration.
- 

To configure PTP Hybrid mode:

#### 1. Configure Global Frequency Synchronization

```
Router(config)# frequency synchronization
Router(config)# commit
```

#### 2. Configure Frequency Synchronization for an Interface. The time-of-day-priority setting specifies that SyncE to be used as a ToD source if there is no source available with a lower priority.

```
Router(config)# interface HundredGigE 0/0/0/0
Router(config-if)# frequency synchronization
Router(config-if-freqsync)# selection input
Router(config-if-freqsync)# time-of-day-priority 100
Router(config-if-freqsync)# commit
```

## Verification

```
Router # show frequency synchronization selection location 0/RP0/CP$
```

```
Tue Feb  6 06:34:17.627 UTC
Node 0/RP0/CPU0:
=====
Selection point: T0-SEL-B (3 inputs, 1 selected)
Last programmed 00:01:04 ago, and selection made 00:00:24 ago
Next selection points
SPA scoped : None
Node scoped : CHASSIS-TOD-SEL
Chassis scoped: LC_TX_SELECT
Router scoped : None
Uses frequency selection
Used for local line interface output
S Input Last Selection Point QL Pri Status
== =====
1 HundredGigE 0/0/0/0 0/2/CPU0 ETH_RXMUX 1 ePRTC 1 Locked
PTP [0/RP0/CPU0] n/a PRS 254 Available
Internal0 [0/RP0/CPU0] n/a ST3E 255 Available

Selection point: T4-SEL (3 inputs, 1 selected)
Last programmed 00:01:04 ago, and selection made 00:00:24 ago
Next selection points
SPA scoped : None
Node scoped : None
Chassis scoped: None
Router scoped : None
Uses frequency selection
Used for local clock interface output
S Input Last Selection Point QL Pri Status
== =====
1 HundredGigE 0/0/0/0 0/2/CPU0 ETH_RXMUX 1 ePRTC 1 Locked
PTP [0/RP0/CPU0] n/a PRS 254 Available
Internal0 [0/RP0/CPU0] n/a ST3E 255 Available

Selection point: 1588-SEL (2 inputs, 1 selected)
Last programmed 00:01:04 ago, and selection made 00:00:24 ago
Next selection points
SPA scoped : None
Node scoped : None
Chassis scoped: None
Router scoped : None
Uses frequency selection
S Input Last Selection Point QL Pri Status
== =====
1 HundredGigE 0/0/0/0 0/2/CPU0 ETH_RXMUX 1 ePRTC 1 Locked
Internal0 [0/RP0/CPU0] n/a ST3E 255 Available

Selection point: CHASSIS-TOD-SEL (2 inputs, 1 selected)
Last programmed 00:00:53 ago, and selection made 00:00:51 ago
Next selection points
SPA scoped : None
Node scoped : None
Chassis scoped: None
Router scoped : None
Uses time-of-day selection
S Input Last Selection Point Pri Time Status
== =====
1 PTP [0/RP0/CPU0] n/a 100 Yes Available
HundredGigE 0/0/0/0 0/RP0/CPU0 T0-SEL-B 1 100 No Available

RP/0/RP0/CPU0:SF-D#
```

```

RP/0/RP0/CPU0:SF-D#
RP/0/RP0/CPU0:SF-D#show frequency synchronization selection location 0/RP0/CP$
Thu Jan 1 00:16:56.105 UTC
Node 0/RP0/CPU0:
=====
Selection point: T0-SEL-B (3 inputs, 1 selected)
Last programmed 00:01:09 ago, and selection made 00:00:29 ago
Next selection points
SPA scoped : None
Node scoped : CHASSIS-TOD-SEL
Chassis scoped: LC_TX_SELECT
Router scoped : None
Uses frequency selection
Used for local line interface output
S Input Last Selection Point QL Pri Status
== =====
1 HundredGigE 0/0/0/0 0/2/CPU0 ETH_RXMUX 1 ePRTC 1 Locked
PTP [0/RP0/CPU0] n/a PRS 254 Available
Internal0 [0/RP0/CPU0] n/a ST3E 255 Available

Selection point: T4-SEL (3 inputs, 1 selected)
Last programmed 00:01:09 ago, and selection made 00:00:29 ago
Next selection points
SPA scoped : None
Node scoped : None
Chassis scoped: None
Router scoped : None
Uses frequency selection
Used for local clock interface output
S Input Last Selection Point QL Pri Status
== =====
1 HundredGigE 0/0/0/0 0/2/CPU0 ETH_RXMUX 1 ePRTC 1 Locked
PTP [0/RP0/CPU0] n/a PRS 254 Available
Internal0 [0/RP0/CPU0] n/a ST3E 255 Available

Selection point: 1588-SEL (2 inputs, 1 selected)
Last programmed 00:01:09 ago, and selection made 00:00:29 ago
Next selection points
SPA scoped : None
Node scoped : None
Chassis scoped: None
Router scoped : None
Uses frequency selection
S Input Last Selection Point QL Pri Status
== =====
1 HundredGigE 0/0/0/0 0/2/CPU0 ETH_RXMUX 1 ePRTC 1 Locked
Internal0 [0/RP0/CPU0] n/a ST3E 255 Available

Selection point: CHASSIS-TOD-SEL (2 inputs, 1 selected)
Last programmed 00:00:57 ago, and selection made 00:00:56 ago
Next selection points
SPA scoped : None
Node scoped : None
Chassis scoped: None
Router scoped : None
Uses time-of-day selection
S Input Last Selection Point Pri Time Status
== =====
1 PTP [0/RP0/CPU0] n/a 100 Yes Available
HundredGigE 0/0/0/0 0/RP0/CPU0 T0-SEL-B 1 100 No Available

```





# CHAPTER 11

## Configuration and File System Management

This module describes methods for configuration management and file transfer enhancements.

- [Secure file transfer from the Router, on page 167](#)
- [Auto-Save Configuration, on page 170](#)
- [Auto-Save and Copy Router Configuration Using Public Key Authentication, on page 172](#)
- [Increasing Commit Limit, on page 174](#)
- [View VRF-specific Configuration, on page 178](#)

### Secure file transfer from the Router

*Table 37: Feature History Table*

Feature Name	Release Information	Feature Description
Secure file transfer from the Router	Release 7.9.1	Your routers are now enabled to transfer files securely to an archive server. It's made possible because the copy command now supports SFTP (Secure File Transfer Protocol) and SCP (Secure Copy Protocol) using the underlying SSH protocol implementation. Secure transfer of files from the router maintains the integrity, confidentiality, and availability of network configurations.  This feature modifies the <b>copy</b> command.

You can duplicate files or data in the router from one location to another using the **copy** command. This functionality helps to create a copy of a file, folder, or data set and place it in a specific destination. You can use the copy functionality to back up files, move data between directories, create duplicates of the files for editing or distribution without modifying the original content. It also allows you to retain the original data while making a duplicate that you can further manipulate independently.

Starting with Cisco IOS XR Release 7.9.1, we've enhanced the functionality of the copy command to support secure file transfer from the router. Secure file transfer protects data during transit using the SFTP (Secure File Transfer Protocol) and SCP (Secure Copy Protocol) when sharing files within or across networks. The SFTP and SCP functionalities in the copy feature use the SSH protocol implementation in the router to secure transfer the files to a remote server.

You can use the following options in the **copy** command for secure file transfer:

- **sftp**: You can transfer the files to a remote location using the **SFTP** file transfer protocol. SFTP is a secure file transfer protocol for transferring large files.
- **scp**: You can transfer the files to a remote location using the **SCP** file transfer protocol. SCP is a secure copy protocol to transfer files between servers.

Starting Cisco IOS XR Software Release 7.10.1, you can use public-key authentication while copying the running configuration.

### Configuration Example for SCP and SFTP Using Public-Key Authentication

While you're using public-key authentication for copying running configuration from the router to a remote server, you don't need to mention **password** in the command. The following example shows how you can configure public-key authentication while copying configuration using the SCP protocol:

```
Router#copy running-config scp://root@192.0.4.2//var/opt/run_conf_scp.txt
```

## Prerequisites for secure file transfer

Enable the SSH Server in the router:

```
Router# config
Router(config)# ssh server v2
Router(config)# ssh server vrf default
Router(config)# ssh server netconf vrf default
Router(config)# commit
```

## Secure file transfer using SFTP

You can copy the running configuration file from the router to a remote server using SFTP as follows:

```
Router# copy running-config sftp://root:testpassword@192.0.2.1//var/opt/run_conf_sftp.txt
```

```
Destination file name (control-c to cancel): [/var/opt/run_conf_sftp.txt]?
```

```
.
215 lines built in 1 second
[OK]Connecting to 192.0.2.1...22
Password:
sftp> put /tmp/tmpsymlink/nvgen-34606-_proc_34606_fd_75 /var/opt/run_conf_sftp.txt
```

```
/tmp/tmpsymlink/nvgen-34606-_proc_34606_fd_75
```

```
Transferred 3271 Bytes
3271 bytes copied in 0 sec (3271000)bytes/sec
sftp> exit
```

### Verification in the SFTP Server

```
[root@sftp_server ~]# ls -ltr /var/opt/run_conf_sftp.txt
-rw-r--r-- 1 root root 3271 Mar 21 18:07 /var/opt/run_conf_sftp.txt
```

## Secure file transfer using SCP

You can copy the running configuration file from the router to a remote server using SFTP as follows:

```
Router# copy running-config sftp://root:testpassword@192.0.2.1//var/opt/run_conf_sftp.txt
```

```
Destination file name (control-c to cancel): [/var/opt/run_conf_sftp.txt]?
```

```
.
215 lines built in 1 second
[OK]Connecting to 192.0.2.1...22
Password:
sftp> put /tmp/tmpsymlink/nvgen-34606-_proc_34606_fd_75 /var/opt/run_conf_sftp.txt
```

```
/tmp/tmpsymlink/nvgen-34606-_proc_34606_fd_75
```

```
Transferred 3271 Bytes
3271 bytes copied in 0 sec (3271000)bytes/sec
sftp> exit
```

### Verification in the SFTP Server

```
[root@sftp_server ~]# ls -ltr /var/opt/run_conf_sftp.txt
-rw-r--r-- 1 root root 3271 Mar 21 18:07 /var/opt/run_conf_sftp.txt
```

# Auto-Save Configuration

Table 38: Feature History Table

Feature Name	Release Information	Feature Description
Auto-Save with Secure File-Transfer and Additional Configurable Parameters	Release 7.9.1	<p>Apart from automatically backing up the running configuration after every commit, you can also do the following with Auto-Save:</p> <ul style="list-style-type: none"> <li>• Save running configurations to remote systems using Secure Copy Protocol (SCP) and Secure File Transfer Protocol (SFTP).</li> <li>• Configure wait-time between two subsequent auto-saves.</li> <li>• Append time-stamp to the file name of the saved configuration.</li> <li>• Save the encrypted password.</li> <li>• Specify the maximum number of files that you can auto-save.</li> </ul> <p>The feature introduces these changes:</p> <p>CLI: Modified the <b>configuration commit auto-save</b> command by adding the following keywords:</p> <ul style="list-style-type: none"> <li>• <b>filename scp</b></li> <li>• <b>filename sftp</b></li> <li>• <b>wait-time</b></li> <li>• <b>timestamp</b></li> <li>• <b>password</b></li> <li>• <b>maximum</b></li> </ul> <p>Yang Data Model:</p> <ul style="list-style-type: none"> <li>• New XPath for Cisco-IOS-XR-config-autosave-cfg</li> <li>• New XPath for Cisco-IOS-XR-um-config-commit-cfg</li> </ul>

You can configure the router to automatically take the backup of the running configuration by using **configuration commit auto-save** command. This auto-save feature saves the configuration to the specified location on the router after every **commit** is made. These auto-save files are stored in the form of Linux files.

Starting Cisco IOS XR Software Release 7.9.1, the auto-save feature is enhanced to provide a set of functionalities. Use the following keywords to achieve the same:

- **scp and sftp** - You can save the running configuration backup files to remote location using **scp** and **sftp** file transfer protocols. SCP is a secure copy protocol to transfer files between servers. Whereas SFTP is a secure file transfer protocol for transferring large files.
- **password** - You can save encrypted passwords for the remote and non-remote URLs.
- **maximum** - You can mention maximum number of files that can be saved automatically. Once the maximum number of auto-saved file is reached, the newer auto-save files starts replacing the older auto-save files. The default value of **maximum** is 1. You can save upto 4294967295 files.
- **timestamp** - Using this keyword, the time-stamp can be appended to the auto-saved configuration file name. The **timestamp** uses the time and timezone configured on the router. The saved file displays timestamp in <day> <month> <date> <hours> <minutes> <seconds> <milliseconds> format. Here is an example of auto-saved file with time-stamp - : *test\_123.autosave.1.ts.Tue\_Jan\_31\_15-15-51\_805\_IST*
- **wait-time** - You can specify how long to wait before next auto-save happens in terms of days, months or hours after the commit is made. The default value of **wait-time** is zero.

### Restriction for Auto-Save Configuration

The auto-save configuration is only available on the local paths, scp, and sftp paths.

Starting Cisco IOS XR Software Release 7.10.1, you can use public-key authentication while automatically saving the running configuration. For more detailed information on how to use public-key authentication, see [Auto-Save and Copy Router Configuration Using Public Key Authentication, on page 172](#).

## Configure Auto-Save

Use the **configuration commit auto-save** command to auto save the configuration.

```
Router#configure
Router(config)#configuration commit auto-save
Router(config-cfg-autosave)#commit
```

You can also configure options such as **password**, **timestamp**, **maximum**, and **wait-time** with the **configuration commit auto-save** command. The location to save the file-name must be specified in <protocol>://<user>@<host>:<port>/<url-path>/<file-name> format.

```
Router(config-cfg-autosave)#configuration commit auto-save filename
sftp://user1@server1://test-folder/test_123
Router(config-cfg-autosave)#password clear encryption-default cisco
Router(config-cfg-autosave)#timestamp
Router(config-cfg-autosave)#maximum 10
Router(config-cfg-autosave)#wait-time days 0 hours 0 minutes 0 seconds 5
Router(config-cfg-autosave)#commit
```

### Running Configuration

```
Router#show running-config configuration commit auto-save
configuration commit auto-save
  filename sftp://user1@server1://test-folder/test_123
  password encrypted encryption-default <password for above user>
  timestamp
  maximum 10
  wait-time days 0 hours 0 minutes 0 seconds 5
!
```

## Auto-Save and Copy Router Configuration Using Public Key Authentication

Table 39: Feature History Table

Feature Name	Release Information	Feature Description
Auto-Save and Copy Router Configuration Using Public Key Authentication	Release 7.10.1	<p>You can now experience passwordless authentication while automatically saving running configurations and securely copying them on the router. The feature uses public key-based authentication, a secure logging method using a secure shell (SSH), which provides increased data security. This feature offers automatic authentication and single sign-on benefits, which also aids in a secure automation process.</p> <p>The feature introduces these changes:</p> <p><b>CLI:</b></p> <ul style="list-style-type: none"> <li>• The <b>configuration commit auto-save</b> command supports password-less authentication.</li> <li>• The <b>copy</b> command supports password-less authentication.</li> </ul>

From Cisco IOS XR Software Release 7.10.1, you don't need to remember and enter the **password** as you can use public key-based authentication while doing the following:

- Automatically saving your running configuration
- Copying the configuration from a source (such as a network server) to a destination (such as a flash disk)

Password is automatically verified when you have enabled SSH connection using public key-based authentication. Using public key-based authentication avoids several problems such as password disclosure and password leakage.

Public key is mathematically related to private key. The private key is secret, whereas the public key is available on the servers. You can copy the public key to the SSH server from the SSH client. Then, when you try to secure the running configuration, the SSH server tries to authenticate by generating a challenge using the public key. Only the private key can answer this challenge. As the keys are related, log-in is successful.

### Prerequisites for Auto-Save and Copy Router Configuration Using Public Key Authentication

Ensure you have enabled public key-based authentication of SSH clients, using the following steps:

- Generate RSA key pair on the router configured as the SSH client. Use the **crypto key generate authentication-ssh rsa** command to generate the RSA key pair.
- Use the **show crypto key mypubkey authentication-ssh rsa** command to view the details of the RSA key. The key value starts with *ssh-rsa* in this output.
- Copy the RSA public key from the SSH client to the SSH server.

For more detailed information on how to enable SSH connection using public-key based authentication, see *Public Key Based Authentication of SSH Clients* in System Security Configuration Guide for Cisco 8000 Series Routers.

## Configuration Example for Auto-Save Using Public Key Authentication

When you are using public key authentication, you don't need to mention **password**.

```
Router(config-cfg-autosave)#configuration commit auto-save filename
sftp://user1@server1://test-folder/test_123
Router(config-cfg-autosave)#timestamp
Router(config-cfg-autosave)#maximum 10
Router(config-cfg-autosave)#wait-time days 0 hours 0 minutes 0 seconds 5
Router(config-cfg-autosave)#commit
```

### Running Configuration

```
Router#show running-config configuration commit auto-save
configuration commit auto-save
  filename sftp://user1@server1://test-folder/test_123
  timestamp
  maximum 10
  wait-time days 0 hours 0 minutes 0 seconds 5
!
```

# Increasing Commit Limit

Table 40: Feature History Table

Feature Name	Release Information	Feature Description
Increasing Commit Limit	Release 24.2.11	<p>The maximum number of commits is increased in the router that allows you to configure complex topology changes without interruptions caused by the default blocking of commit changes during rebase or ASCII backup operations. You can prevent the commit operation from getting blocked by using the <b>cfs check</b> command, which increases the commit (pacount) count from 20 to 40, and the commit file diff size (configuration data) from 2 MB to 4 MB, and by using the <b>clear configuration ascii inconsistency</b> command, which performs an ASCII backup after 55 minutes.</p> <p>The feature modifies the following commands:</p> <ul style="list-style-type: none"> <li>• <b>cfs check</b></li> <li>• <b>clear configuration ascii inconsistency</b></li> </ul>

The Cisco IOS XR Routers use a two-stage configuration model. The first stage is target configuration, where you build the configurations using necessary commands in the command line interface. The second stage is the commit, where the configuration made in the target stage is added to the router configuration using the **commit** command. After each commit, the router generates a file for the newly configured changes and adds it to its running configuration, making it an integral part of the running configuration.



**Note** This target configuration does not impact the router's running configuration.

The Cisco IOS XR routers perform rebase and ASCII backup operations to maintain the real time configuration in the backup copy. The rebase and ASCII backup operations block you from committing configurations to the router.





**Note** Starting with Release 24.3.1, the rebase operation no longer blocks the commit operation.

This allows you to configure complex topology changes without being interrupted by the default blocking of commit changes during the rebase operation. For more information, see the section [Concurrent Configuration Rebase during Commit](#), on page 177.

In rebase, the router automatically saves your changes to the backup binary configuration file after 20 commits, or 2 MB of configuration data. The router blocks the commit while saving the configuration to the backup file. The router takes a few seconds to complete the rebase operation, during which, if you terminate the CLI session, the router loses the target configurations in the blocked commit.

In ASCII backup, the router automatically saves a copy of its running configuration in the ASCII format. This backup process takes place if there has been a commit to the router configuration and when the ASCII backup timer completes a 55-minute window after the previous backup event. However, if there was no commit when the ASCII backup timer completes 55 minutes, the counter is reset without any backup. During the ASCII backup, the router blocks the configuration commits.

Starting with Release 24.2.11, we have made the following enhancements:

- You can use the **cfs check** command to increase the rebase limits in the router from 20 to 40 commits and the configuration data from 2 MB to 4 MB. When configuring the router, you can check the current commit count and configuration data size using the **show cfmgr commitdb** command. If the commit count is 20 or higher, or the configuration data size is 2 MB or above, the router will initiate a rebase within 10 seconds. By using the **cfs check** command to increase the commit count to 40 and the configuration data to 4 MB, you can commit without delay.
- You can use the **clear configuration ascii inconsistency** command to perform an ASCII backup and reset the ASCII backup timer to zero. Once the backup is complete, the router will automatically initiate the next periodic ASCII backup operation only after 55 minutes from the time the **clear configuration ascii inconsistency** command is executed.

## Guidelines and Restrictions for Increasing the Commit Limit

- The **clear configuration ascii inconsistency** command initiates an ASCII backup and resets the ASCII backup timer count to zero. Following this, the router will automatically initiate the next periodic ASCII backup operation only after 55 minutes from the time **clear configuration ascii inconsistency** command is executed. For example, if you execute a commit operation after executing a **clear configuration ascii inconsistency** command, the router will perform an ASCII backup operation 55 minutes after the **clear configuration ascii inconsistency** command was executed, and merge the new commit into ASCII backup. Hence, before the next 55 minutes, you must execute the **clear configuration ascii inconsistency** command again to reset the ASCII backup timer to zero.
- When the router enters standby mode or reloads, the ASCII timer does not reset to zero, and the router performs an ASCII backup operation 55 minutes after the first commit operation before the standby mode or reload.
- Cisco does not recommend executing **clear configuration inconsistency** and **clear configuration ascii inconsistency** commands regularly after each commit, as it causes hard disk wear and tear. You should execute these commands only before a commit or sequence of commits that must be done within a specific timeframe and without being delayed by rebase and ASCII backup operations. As these commands

perform disk input and output operations in the background, frequent execution of these commands causes frequent access to the hard disk, which increases the wear and tear on the hard disk.

## Increasing the Rebase Limits

You can increase the rebase limits as follows:

1. Use the **cfs check** command to increase the commit count to 40 and the configuration data to 4 MB.

```
Router# cfs check
Creating any missing directories in Configuration File system...OK
Initializing Configuration Version Manager...OK
Syncing commit database with running configuration...OK
```

2. Verify if the **cfs check** command is executed using the **show configuration history** command.

```
Router# show configuration history last 5
Sno.  Event      Info                                     Time Stamp
~~~~  ~~~~~      ~~~~                                     ~~~~~~
1      cfs check  completed                               Wed Jan 10 11:42:21 2024
2      commit    id 1000000001                          Wed Jan 10 11:39:26 2024
3      startup   configuration applied                   Wed Jan 10 11:39:02 2024
```

## Perform ASCII Backup and Rest ASCII Backup Timer

You can perform ASCII backup and rest ASCII backup timer as follows:

1. Use the **clear configuration ascii inconsistency** command to perform ASCII backup at that instance and reset the ASCII backup timer count to zero.

```
Router# clear configuration ascii inconsistency
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!! It is recommended to run this command only when all nodes in router      !!!!
!!!! are in IOS-XR RUN state. To determine node state, run following command: !!!!
!!!! 'show platform'.                                                         !!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Warning: !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Proceed with the command?[confirm] y
  Ascii configuration backup is in progress...
Configuration ascii backup complete
```

2. Verify if the **clear configuration ascii inconsistency** command is executed using the **show configuration history** command.

```
Router# show configuration history last 5
Sno.  Event      Info                                     Time Stam
~~~~  ~~~~~      ~~~~                                     ~~~~~~
1      backup     Periodic ASCII backup                   Wed Jan 10 11:48:20 2024
2      cfs check  completed                               Wed Jan 10 11:42:21 2024
3      commit    id 1000000001                          Wed Jan 10 11:39:26 2024
4      startup   configuration applied                   Wed Jan 10 11:39:02 2024
```

## Concurrent Configuration Rebase during Commit

Table 41: Feature History Table

Feature Name	Release Information	Feature Description
Concurrent Configuration Rebase during Commit	Release 24.3.1	<p>Introduced in this release on: Fixed Systems (8200, 8700); Centralized Systems (8600); Modular Systems (8800 [LC ASIC: Q100, Q200, P100])</p> <p>The router performs the commit and rebase operations simultaneously, ensuring that the commit operation remains unblocked during the rebase operation.</p> <p>This removes the need to use the <b>cfs check</b> command to increase the commit count and the commit file diff size.</p>

Cisco IOS XR routers use a two-stage configuration model. In the first stage, configurations are built using necessary commands in the command line interface, and in the second stage, the configurations are committed to the router.

During rebase and ASCII backup operations, the router blocks configuration commits. However, the "Concurrent Configuration Rebase during Commit" feature allows the router to perform commit and rebase operations simultaneously, ensuring that the commit operation remains unblocked during the rebase operation.

The Cisco IOS XR routers perform rebase and ASCII backup operations to maintain the real time configuration in the backup copy.

Before Release 24.3.1,

- The rebase and ASCII backup operations block you from committing configurations to the router.
- You can increase the maximum number of commits and reset the ASCII backup timer to allow the router to configure complex topology changes without interruptions caused by the default blocking of commit changes during rebase or ASCII backup operations. For more information, see the section [Increasing Commit Limit, on page 174](#).

From Release 24.3.1,

- The router performs the commit and rebase operations simultaneously, ensuring that the commit operation remains unblocked during the rebase operation. This removes the need to use the **cfs check** command to increase the commit count and the commit file diff size.
- However, the ASCII backup operations still block the commit operation. You can reset the ASCII backup timer using the clear configuration ascii inconsistency command. This allows the router to perform an ASCII backup after 55 minutes and perform commit operations without being blocked by ASCII backup operations. For more information on ASCII backup, see the section [Increasing Commit Limit, on page 174](#).

# View VRF-specific Configuration

Table 42: Feature History Table

Feature Name	Release Information	Feature Description
View VRF-specific Configuration	Release 24.2.11	<p>You can now filter the configurations associated with a specific VRF using the <b>show running-configuration filter vrf</b> command. Earlier, the <b>show running configuration</b> command displayed configuration under a specific keyword only and that may not publish all configurations related to the object.</p> <p><b>CLI:</b></p> <p><a href="#">show running-configuration filter vrf</a></p>

Starting Cisco IOS XR Software Release 24.2.11, you can use the **show running-configuration filter vrf** command publishes all running configuration data associated with a VRF name, including configurations under various CLI link-points or keywords. Also, it reveals logical dependencies, showing parent interface configurations for subinterfaces and bundle configurations for member interfaces that are part of a VRF.

## Restrictions for Filtering Running Configuration for VRF

- The **show running-configuration filter vrf** doesn't support filtering for default VRFs that aren't explicitly labeled in the configuration.
- Logical dependencies are only automatically displayed for main interfaces with their subinterfaces and bundle interfaces with their members; other dependencies require a specific request.
- Only a subset of the data from the **show running configuration** command is displayed by the new CLI.
- The accuracy of the data provided by the new CLI isn't independently verified.

## View Running Configuration for a specific VRF

Use the **show running-configuration filter vrf** command to filter the running configuration for a specific vrf and list all the configurations related to that object.

### Example:

```
Router#show running-configuration filter vrf vrf80

!! Building configuration...
!! IOS XR Configuration 24.2.1.32M
!! Last configuration change at Mon Jan 15 05:09:20 2024 by cisco
```

```
!  
vrf vrf80  
  address-family ipv4 unicast  
    import route-target  
      1:80  
    !  
    export route-target  
      1:80  
    !  
  !  
  address-family ipv6 unicast  
    import route-target  
      1:80  
    !  
    export route-target  
      1:80  
    !  
  !  
  !  
  neighbor 192.0.2.1  
    remote-as 200  
    ebgp-multihop 4  
    update-source Loopback90  
    address-family ipv4 unicast  
      route-policy PASS in  
      route-policy PASS out  
    !  
  !  
  !  
  !  
end
```

---

