



System Management Configuration Guide for Cisco NCS 5500 Series Routers, IOS XR Release 7.3.x

First Published: 2021-02-10

Last Modified: 2021-02-11

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.

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)

© 2021 Cisco Systems, Inc. All rights reserved.



CONTENTS

PREFACE

Preface xiii

Changes to this Document xiii

Communications, Services, and Additional Information xiii

CHAPTER 1

New and Changed System Management Features 1

System Management Features Added or Modified in IOS XR Release 7.3.x 1

CHAPTER 2

Configuring Manageability 3

Information about XML Manageability 3

How to Configure Manageability 3

Configuring the XML Agent 3

Configuration Examples for Manageability 4

Enabling VRF on an XML Agent: Examples 4

CHAPTER 3

Configuring Physical and Virtual Terminals 7

Prerequisites for Implementing Physical and Virtual Terminals 7

Information About Implementing Physical and Virtual Terminals 7

Line Templates 7

Line Template Configuration Mode 8

Line Template Guidelines 8

Terminal Identification 9

vtty Pools 9

How to Implement Physical and Virtual Terminals on Cisco IOS XR Software 10

Modifying Templates 10

Creating and Modifying vtty Pools 11

Monitoring Terminals and Terminal Sessions 13

Configuration Examples for Implementing Physical and Virtual Terminals	14
--	----

CHAPTER 4
Configuring Simple Network Management Protocol 17

Prerequisites for Implementing SNMP	17
Restrictions for SNMP use on Cisco IOS XR Software	17
Information about Implementing SNMP	18
SNMP Functional Overview	18
SNMP Manager	18
SNMP Agent	18
MIB	18
SNMP Versions	20
Comparison of SNMPv1, v2c, and v3	20
Security Models and Levels for SNMPv1, v2, v3	21
SNMPv3 Benefits	22
SNMPv3 Costs	22
User-Based Security Model	23
View-Based Access Control Model	23
IP Precedence and DSCP Support for SNMP	24
Session MIB support on subscriber sessions	24
SNMP Notifications	24
Session Types	25
How to Implement SNMP on Cisco IOS XR Software	25
Configuring SNMPv3	26
Configure to Drop Error PDUs	28
Configuring SNMPv3: Examples	29
Configuring SNMP Trap Notifications	33
Configure to Drop Error PDUs	34
Configuring Trap Notifications: Example	35
Setting the Contact, Location, and Serial Number of the SNMP Agent	36
Defining the Maximum SNMP Agent Packet Size	37
Changing Notification Operation Values	38
Setting IP Precedence and DSCP Values	39
Setting an IP Precedence Value for SNMP Traffic: Example	40
Setting an IP DSCP Value for SNMP Traffic: Example	41

Displaying SNMP Context Mapping	41
Monitoring Packet Loss	41
Configuring MIB Data to be Persistent	42
Configuring LinkUp and LinkDown Traps for a Subset of Interfaces	43
SNMP Context Mapping Configuration	45

CHAPTER 5

Configuring Object Tracking 49

Configuring Object Tracking	49
Prerequisites for Implementing Object Tracking	49
Information about Object Tracking	50
How to Implement Object Tracking	50
Tracking the Line Protocol State of an Interface	50
Tracking IP Route Reachability	52
Building a Track Based on a List of Objects	54
Building a Track Based on a List of Objects - Threshold Percentage	56
Building a Track Based on a List of Objects - Threshold Weight	58
Tracking IPSLA Reachability	59
Tracking BGP Neighbor Address-Family State	60
Configuration Examples for Configuring Object Tracking	61
Additional References	63

CHAPTER 6

Configuring Cisco Discovery Protocol 65

Prerequisites for Implementing CDP	65
Information About Implementing CDP	65
How to Implement CDP on Cisco IOS XR Software	67
Enabling CDP	67
Modifying CDP Default Settings	68
Monitoring CDP	69
Examples	70
Configuration Examples for Implementing CDP	73

CHAPTER 7

Configuring Periodic MIB Data Collection and Transfer 75

Prerequisites for Periodic MIB Data Collection and Transfer	75
Information About Periodic MIB Data Collection and Transfer	75

SNMP Objects and Instances	75
Bulk Statistics Object Lists	76
Bulk Statistics Schemas	76
Bulk Statistics Transfer Options	76
Benefits of Periodic MIB Data Collection and Transfer	76
How to Configure Periodic MIB Data Collection and Transfer	77
Configuring a Bulk Statistics Object List	77
Configuring a Bulk Statistics Schema	78
Configuring Bulk Statistics Transfer Options	80
Periodic MIB Data Collection and Transfer: Example	83

CHAPTER 8

Configuring Flexible Command Line Interface	85
Flexible CLI Configuration Groups	85
Flexible Configuration Restrictions	85
Configuring a Configuration Group	87
Simple Configuration Group: Example	88
Configuration Group Applied to Different Places: Example	89
Verifying the Configuration of Configuration Groups	89
Regular Expressions in Configuration Groups	91
Configuration Examples Using Regular Expressions	98
Configuration Group with Regular Expression: Example	98
Configuration Group Inheritance with Regular Expressions: Example	100
Layer 2 Transport Configuration Group: Example	101
Configuration Group Precedence: Example	101
Changes to Configuration Group are Automatically Inherited: Example	102
Configuration Examples for Flexible CLI Configuration	102
Basic Flexible CLI Configuration: Example	102
Interface MTU Settings for Different Interface Types: Example	104
ACL Referencing: Example	106
Local Configuration Takes Precedence: Example	107
ISIS Hierarchical Configuration: Example	108
OSPF Hierarchy: Example	112
Link Bundling Usage: Example	115

CHAPTER 9	Managing Router Hardware	117
	Clear the Memory and the Partitions of a Card	117
	System Logs during RSP Switchover	120
CHAPTER 10	Configuring Network Time Protocol	121
	Prerequisites for Implementing NTP on Cisco IOS XR Software	121
	Information About Implementing NTP	121
	Configuring Poll-Based Associations	123
	Configuring Broadcast-Based NTP Associates	125
	Configuring NTP Access Groups	127
	Configuring NTP Authentication	129
	Disabling NTP Services on a Specific Interface	131
	Configuring the Source IP Address for NTP Packets	132
	Configuring the System as an Authoritative NTP Server	134
	FQDN for NTP Server	135
	Configure FQDN for NTP server	135
	Updating the Hardware Clock	136
	Verifying the Status of the External Reference Clock	137
	Examples	138
	Configuration Examples for Implementing NTP	139
	Configuring NTP server inside VRF interface	142
CHAPTER 11	Configuring Precision Time Protocol	145
	PTP Overview	145
	Restrictions for PTP	148
	PTP Support Information	149
	PTP Hardware Support Matrix	149
	Slow Tracking	156
	ITU-T Telecom Profiles for PTP	157
	G.8265.1	157
	G.8275.1	158
	G.8275.2	160
	Configuring PTP	161

Configuring Global G.8275.1 Profile	161
Configuring PTP Master Interface	163
Configuring PTP Slave Interface	163
Configuring PTP Hybrid Mode	164
Configuring PTP Telecom Profile Interface	166
Configuring PTP Telecom Profile Clock	168
Configuring PTP Delay Asymmetry	169
Configuration Examples	172
Slave Configuration Example	172
Master Configuration Example	172
PTP Hybrid Mode Configuration Example	172
ITU-T Telecom Profile Examples:	173
G.8265.1 Profile Configuration Examples	173
G.8275.1 Profile Configuration Examples	174
G.8275.2 Profile Configuration Examples	175
Configure E-SyncE on Primary and Secondary Interface	177

CHAPTER 12
Configuring Smart Licensing 181

What is Smart Licensing?	181
What is Flexible Consumption Model?	182
Software Innovation Access	185
Verification	187
How Does Smart Licensing Work?	188
What is Cisco Smart Software Manager?	189
Smart Licensing Deployment Options	189
Configuring Smart Licensing	191
Prerequisites for Configuring Smart Licensing	191
Setting up the Router for Smart Licensing	191
Configuring a Communications Connection Between the Router and Cisco Smart Software Manager	192
Configuring a Direct Cloud Connection	192
Configuring a Connection Through an HTTP Proxy	193
Connecting to CSSM On-Premise	196
Installing CSSM On-Premise	198

Registering and Activating Your Router	198
Generating a New Token from CSSM	199
Registering Your Device With the Token	201
Renewing Your Smart Licensing Registration	202
Deregistering Your Router from CSSM	202
Verifying the Smart Licensing Configuration	203
Smart Licensing Configuration Examples	205
Example: Viewing the Call Home Profile	205
Example: Viewing License Information Before Registration	205
Example: Registering the Router	208
Example: Viewing License Information After Registration	208
Specific License Reservation	210
Configuring Specific License Reservation	211
Enabling Specific License Reservation	212
Generating a Request Code from the Router	212
Reserving a SLR License in Cisco SSM	212
Registering a Router with the Authorization Code	212
To cancel, verify, remove, and return SLR	213
Cancelling a SLR Request	213
Verifying the License Status After Cancelling SLR	213
Removing SLR from a Router	214
Generating a Return Code from a Router	214
Removing the Product Instance from CSSM	214
Configuration Examples for SLR	215
Enabling FCM	215
Enabling SLR	215
Viewing the License Information Before Registering	215
Viewing the License Status After Registering	219

CHAPTER 13
Configuring Zero Touch Provisioning 223

Manual ZTP Invocation	224
Authentication on Data Ports	225
ZTP Bootscript	227
ZTP Utilities	228

Customize the ZTP Configurable Options 229

Examples 230

CHAPTER 14

Upgrading Field-Programmable Device 233

Prerequisites for FPD Image Upgrades 233

Overview of FPD Image Upgrade Support 233

FPD upgrade service 233

Determining Upgrade Requirement 234

Manual FPD upgrade 234

How to Upgrade FPD Images 235

Configuration Examples for FPD Image Upgrade 238

Auto FPD Upgrade 241

Limitations and Usage Guidelines 241

Automatic FPD Upgrade for PSU 243

Upgrade Failure 243

CHAPTER 15

Y.1564 - Ethernet Service Activation Test 245

Y.1564 - Ethernet Service Activation Test 245

CoS DEI Support Matrix 247

Supported Bandwidth Parameters 247

Service Activation Test Target Matrix 247

Configuration Examples 248

Configuring Ethernet Service Activation Test Color Profile 248

Verification 248

CHAPTER 16

Configuration and File System Management 251

Auto-Save Configuration 251

Configure Auto-Save 251

Increasing Commit Limit 252

Guidelines and Restrictions for Increasing the Commit Limit 253

Increasing the Rebase Limits 253

Perform ASCII Backup and Rest ASCII Backup Timer 254

CHAPTER 17

Synchronous Ethernet ESMC and SSM 255

Frequency Synchronization Timing Concepts	255
Sources	255
Selection Points	256
Restrictions	257
SyncE Hardware Support Matrix	257
Configuring Frequency Synchronization	259
Enabling Frequency Synchronization on the Router	259
Configuring Frequency Synchronization on an Interface	259
Configuring Frequency Synchronization on a Clock Interface	260
Verifying the Frequency Synchronization Configuration	260
Verifying the ESMC Configuration	263
Verifying Controllers Timing LEDs	264



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 xiii](#)
- [Communications, Services, and Additional Information, on page xiii](#)

Changes to this Document

Table 1: Changes to this Document

Date	Summary
February 2021	Initial release of this document.

Communications, Services, and Additional Information

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

Cisco Bug Search Tool

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



CHAPTER

1

New and Changed 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 7.3.x, on page 1](#)

System Management Features Added or Modified in IOS XR Release 7.3.x

Feature	Description	Changed in Release	Where Documented
PTP Delay Asymmetry	This feature was introduced.	Release 7.3.1	Configuring PTP Delay Asymmetry, on page 169
Increasing Commit Limit	This feature was introduced.	Release 7.3.6	Increasing Commit Limit, on page 252



CHAPTER 2

Configuring Manageability

This module describes the configuration required to enable the Extensible Markup Language (XML) agent services. The XML Parser Infrastructure provides parsing and generation of XML documents with Document Object Model (DOM), Simple Application Programming Interface (API) for XML (SAX), and Document Type Definition (DTD) validation capabilities:

- DOM allows customers to programmatically create, manipulate, and generate XML documents.
- SAX supports user-defined functions for XML tags.
- DTD allows for validation of defined document types.
- [Information about XML Manageability, on page 3](#)
- [How to Configure Manageability, on page 3](#)
- [Configuration Examples for Manageability, on page 4](#)

Information about XML Manageability

The Cisco IOS XR Extensible Markup Language (XML) API provides a programmable interface to the router for use by external management applications. This interface provides a mechanism for router configuration and monitoring utilizing XML formatted request and response streams. The XML interface is built on top of the Management Data API (MDA), which provides a mechanism for Cisco IOS XR components to publish their data models through MDA schema definition files.

Cisco IOS XR software provides the ability to access the router via XML using a dedicated TCP connection, Secure Socket Layer (SSL), or a specific VPN routing and forwarding (VRF) instance.

How to Configure Manageability

Configuring the XML Agent

This explains how to configure the XML agent.

SUMMARY STEPS

1. **xml agent** [ssl]
2. **iteration on size** *iteration-size*
3. **session timeout** *timeout*

4. **throttle** { *memory size* | *process-rate tags* }
5. **vrf** { *vrfname* | **default** } [*ipv4 access-list access-list-name*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	xml agent [<i>ssl</i>] Example: RP/0/RP0/CPU0:router(config)# xml agent ssl	Enables Extensible Markup Language (XML) requests over a dedicated TCP connection and enters XML agent configuration mode. Use the ssl keyword to enable XML requests over Secure Socket Layer (SSL).
Step 2	iteration on size <i>iteration-size</i> Example: RP/0/RP0/CPU0:router(config-xml-agent)# iteration on size 500	Configures the iteration size for large XML agent responses in KBytes. The default is 48.
Step 3	session timeout <i>timeout</i> Example: RP/0/RP0/CPU0:router(config-xml-agent)# session timeout 5	Configures an idle timeout for the XML agent in minutes. By default, there is no timeout.
Step 4	throttle { <i>memory size</i> <i>process-rate tags</i> } Example: RP/0/RP0/CPU0:router(config-xml-agent)# throttle memory 300	Configures the XML agent processing capabilities. <ul style="list-style-type: none"> Specify the memory size in Mbytes. Values can range from 100 to 600. In IOS XR 64 bit, the values range from 100 to 1024. The default is 300. Specify the process-rate as the number of tags that the XML agent can process per second. Values can range from 1000 to 30000. By default the process rate is not throttled.
Step 5	vrf { <i>vrfname</i> default } [<i>ipv4 access-list access-list-name</i>] Example: RP/0/RP0/CPU0:router(config-xml-agent)# vrf vrf1	Configures the dedicated agent or SSL agent to receive and send messages via the specified VPN routing and forwarding (VRF) instance.

Configuration Examples for Manageability

Enabling VRF on an XML Agent: Examples

The following example illustrates how to configure the dedicated XML agent to receive and send messages via VRF1, VRF2 and the default VRF:

```
RP/0/RP0/CPU0:router(config)# xml agent
RP/0/RP0/CPU0:router(config-xml-agent)# vrf VRF1
RP/0/RP0/CPU0:router(config-xml-agent)# vrf VRF2
```

The following example illustrates how to remove access to VRF2 from the dedicated agent:

```
RP/0/RP0/CPU0:router(config)# xml agent ssl
RP/0/RP0/CPU0:router(config-xml-ssl)# vrf VRF1
RP/0/RP0/CPU0:router(config-xml-ssl-vrf)# vrf VRF2

RP/0/RP0/CPU0:router(config)# xml agent
RP/0/RP0/CPU0:router(config-xml-agent)# no vrf VRF1
```

The following example shows how to configure the XML SSL agent to receive and send messages through VRF1, VRF2 and the default VRF:

```
RP/0/RP0/CPU0:router(config)# xml agent ssl
RP/0/RP0/CPU0:router(config-xml-agent)# vrf VRF1
RP/0/RP0/CPU0:router(config-xml-agent)# vrf VRF2
```

The following example removes access for VRF2 from the dedicated XML agent:

```
RP/0/RP0/CPU0:router(config)# xml agent ssl
RP/0/RP0/CPU0:router(config-xml-agent)# no vrf VRF2
```




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 7](#)
- [Information About Implementing Physical and Virtual Terminals, on page 7](#)
- [How to Implement Physical and Virtual Terminals on Cisco IOS XR Software, on page 10](#)
- [Configuration Examples for Implementing Physical and Virtual Terminals, on page 14](#)

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 NCS 5500 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.



Note Before creating or modifying the vty pools, enable the telnet server using the **telnet server** command in XR Config mode. See Cisco IOS XR IP Addresses and Services Configuration Guide and Cisco IOS XR IP Addresses and Services Command Reference 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
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	line {console default} Example: 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"> • console —Enters line template configuration mode for the console template. • default —Enters line template configuration mode for the default line template.
Step 3	Configure the terminal attribute settings for the specified template using the commands in line template configuration mode.	—
Step 4	Use one of the following commands: <ul style="list-style-type: none"> • end • commit Example: RP/0/RP0/CPU0:router(config-line)# end	Saves configuration changes. <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre>

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

Creating and Modifying vty Pools

This task explains how to create and modify vty pools.

You can omit Step 3 to Step 5 (**line template** and **exit** commands) if you are configuring the default line template to reference a vty pool.

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

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

	Command or Action	Purpose
Step 7	Use the commit or end command.	<p>commit —Saves the configuration changes and remains within the configuration session.</p> <p>end —Prompts user to take one of these actions:</p> <ul style="list-style-type: none"> • 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.

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

	Command or Action	Purpose
		<ul style="list-style-type: none"> Specifying the <code>show line vty number EXEC</code> command displays the terminal parameters for the specified vty.
Step 2	(Optional) show terminal Example: <pre>RP/0/RP0/CPU0:router# show terminal</pre>	Displays the terminal attribute settings for the current terminal line.
Step 3	(Optional) show users Example: <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:

```
RP/0/RP0/CPU0:router# show running-config line console
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:router# show line console location 0/0/CPU0
Tue Nov 24 03:10:24.656 UTC
Tty          Speed      Overruns      Acc I/O
*con0/0/CPU0 9600        0/0          -/-

Line "con0_RP1_CPU0", Location "0/RP1/CPU0", Type "Console"
Length: 24 lines, Width: 80 columns
Baud rate (TX/RX) is 9600, "No" Parity, 2 stopbits, 8 databits
Template: console
Capabilities: Timestamp Enabled
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 100 105 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 17](#)
- [Restrictions for SNMP use on Cisco IOS XR Software, on page 17](#)
- [Information about Implementing SNMP, on page 18](#)
- [Session MIB support on subscriber sessions , on page 24](#)
- [How to Implement SNMP on Cisco IOS XR Software, on page 25](#)
- [SNMP Context Mapping Configuration, on page 45](#)

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.

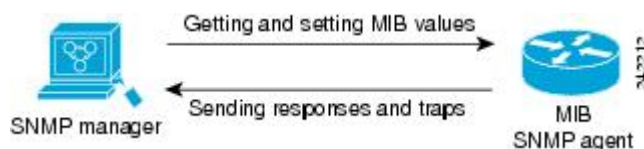
MIB

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





Note A few exceptions while performing SNMP walk on the NC55-6X200-DWDM-S line card on the NCS 5500 Series Routers are as follows:

1. Though the below mentioned OIDs are valid, they are marked as inaccessible in the OTN MIB standard. Hence they will not be polled during MIB walk.
 - OtnNearEndCurIntervalType : .1.3.6.1.4.1.9.9.639.1.2.3.1.1
 - OtnNearEndCurrentMonType : .1.3.6.1.4.1.9.9.639.1.2.3.1.2
 - OtnFarEndCurIntervalType : .1.3.6.1.4.1.9.9.639.1.2.4.1.1
 - OtnFarEndCurrentMonType : .1.3.6.1.4.1.9.9.639.1.2.4.1.2
2. OtnStatus : .1.3.6.1.4.1.9.9.639.1.1.1.1.5 OID is implicitly enabled for the interfaces of NC55-6X200-DWDM-S line card. Hence a MIB walk corresponding to the OtnStatus is not supported.

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.

From Release 6.3.2 onwards, 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 from Release 6.3.2 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 21](#) 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

Feature	SNMP v1	SNMP v2c	SNMP v3
Inform Operation	No	Yes (No on the Cisco IOS XR software)	Yes (No on the Cisco IOS XR software)
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 ¹ -MD5 ² algorithm or the HMAC-SHA ³ .
v3	authPriv	HMAC-MD5 or HMAC-SHA	DES	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides DES ⁴ 56-bit encryption in addition to authentication based on the CBC ⁵ DES (DES-56) standard.

Model	Level	Authentication	Encryption	What Happens
v3	authPriv	HMAC-MD5 or HMAC-SHA	3DES	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides 168-bit 3DES ⁶ 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 ⁷ level of encryption.

¹ Hash-Based Message Authentication Code

² Message Digest 5

³ Secure Hash Algorithm

⁴ Data Encryption Standard

⁵ Cipher Block Chaining

⁶ Triple Data Encryption Standard

⁷ Advanced Encryption Standard

Use of 3DES and AES encryption standards requires that the security package (k9sec) 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

Security Model	Security Level
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.

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.



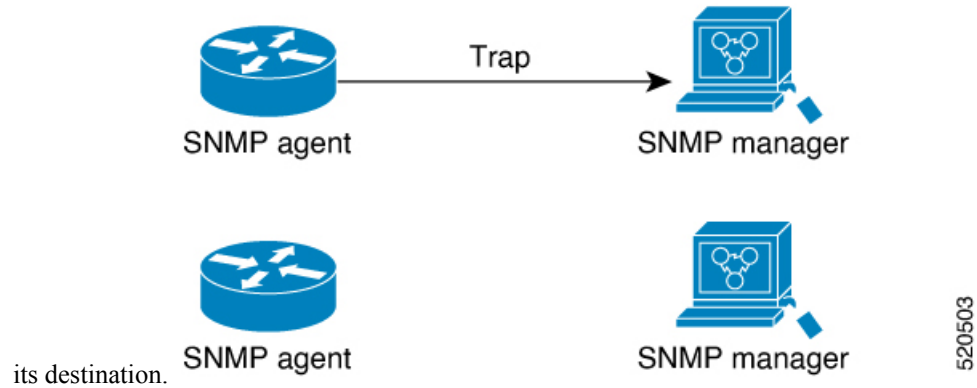
Note Inform requests (inform operations) are supported in Cisco IOS XR software.

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.

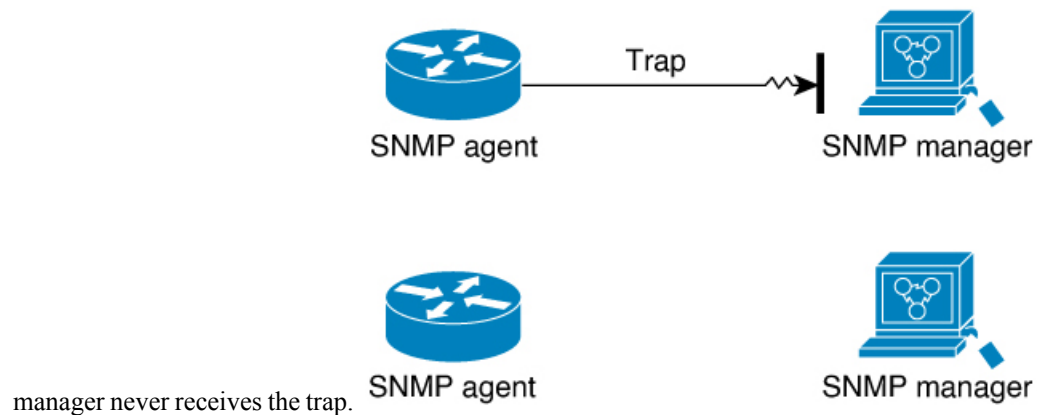
However, traps are often preferred because informs consume more resources in the router and in the network. Unlike a trap, which is discarded as soon as it is sent, an inform request must be held in memory until a response is received or the request times out. Also, traps are sent only once, and an inform 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 NCS 5500 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.

SUMMARY STEPS

1. **configure**
2. (Optional) **snmp-server engineid local** *engine-id*
3. (Optional) **snmp-server vrf** *vrf-name*
4. **snmp-server view** *view-name oid-tree* {**included** | **excluded**}
5. **snmp-server group** *name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *view*] [**write** *view*] [**notify** *view*] [*access-list-name*]
6. **snmp-server user** *username groupname* {**v1** | **v2c** | **v3** [**auth** {**md5** | **sha**} {**clear** | **encrypted**} *auth-password* [**priv** **des56** {**clear** | **encrypted**} *priv-password*]]} [*access-list-name*]
7. Use the **commit** or **end** command.
8. (Optional) **show snmp**
9. (Optional) **show snmp engineid**
10. (Optional) **show snmp group**
11. (Optional) **show snmp users**
12. (Optional) **show snmp view**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	(Optional) snmp-server engineid local <i>engine-id</i> Example: RP/0/RP0/CPU0:router(config)# 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	(Optional) snmp-server vrf <i>vrf-name</i> Example:	Configures VRF properties of SNMP.

	Command or Action	Purpose
	RP/0/RP0/CPU0:router(config)# snmp-server vrf vrfname	
Step 4	snmp-server view <i>view-name oid-tree {included excluded}</i> Example: RP/0/RP0/CPU0:router(config)# snmp-server view view_name 1.3.6.1.2.1.1.5 included	Creates or modifies a view record.
Step 5	snmp-server group <i>name {v1 v2c v3 {auth noauth priv}}</i> [<i>read view</i>] [<i>write view</i>] [<i>notify view</i>] [<i>access-list-name</i>] Example: RP/0/RP0/CPU0:router(config)# 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 6	snmp-server user <i>username groupname {v1 v2c v3 {auth {md5 sha} {clear encrypted} auth-password [priv des56 {clear encrypted} priv-password]]} [access-list-name]</i> Example: RP/0/RP0/CPU0:router(config)# snmp-server user noauthuser group_name v3	Configures a new user to an SNMP group.
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: <ul style="list-style-type: none"> • 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 Example: RP/0/RP0/CPU0:router# show snmp	Displays information about the status of SNMP.
Step 9	(Optional) show snmp engineid Example: RP/0/RP0/CPU0:router# show snmp engineid	Displays information about the local SNMP engine.

	Command or Action	Purpose
Step 10	(Optional) show snmp group Example: RP/0/RP0/CPU0:router# show snmp group	Displays information about each SNMP group on the network.
Step 11	(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 12	(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.

SUMMARY STEPS

1. **configure**
2. **snmp-server drop unknown-user**
3. Use the **commit** or **end** command.

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration 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:

	Command or Action	Purpose
		<ul style="list-style-type: none"> • 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:

```
config
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:

```
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

vldefault 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: vldefault
```



```
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 Because 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 Because 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

This task explains how to configure the router to send SNMP trap notifications.



Note You can omit [#unique_53](#) if you have already completed the steps documented under the [#unique_53](#) task.

SUMMARY STEPS

1. **configure**
2. **snmp-server group** *name* {**v1 v2 v3** {**auth** | **noauth** | **priv**}} [**read view**] **write view**] [**notify view**] [*access-list-name*]
3. **snmp-server user** *username* *groupname* {**v1 v2c v3** {**auth** | **md5** | **sha**} {**clear** | **encrypted**} *auth-password*] [**priv des56** {**clear** | *access-list-name*]
4. [**snmp-server host** *address* [**traps**] [**version** {**1** | **2c** | **3** [**auth** | **priv**}}] *community-string* [**udp-port port**] [*notification-type*]
5. **snmp-server traps** [*notification-type*]
6. Use the **commit** or **end** command.
7. (Optional) **show snmp host**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	snmp-server group <i>name</i> { v1 v2 v3 { auth noauth priv }} [read view] write view] [notify view] [<i>access-list-name</i>] Example: RP/0/RP0/CPU0:router(config)# snmp-server group <i>group_name</i> <i>v3</i> 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 <i>username</i> <i>groupname</i> { v1 v2c v3 { auth md5 sha } { clear encrypted } <i>auth-password</i>] [priv des56 { clear <i>access-list-name</i>] Example: RP/0/RP0/CPU0:router(config)# snmp-server group <i>group_name</i> <i>v3</i> 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 host <i>address</i> [traps] [version { 1 2c 3 [auth priv }}] <i>community-string</i> [udp-port port] [<i>notification-type</i>] Example:	Specifies SNMP trap notifications, the version of SNMP to use, the security level of the notifications, and the recipient (host) of the notifications.

	Command or Action	Purpose
	RP/0/RP0/CPU0:router(config)# snmp-server host 12.26.25.61 traps version 3 noauth userV3noauth	
Step 5	snmp-server traps [<i>notification-type</i>] 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. <ul style="list-style-type: none"> • If a trap is not specified with the <i>notification-type</i> 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 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: <ul style="list-style-type: none"> • 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 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.

SUMMARY STEPS

1. **configure**
2. **snmp-server drop unknown-user**
3. Use the **commit** or **end** command.

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration 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: <ul style="list-style-type: none"> • 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
!
```

In the following example, the output of the **show snmp host** command 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

```
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.

SUMMARY STEPS

1. **configure**
2. (Optional) **snmp-server contact** *system-contact-string*
3. (Optional) **snmp-server location** *system-location*
4. (Optional) **snmp-server chassis-id** *serial-number*
5. Use the **commit** or **end** command.

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	(Optional) snmp-server contact <i>system-contact-string</i> 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 <i>system-location</i> 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 <i>serial-number</i> 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: <ul style="list-style-type: none"> • 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.

SUMMARY STEPS

1. **configure**
2. (Optional) **snmp-server packetsize** *byte-count*
3. Use the **commit** or **end** command.

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	(Optional) snmp-server packetsize <i>byte-count</i> 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: <ul style="list-style-type: none"> • 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.

SUMMARY STEPS

1. **configure**
2. (Optional) **snmp-server trap-source** *type interface-path-id*
3. (Optional) **snmp-server queue-length** *length*

4. (Optional) **snmp-server trap-timeout** *seconds*
5. Use the **commit** or **end** command.

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# <code>configure</code>	Enters global configuration mode.
Step 2	(Optional) snmp-server trap-source <i>type interface-path-id</i> Example: RP/0/RP0/CPU0:router(config)# <code>snmp-server trap-source POS 0/0/1/0</code>	Specifies a source interface for trap notifications.
Step 3	(Optional) snmp-server queue-length <i>length</i> Example: RP/0/RP0/CPU0:router(config)# <code>snmp-server queue-length 20</code>	Establishes the message queue length for each notification.
Step 4	(Optional) snmp-server trap-timeout <i>seconds</i> Example: RP/0/RP0/CPU0:router(config)# <code>snmp-server trap-timeout 20</code>	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: <ul style="list-style-type: none"> • 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 IP Precedence or IP DSCP for SNMP traffic.

Before you begin

SNMP must be configured.

SUMMARY STEPS

1. **configure**
2. Use one of the following commands:
 - **snmp-server ipv4 precedence** *value*
 - **snmp-server ipv4 dscp** *value*
3. Use the **commit** or **end** command.

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: <pre>RP/0/RP0/CPU0:router# configure</pre>	Enters global configuration mode.
Step 2	Use one of the following commands: <ul style="list-style-type: none"> • snmp-server ipv4 precedence <i>value</i> • snmp-server ipv4 dscp <i>value</i> Example: <pre>RP/0/RP0/CPU0:router(config)# snmp-server dscp 24</pre>	Configures an IP precedence or IP 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: <ul style="list-style-type: none"> • 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 IP Precedence value to 7:

```
configure
 snmp-server ipv4 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 IP 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
```

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.

SUMMARY STEPS

1. **show snmp context-mapping**

DETAILED STEPS

	Command or Action	Purpose
Step 1	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.

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	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, an 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.

SUMMARY STEPS

1. (Optional) **snmp-server entityindex persist**
2. (Optional) **snmp-server mibs cbqosmib persist**

3. (Optional) **snmp-server cbqosmib cache refresh time** *time*
4. (Optional) **snmp-server cbqosmib cache service-policy count** *count*
5. **snmp-server ifindex persist**

DETAILED STEPS

	Command or Action	Purpose
Step 1	(Optional) snmp-server entityindex persist Example: <pre>RP/0/RP0/CPU0:router(config)# snmp-server entityindex persist</pre>	Enables the persistent storage of ENTITY-MIB data.
Step 2	(Optional) snmp-server mibs cbqosmib persist Example: <pre>RP/0/RP0/CPU0:router(config)# snmp-server mibs cbqosmib persist</pre>	Enables persistent storage of the CISCO-CLASS-BASED-QOS-MIB data.
Step 3	(Optional) snmp-server cbqosmib cache refresh time <i>time</i> Example: <pre>RP/0/RP0/CPU0:router(config)# snmp-server mibs cbqosmib cache refresh time 45</pre>	Enables QoS MIB caching with a specified cache refresh time.
Step 4	(Optional) snmp-server cbqosmib cache service-policy count <i>count</i> Example: <pre>RP/0/RP0/CPU0:router(config)# snmp-server mibs cbqosmib cache service-policy count 50</pre>	Enables QoS MIB caching with a limited number of service policies to cache.
Step 5	snmp-server ifindex persist Example: <pre>RP/0/RP0/CPU0:router(config)# snmp-server ifindex persist</pre>	Enables ifIndex 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.

SUMMARY STEPS

1. **configure**
2. **snmp-server interface subset** *subset-number* **regular-expression** *expression*
3. **notification linkupdown disable**
4. Use the **commit** or **end** command.
5. (Optional) **show snmp interface notification subset** *subset-number*
6. (Optional) **show snmp interface notification regular-expression** *expression*
7. (Optional) **show snmp interface notification type** *interface-path-id*

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	snmp-server interface subset <i>subset-number</i> regular-expression <i>expression</i> 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 <i>subset-number</i> 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 <i>expression</i> argument must be entered surrounded by double quotes. Refer to the <i>Understanding Regular Expressions, Special Characters, and Patterns</i> module in for more information regarding regular expressions.
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: <ul style="list-style-type: none"> • 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.

	Command or Action	Purpose
Step 5	(Optional) show snmp interface notification subset <i>subset-number</i> Example: <pre>RP/0/RP0/CPU0:router# show snmp interface notification subset 10</pre>	Displays the linkUp and linkDown notification status for all interfaces identified by the subset priority.
Step 6	(Optional) show snmp interface notification regular-expression <i>expression</i> Example: <pre>RP/0/RP0/CPU0:router# show snmp interface notification regular-expression "^Gig[a-zA-Z]+[0-9/]+\."</pre>	Displays the linkUp and linkDown notification status for all interfaces identified by the regular expression.
Step 7	(Optional) show snmp interface notification type <i>interface-path-id</i> Example: <pre>RP/0/RP0/CPU0:router# show snmp interface notification tengige 0/4/0/3.10</pre>	Displays the linkUp and linkDown notification status for the specified interface.

SNMP Context Mapping Configuration

Configuration of VRF Aware SNMP Context for Polling BGP Data

VRF awareness is usually done using existing, non-VRF aware MIB definitions. This means that MIB definition doesn't mention anything about VRFs. However they could be used within VRF context.

The VRF-awareness is done using SNMP contexts, where a SNMP context maps to a specific VRF.

Before you begin

- Ensure that MIB implementation is VRF-aware.
- Ensure that the implementation of all get requests support VRF context.

The following example configures VRF aware SNMP context to allow polling BGP data using BGP4-MIB.

```
snmp-server vrf <vrf_1> context <context_1>
snmp-server community <vrf_1> RW
snmp-server context <context_1>
snmp-server community-map <vrf_1> context <context_1>
snmp-server host <IP> traps version 2c <vrf_1>
```

Verification

The following configuration extracts BGP data from a peer VRF using context.

```
snmp-server vrf V1
  context V1_bgp
!
snmp-server community V1 RW
```

```

snmp-server context V1_bgp
snmp-server community-map V1 context V1_bgp
router bgp 65000
  nsr
  address-family ipv4 unicast
  !
  address-family vpnv4 unicast
  !
  neighbor 192.0.2.254
    remote-as 65001
    address-family ipv4 unicast
      route-policy ALL in
      route-policy ALL out
    !
  !
vrf V1
  rd 111:111
  address-family ipv4 unicast
  !
  neighbor 192.0.2.255
    remote-as 65003
    address-family ipv4 unicast
  !
!
!
end

```

Configuration of OSPF processes Using SNMP Context

The following example configures data polling from two OSPF processes.

```

snmp-server community com1 RW
snmp-server community com2 RW
snmp-server context ctx1
snmp-server context ctx2
snmp-server community-map com1 context ctx1
snmp-server community-map com2 context ctx2
router ospf one
  snmp context ctx1
  area 0
    interface GigabitEthernet0/2/0/0
    !
  !
!
router ospf two
  snmp context ctx2
  area 0
    interface GigabitEthernet0/2/0/1
    !
  !
!

```

Configuration of OSPF Neighbour in VRF

The following example configures OSPF neighbours in VRF using SNMP context.

```

snmp-server vrf VRF_A
  context ctx1
!
snmp-server community com1 RW
snmp-server context ctx1
snmp-server community-map com1 context ctx1
router ospf core

```



```
vrf VRF_A
  snmp context ctx1
!
end
```




CHAPTER 5

Configuring Object Tracking

This module describes the configuration of object tracking on your Cisco IOS XR network. For complete descriptions of the commands listed in this module, see **Additional References** section. To locate documentation for other commands that might appear in the course of performing a configuration task, see **Technical Documentation** section in the Additional References topic.

- [Configuring Object Tracking, on page 49](#)
- [Prerequisites for Implementing Object Tracking, on page 49](#)
- [Information about Object Tracking, on page 50](#)
- [How to Implement Object Tracking, on page 50](#)
- [Configuration Examples for Configuring Object Tracking, on page 61](#)
- [Additional References, on page 63](#)

Configuring Object Tracking

This module describes the configuration of object tracking on your Cisco IOS XR network. For complete descriptions of the commands listed in this module, see **Additional References** section. To locate documentation for other commands that might appear in the course of performing a configuration task, see **Technical Documentation** section in the Additional References topic.

Prerequisites for Implementing Object Tracking

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.



Note

Object Tracking is an optional package. You must check if this package is installed on your system by running the command **show install active summary**.

Information about Object Tracking

Object tracking is a mechanism to track an object and to take an action on another object with no relationship to the tracked objects, based on changes to the properties of the object being tracked.

Each tracked object is identified by a unique name specified on the tracking command-line interface (CLI). Cisco IOS XR processes then use this name to track a specific object.

The tracking process periodically polls the tracked object and reports any changes to its state in terms of its being up or down, either immediately or after a delay, as configured by the user.

Multiple objects can also be tracked by means of a list, using a flexible method for combining objects with Boolean logic. This functionality includes:

- **Boolean AND function**—When a tracked list has been assigned a Boolean AND function, each object defined within a subset must be in an up state, so that the tracked object can also be in the up state.
- **Boolean OR function**—When the tracked list has been assigned a Boolean OR function, it means that at least one object defined within a subset must also be in an up state, so that the tracked object can also be in the up state.

How to Implement Object Tracking

This section describes the various object tracking procedures.

Tracking the Line Protocol State of an Interface

Perform this task in global configuration mode to track the line protocol state of an interface.

A tracked object is considered up when a line protocol of the interface is up.

After configuring the tracked object, you may associate the interface whose state should be tracked and specify the number of seconds to wait before the tracking object polls the interface for its state.

SUMMARY STEPS

1. **configure**
2. **track** *track-name*
3. **type line-protocol state**
4. **interface** *type interface-path-id*
5. **exit**
6. (Optional) **delay** {**up** *seconds* | **down** *seconds*}
7. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	track track-name Example: RP/0/RP0/CPU0:router(config)# track track1	Enters track configuration mode. <ul style="list-style-type: none"> • <i>track-name</i>—Specifies a name for the object to be tracked. Note Special characters are not allowed in a <i>track-name</i> .
Step 3	type line-protocol state Example: RP/0/RP0/CPU0:router(config-track)# type line-protocol state	Creates a track based on the line protocol of an interface.
Step 4	interface type interface-path-id Example: RP/0/RP0/CPU0:router(config-track-line-prot)# interface atm 0/2/0/0.1	Specifies the interface to track the protocol state. <ul style="list-style-type: none"> • <i>type</i>—Specifies the interface type. For more information, use the question mark (?) online help function. • <i>interface-path-id</i>—Identifies a physical interface or a virtual interface. Note Use the show interfaces command to see a list of all possible interfaces currently configured on the router. Note The loopback and null interfaces are always in the up state and, therefore, cannot be tracked.
Step 5	exit Example: RP/0/RP0/CPU0:router(config-track-line-prot)# exit	Exits the track line protocol configuration mode.
Step 6	(Optional) delay {up seconds down seconds} Example: RP/0/RP0/CPU0:router(config-track)# delay up 10	Schedules the delay that can occur between tracking whether the object is up or down.
Step 7	Use one of the following commands:	Saves configuration changes.

Command or Action	Purpose
<ul style="list-style-type: none"> • end • commit <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-track)# end</pre> <p>or</p> <pre>RP/0/RP0/CPU0:router(config-track)# commit</pre>	<ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Tracking IP Route Reachability

When a host or a network goes down on a remote site, routing protocols notify the router and the routing table is updated accordingly. The routing process is configured to notify the tracking process when the route state changes due to a routing update.

A tracked object is considered up when a routing table entry exists for the route and the route is accessible.

SUMMARY STEPS

1. **configure**
2. **track** *track-name*
3. **type route reachability**
4. Use one of the following commands:
 - **vrf** *vrf-table-name*
 - **route ipv4** *IP-prefix/mask*
5. **exit**
6. (Optional) **delay** {**up** *seconds* | **down** *seconds*}
7. Use the **commit** or **end** command.

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: <pre>RP/0/RP0/CPU0:router# configure</pre>	Enters global configuration mode.
Step 2	track track-name Example: <pre>RP/0/RP0/CPU0:router(config)# track track1</pre>	Enters track configuration mode. <ul style="list-style-type: none"> • <i>track-name</i>—Specifies a name for the object to be tracked. <p>Note Special characters are not allowed in a <i>track-name</i>.</p>
Step 3	type route reachability Example: <pre>RP/0/RP0/CPU0:router(config-track)# type route reachability vrf internet</pre>	Configures the routing process to notify the tracking process when the state of the route changes due to a routing update.
Step 4	Use one of the following commands: <ul style="list-style-type: none"> • vrf vrf-table-name • route ipv4 IP-prefix/mask Example: <pre>RP/0/RP0/CPU0:router(config-track-route)# vrf vrf-table-4</pre> <p>or</p> <pre>RP/0/RP0/CPU0:router(config-track-route)# route ipv4 10.56.8.10/16</pre>	Configures the type of IP route to be tracked, which can consist of either of the following, depending on your router type: <ul style="list-style-type: none"> • <i>vrf-table-name</i>—A VRF table name. • <i>IP-prefix/mask</i>—An IP prefix consisting of the network and subnet mask (for example, 10.56.8.10/16).
Step 5	exit Example: <pre>RP/0/RP0/CPU0:router(config-track-line-prot)# exit</pre>	Exits the track line protocol configuration mode.
Step 6	(Optional) delay {up seconds down seconds} Example: <pre>RP/0/RP0/CPU0:router(config-track)# delay up 10</pre>	Schedules the delay that can occur between tracking whether the object is up or down.
Step 7	Use the commit or end command.	<p>commit —Saves the configuration changes, and remains within the configuration session.</p> <p>end —Prompts user to take one of these actions:</p> <ul style="list-style-type: none"> • Yes — Saves configuration changes and exits the configuration session.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • No —Exits the configuration session without committing the configuration changes. • Cancel —Remains in the configuration mode, without committing the configuration changes.

Building a Track Based on a List of Objects

Perform this task in the global configuration mode to create a tracked list of objects (which, in this case, are lists of interfaces or prefixes) using a Boolean expression to determine the state of the list.

A tracked list contains one or more objects. The Boolean expression enables two types of calculations by using either AND or OR operators. For example, when tracking two interfaces, using the AND operator, up means that *both* interfaces are up, and down means that *either* interface is down.



Note An object must exist before it can be added to a tracked list.

The NOT operator is specified for one or more objects and negates the state of the object.

After configuring the tracked object, you must associate the interface whose state should be tracked and you may optionally specify the number of seconds to wait before the tracking object polls the interface for its state.

SUMMARY STEPS

1. **configure**
2. **track** *track-name*
3. **type list boolean { and | or }**
4. **object** *object-name* [**not**]
5. **exit**
6. (Optional) **delay {up seconds|down seconds}**
7. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	track <i>track-name</i>	Enters track configuration mode.

	Command or Action	Purpose
	Example: RP/0/RP0/CPU0:router(config)# track track1	<ul style="list-style-type: none"> • <i>track-name</i>—Specifies a name for the object to be tracked. <p>Note Special characters are not allowed in a <i>track-name</i>.</p>
Step 3	type list boolean { and or } Example: RP/0/RP0/CPU0:router(config-track)# type list boolean and	Configures a Boolean list object and enters track list configuration mode. <ul style="list-style-type: none"> • boolean—Specifies that the state of the tracked list is based on a Boolean calculation. • and—Specifies that the list is up if all objects are up, or down if one or more objects are down. For example when tracking two interfaces, up means that both interfaces are up, and down means that either interface is down. • or—Specifies that the list is up if at least one object is up. For example, when tracking two interfaces, up means that either interface is up, and down means that both interfaces are down.
Step 4	object object-name [not] Example: RP/0/RP0/CPU0:router(config-track-list)# object 3 not	Specifies the object to be tracked by the list <ul style="list-style-type: none"> • <i>object-name</i>—Name of the object to track. • not—Negates the state of the object.
Step 5	exit Example: RP/0/RP0/CPU0:router(config-track-line-prot)# exit	Exits the track line protocol configuration mode.
Step 6	(Optional) delay {up seconds down seconds} Example: RP/0/RP0/CPU0:router(config-track)# delay up 10	Schedules the delay that can occur between tracking whether the object is up or down.
Step 7	Use one of the following commands: <ul style="list-style-type: none"> • end • commit Example: RP/0/RP0/CPU0:router(config-track)# end or RP/0/RP0/CPU0:router(config-track)# commit	Saves configuration changes. <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:</pre> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Building a Track Based on a List of Objects - Threshold Percentage

Perform this task in the global configuration mode to create a tracked list of objects (which, in this case, are lists of interfaces or prefixes) using a threshold percentage to determine the state of the list.

SUMMARY STEPS

1. **configure**
2. **track** *track-name*
3. **type list threshold percentage**
4. **object** *object-name*
5. **threshold percentage up percentage down percentage**
6. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	track <i>track-name</i> Example: RP/0/RP0/CPU0:router(config)# track track1	Enters track configuration mode. <ul style="list-style-type: none"> • <i>track-name</i>—Specifies a name for the object to be tracked. <p>Note Special characters are not allowed in a <i>track-name</i>.</p>

	Command or Action	Purpose
Step 3	type list threshold percentage Example: <pre>RP/0/RP0/CPU0:router(config-track)# type list threshold percentage</pre>	Configures a track of type threshold percentage list.
Step 4	object object-name Example: <pre>RP/0/RP0/CPU0:router(config-track-list-threshold)# object 1 RP/0/RP0/CPU0:router(config-track-list-threshold)# object 2 RP/0/RP0/CPU0:router(config-track-list-threshold)# object 3 RP/0/RP0/CPU0:router(config-track-list-threshold)# object 4</pre>	Configures object 1, object 2, object 3 and object 4 as members of track type track1.
Step 5	threshold percentage up percentage down percentage Example: <pre>RP/0/RP0/CPU0:router(config-track-list-threshold)# threshold percentage up 50 down 33</pre>	<p>Configures the percentage of objects that need to be UP or DOWN for the list to be considered UP or Down respectively.</p> <p>For example, if object 1, object 2, and object 3 are in the UP state and object 4 is in the DOWN state, the list is considered to be in the UP state.</p>
Step 6	<p>Use one of the following commands:</p> <ul style="list-style-type: none"> • end • commit Example: <pre>RP/0/RP0/CPU0:router(config-track)# end</pre> <p>or</p> <pre>RP/0/RP0/CPU0:router(config-track)# commit</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Building a Track Based on a List of Objects - Threshold Weight

Perform this task in the global configuration mode to create a tracked list of objects (which, in this case, are lists of interfaces or prefixes) using a threshold weight to determine the state of the list.

SUMMARY STEPS

1. **configure**
2. **track** *track-name*
3. **type list threshold weight**
4. **object** *object-name* **weight** *weight*
5. **threshold weight up** *weight* **down** *weight*
6. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	track <i>track-name</i> Example: RP/0/RP0/CPU0:router(config)# track track1	Enters track configuration mode. <ul style="list-style-type: none"> • <i>track-name</i>—Specifies a name for the object to be tracked. <p>Note Special characters are not allowed in a <i>track-name</i>.</p>
Step 3	type list threshold weight Example: RP/0/RP0/CPU0:router(config-track)# type list threshold weight	Configures a track of type, threshold weighted list.
Step 4	object <i>object-name</i> weight <i>weight</i> Example: RP/0/RP0/CPU0:router(config-track-list-threshold)# object 1 weight 10 RP/0/RP0/CPU0:router(config-track-list-threshold)# object 2 weight 5 RP/0/RP0/CPU0:router(config-track-list-threshold)# object 3 weight 3	Configures object 1, object 2 and object 3 as members of track t1 and with weights 10, 5 and 3 respectively.

	Command or Action	Purpose
Step 5	threshold weight up weight down weight Example: <pre>RP/0/RP0/CPU0:router(config-track-list-threshold)# threshold weight up 10 down 5</pre>	Configures the range of weights for the objects that need to be UP or DOWN for the list to be considered UP or DOWN respectively. In this example, the list is considered to be in the DOWN state because objects 1 and 2 are in the UP state and the cumulative weight is 15 (not in the 10-5 range).
Step 6	Use one of the following commands: <ul style="list-style-type: none"> • end • commit Example: <pre>RP/0/RP0/CPU0:router(config-track)# end</pre> or <pre>RP/0/RP0/CPU0:router(config-track)# commit</pre>	Saves configuration changes. <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Tracking IPSLA Reachability

Use this task to enable the tracking of the return code of IP service level agreement (SLA) operations.

SUMMARY STEPS

1. **configure**
2. **track track-name**
3. **type rtr ipsla-no reachability**
4. Use the **commit** or **end** command.

DETAILED STEPS

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

	Command or Action	Purpose
	RP/0/RP0/CPU0:router# configure	
Step 2	track track-name Example: RP/0/RP0/CPU0:router(config)# track t1	Enters track configuration mode. Note Special characters are not allowed in a <i>track-name</i> .
Step 3	type rtr ipsla-no reachability Example: RP/0/RP0/CPU0:router(config-track)# type rtr 100 reachability	Specifies the IP SLA operation ID to be tracked for reachability. Values for the <i>ipsla-no</i> can range from 1 to 2048.
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: <ul style="list-style-type: none"> • 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 IPSLA Tracking: Example

This example shows the configuration of IPSLA tracking:

```
RP/0/RP0/CPU0:router(config)# track track1
RP/0/RP0/CPU0:router(config-track)# type rtr 1 reachability
RP/0/RP0/CPU0:router(config-track)# delay up 5
RP/0/RP0/CPU0:router(config-track)# delay down 10
```

Tracking BGP Neighbor Address-Family State

Perform this task in global configuration mode to track the BGP neighbor address-family state.

By tracking this object, you can detect the connectivity state of the neighbor . A tracked object is up when a state of the neighbor is up. Associate the neighbor whose state must be tracked after configuring the tracked object. Based on the state of the neighbor you can reroute the traffic.

Configuration Example

```
/* Track neighbor address-family state */
Router# configure
Router(config)# track neighbor-A
Router(config-track)# type bgp neighbor address-family state
```

```
Router(config-track-bgp-nbr-af)# address-family 12vpn evpn
Router(config-track-bgp-neighbor)# neighbor 172.16.0.1
Router(config-track-bgp-neighbor)# exit
```

Configuration Examples for Configuring Object Tracking

Tracking Whether the Interface Is Up or Down: Running Configuration Example

```
track connection100
  type list boolean and
  object object3 not
  delay up 10
!
interface service-ipsec 23
  line-protocol track connection100
!
```

Tracking the Line Protocol State of an Interface: Running Configuration Example

In this example, traffic arrives from interface service-ipsec1 and exits through interface gigabitethernet0/0/0/3:

```
track IPSec1
  type line-protocol state
  interface gigabitethernet0/0/0/3
!
interface service-ipsec 1
  ipv4 address 70.0.0.1 255.255.255.0
  profile vrfl_profile_ipsec
  line-protocol track IPSec1
  tunnel source 80.0.0.1
  tunnel destination 80.0.0.2
  service-location preferred-active 0/0/1
!
```

This example displays the output from the **show track** command after performing the previous example:

```
RP/0/RP0/CPU0:router# show run track

Track IPSec1
Interface GigabitEthernet0_0_0_3 line-protocol
!
Line protocol is UP
1 change, last change 10:37:32 UTC Thu Sep 20 2007
Tracked by:
service-ipsec1
!
```

Tracking IP Route Reachability: Running Configuration Example

In this example, traffic arriving from interface service-ipsec1 has its destination in network 7.0.0.0/24. This tracking procedure follows the state of the routing protocol prefix to signal when there are changes in the routing table.

```
track PREFIX1
  type route reachability
  route ipv4 7.0.0.0/24
  !
  interface service-ipsec 1
  vrf 1
  ipv4 address 70.0.0.2 255.255.255.0
  profile vrf_1_ipsec
  line-protocol track PREFIX1
  tunnel source 80.0.0.2
  tunnel destination 80.0.0.1
  service-location preferred-active 0/2/0
```

Building a Track Based on a List of Objects: Running Configuration Example

In this example, traffic arriving from interface service-ipsec1 exits through interface gigabitethernet0/0/0/3 and interface ATM 0/2/0/0.1. The destination of the traffic is at network 7.0.0.0/24.

If either one of the interfaces or the remote network goes down, the flow of traffic must stop. To do this, we use a Boolean AND expression.

```
track C1
  type route reachability
  route ipv4 3.3.3.3/32
  !
  !
track C2
  type route reachability
  route ipv4 1.2.3.4/32
  !
  !
track C3
  type route reachability
  route ipv4 10.0.20.2/32
  !
  !
track C4
  type route reachability
  route ipv4 10.0.20.0/24
  !
  !
track OBJ
  type list boolean and
  object C1
  object C2
  !
  !
track OBJ2
  type list boolean or
  object C1
```



```
object C2
!
```

Configuring IPSLA based Object Tracking: Configuration Example

This example shows the configuration of IPSLA based object tracking, including the ACL and IPSLA configuration:

ACL configuration:

```
RP/0/RP0/CPU0:router(config)# ipv4 access-list abf-track
RP/0/RP0/CPU0:router(config-ipv4-acl)# 10 permit any nexthop track track1 1.2.3.4
```

Object tracking configuration:

```
RP/0/RP0/CPU0:router(config)# track track1
RP/0/RP0/CPU0:router(config-track)# type rtr 1 reachability
RP/0/RP0/CPU0:router(config-track)# delay up 5
RP/0/RP0/CPU0:router(config-track)# delay down 10
```

IPSLA configuration:

```
RP/0/RP0/CPU0:router(config)# ipsla
RP/0/RP0/CPU0:router(config-ipsla)# operation 1
RP/0/RP0/CPU0:router(config-ipsla-op)# type icmp echo
RP/0/RP0/CPU0:router(config-ipsla-icmp-echo)# source address 2.3.4.5
RP/0/RP0/CPU0:router(config-ipsla-icmp-echo)# destination address 1.2.3.4
RP/0/RP0/CPU0:router(config-ipsla-icmp-echo)# frequency 60
RP/0/RP0/CPU0:router(config-ipsla-icmp-echo)# exit
RP/0/RP0/CPU0:router(config-ipsla-op)# exit
RP/0/RP0/CPU0:router(config-ipsla)# schedule operation 1
RP/0/RP0/CPU0:router(config-ipsla-sched)# start-time now
RP/0/RP0/CPU0:router(config-ipsla-sched)# life forever
```

Additional References

The following sections provide references related to implementing object tracking for IPSec network security.

Related Documents

Related Topic	Document Title
IP SLA configuration information	<i>Implementing IP Service Level Agreements on module in System Monitoring Configuration Guide for Cisco NCS 5500 Series Routers</i>
IP SLA commands	<i>IP Service Level Agreement Commands on module in System Monitoring Command Reference for Cisco NCS 5500 Series Routers and Cisco NCS 540 and NCS 560 Series Routers</i>
Object tracking commands	<i>Object Tracking Commands on module in</i>

Standards

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

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: https://cfnng-stg.cisco.com/mibs .

RFCs

RFCs	Title
RFC 2401	<i>Security Architecture for the Internet Protocol</i>

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/cisco/web/support/index.html



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 65](#)
- [Information About Implementing CDP, on page 65](#)
- [How to Implement CDP on Cisco IOS XR Software, on page 67](#)
- [Configuration Examples for Implementing CDP, on page 73](#)

Prerequisites for Implementing CDP

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.



Note CDP is an optional package. You must check if this package is installed on your system by running the command **show install active summary**.

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 5: 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 task explains how to enable CDP globally on the router and then enable CDP on an interface.

SUMMARY STEPS

1. **configure**
2. **cdp**
3. **interface** *type interface-path-id*
4. **cdp**
5. Use the **commit** or **end** command.

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: <pre>RP/0/RP0/CPU0:router# configure</pre>	Enters global configuration mode.
Step 2	cdp Example: <pre>RP/0/RP0/CPU0:router(config)# cdp</pre>	Enables CDP globally.
Step 3	interface <i>type interface-path-id</i> Example: <pre>RP/0/RP0/CPU0:router(config)# int TenGigE 0/5/0/11/1</pre>	Enters interface configuration mode.
Step 4	cdp Example: <pre>RP/0/RP0/CPU0:router(config-if)# cdp</pre>	Enables CDP on an interface.
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: <ul style="list-style-type: none"> • Yes — Saves configuration changes and exits the configuration session.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • No —Exits the configuration session without committing the configuration changes. • Cancel —Remains in the configuration session, without committing the configuration changes.

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.

SUMMARY STEPS

1. **configure**
2. **cdp advertise v1**
3. **cdp holdtime** *seconds*
4. **cdp timer** *seconds*
5. Use the **commit** or **end** command.
6. (Optional) **show cdp**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration 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. <ul style="list-style-type: none"> • 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 <i>seconds</i> 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.

	Command or Action	Purpose
		<ul style="list-style-type: none"> By default, when CDP is enabled, the receiving networking device holds a CDP packet for 180 seconds before discarding it. <p>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.</p> <ul style="list-style-type: none"> In this example, the value of hold-time for the <i>seconds</i> argument is set to 30.
Step 4	cdp timer <i>seconds</i> Example: <pre>RP/0/RP0/CPU0:router(config)# cdp timer 20</pre>	<p>Specifies the frequency at which CDP update packets are sent.</p> <ul style="list-style-type: none"> By default, when CDP is enabled, CDP update packets are sent at a frequency of once every 60 seconds. <p>Note A lower timer setting causes CDP updates to be sent more frequently.</p> <ul style="list-style-type: none"> 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.	<p>commit —Saves the configuration changes and remains within the configuration session.</p> <p>end —Prompts user to take one of these actions:</p> <ul style="list-style-type: none"> 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: <pre>RP/0/RP0/CPU0:router# show cdp</pre>	<p>Displays global CDP information.</p> <p>The output displays the CDP version running on the router, the hold time setting, and the timer setting.</p>

Monitoring CDP

This task shows how to monitor CDP.



Note The commands can be entered in any order.

SUMMARY STEPS

1. **show cdp entry** *{* | entry-name}* [**protocol** | **version**]
2. **show cdp interface** [*type interface-path-id* | **location node-id**]
3. **show cdp neighbors** [*type interface-path-id* | **location node-id**] [**detail**]
4. **show cdp traffic** [**location node-id**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	show cdp entry <i>{* entry-name}</i> [protocol version] Example: RP/0/RSP0/CPU0:router# show cdp entry *	Displays information about a specific neighboring device or all neighboring devices discovered using CDP.
Step 2	show cdp interface [<i>type interface-path-id</i> location node-id] Example: RP/0/RSP0/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 [<i>type interface-path-id</i> location node-id] [detail] Example: RP/0/RSP0/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/RSP0/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
asr9k-rtr1     Te0/5/0/11/1   152      R           ASR9K Ser  Te0/1/0/9
asr9k-rtr1     Te0/5/0/11/2   156      R           ASR9K Ser  Te0/1/0/10
asr9k-rtr1     Te0/5/0/11/3   160      R           ASR9K Ser  Te0/1/0/11
```

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 TenGigE 0/5/0/11/1 detail
```

```
-----  
Device ID: asr9k-rtr1  
SysName : asr9k-rtr1  
Entry address(es):  
IPv4 address: 90.0.0.2  
Platform: cisco ASR9K Series, Capabilities: Router  
Interface: TenGigE 0/5/0/11/1  
Port ID (outgoing port): TenGigE 0/1/0/9  
Holdtime : 155 sec  
  
Version :  
Cisco IOS XR Software, Version 5.3.1.10I[Default]  
Copyright (c) 2015 by Cisco Systems, Inc.  
  
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 asr9k-rtr1
```

```
-----  
Device ID: asr9k-rtr1  
SysName : asr9k-rtr1  
Entry address(es):  
IPv4 address: 110.0.0.2  
Platform: cisco ASR9K Series, Capabilities: Router  
Interface: TenGigE 0/5/0/11/3  
Port ID (outgoing port): TenGigE 0/1/0/11  
Holdtime : 173 sec  
  
Version :  
Cisco IOS XR Software, Version 5.3.1.10I[Default]  
Copyright (c) 2015 by Cisco Systems, Inc.  
  
advertisement version: 2  
Duplex: full  
  
-----  
Device ID: asr9k-rtr1  
SysName : asr9k-rtr1  
Entry address(es):  
IPv4 address: 100.0.0.2  
Platform: cisco ASR9K Series, Capabilities: Router  
Interface: TenGigE 0/5/0/11/2  
Port ID (outgoing port): TenGigE 0/1/0/10  
Holdtime : 169 sec  
  
Version :  
Cisco IOS XR Software, Version 5.3.1.10I[Default]  
Copyright (c) 2015 by Cisco Systems, Inc.  
  
advertisement version: 2  
Duplex: full  
  
-----  
Device ID: asr9k-rtr1  
SysName : asr9k-rtr1
```

```

Entry address(es):
IPv4 address: 90.0.0.2
Platform: cisco ASR9K Series, Capabilities: Router
Interface: TenGigE 0/5/0/11/1
Port ID (outgoing port): TenGigE 0/1/0/10
Holdtime : 165 sec

```

```

Version :
Cisco IOS XR Software, Version 5.3.1.10I[Default]
Copyright (c) 2015 by Cisco Systems, Inc.

```

```

advertisement version: 2
Duplex: full

```

The following is sample output for the **show cdp interface** command. In this example, CDP information related to Packet over SONET/SDH (POS) interface 0/4/0/0 is displayed.

```

RP/0/RP0/CPU0:router# show cdp interface TenGigE 0/5/0/11/1

TenGigE 0/5/0/11/1 is Up
  Encapsulation ether
  Sending CDP packets every 20 seconds
  Holdtime is 30 seconds

```

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

```

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

CDP counters :
  Packets output: 250, Input: 120
  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: 250, Input: 120
  Unrecognize Hdr version: 0, File open failed: 0

```

The following is sample output for the **show cdp traffic** command. In this example, the optional **location** keyword and *node-id* argument are used to display information about the traffic gathered between devices using CDP from the specified node.

```

RP/0/RP0/CPU0:router# show cdp traffic 0/5/CPU0

CDP counters :
  Packets output: 318, Input: 141
  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: 318, Input: 141
  Unrecognize Hdr version: 0, File open failed: 0

```

Configuration Examples for Implementing CDP

Enabling CDP: Example

The following example shows how to configure CDP globally and then enable CDP on Ethernet interface TenGigE 0/5/0/11/1:

```
cdp
interface 0/5/0/11/1
cdp
```

Modifying Global CDP Settings: Example

The following example shows how to modify global CDP settings. In this example, the timer setting is set to 20 seconds, the hold-time setting is set to 30 seconds, and the version of CDP used to communicate with neighboring devices is set to CDPv1:

```
cdp timer 20
cdp holdtime 30
cdp advertise v1
```

The following example shows how to use the **show cdp** command to verify the CDP global settings:

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

Global CDP information:
Sending CDP packets every 20 seconds
Sending a holdtime value of 30 seconds
Sending CDPv2 advertisements is not enabled
```




CHAPTER 7

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 75](#)
- [Information About Periodic MIB Data Collection and Transfer, on page 75](#)
- [How to Configure Periodic MIB Data Collection and Transfer, on page 77](#)
- [Periodic MIB Data Collection and Transfer: Example, on page 83](#)

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.

SUMMARY STEPS

1. **configure**
2. **snmp-server mib bulkstat object-list** *list-name*
3. **add** {oid | *object-name*}
4. Use the **commit** or **end** command.

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	snmp-server mib bulkstat object-list <i>list-name</i> 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 <i>object-name</i> } 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.

	Command or Action	Purpose
		end — Prompts user to take one of these actions: <ul style="list-style-type: none"> • 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.

SUMMARY STEPS

1. **configure**
2. **snmp-server mib bulkstat schema** *schema-name*
3. **object-list** *list-name*
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*
5. **poll-interval** *minutes*
6. Use the **commit** or **end** command.

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	snmp-server mib bulkstat schema <i>schema-name</i> 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.

	Command or Action	Purpose
Step 3	object-list <i>list-name</i> 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: <ul style="list-style-type: none"> • instance exact {interface <i>interface-id</i> [sub-if] oid <i>oid</i>} • instance wild {interface <i>interface-id</i> [sub-if] oid <i>oid</i>} • instance range start <i>oid</i> end <i>oid</i> • instance repetition <i>oid</i> max <i>repeat-number</i> 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: <ul style="list-style-type: none"> • 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 <i>minutes</i> 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: <ul style="list-style-type: none"> • 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.

SUMMARY STEPS

1. **configure**
2. **snmp-server mib bulkstat transfer-id** *transfer-id*
3. **buffer-size** *bytes*
4. **format** {**bulkBinary** | **bulkASCII** | **schemaASCII**}
5. **schema** *schema-name*
6. **transfer-interval** *minutes*
7. **url primary** *url*
8. **url secondary** *url*
9. **retry** *number*
10. **retain** *minutes*
11. **enable**
12. **commit** *minutes*

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	snmp-server mib bulkstat transfer-id <i>transfer-id</i> Example: RP/0/RP0/CPU0:router(config)# snmp-server mib bulkstat transfer bulkstat1	Identifies the transfer configuration with a name (<i>transfer-id</i> argument) and enters bulk statistics transfer configuration mode.
Step 3	buffer-size <i>bytes</i> 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.

	Command or Action	Purpose
		<p>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.</p>
Step 4	<p>format {bulkBinary bulkASCII schemaASCII}</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-bulk-tr)# format schemaASCII</pre>	<p>(Optional) Specifies the format of the bulk statistics data file (VFile). The default is schemaASCII.</p> <p>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.</p>
Step 5	<p>schema <i>schema-name</i></p> <p>Example:</p> <pre>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</pre>	<p>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).</p>
Step 6	<p>transfer-interval <i>minutes</i></p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-bulk-tr)# transfer-interval 20</pre>	<p>(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.</p>
Step 7	<p>url <i>primary url</i></p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-bulk-tr)# url primary ftp://user:password@host/folder/bulkstat1</pre>	<p>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.</p>
Step 8	<p>url <i>secondary url</i></p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-bulk-tr)# url secondary tftp://10.1.0.1/tftpboot/user/bulkstat1</pre>	<p>(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.</p>
Step 9	<p>retry <i>number</i></p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-bulk-tr)# retry 1</pre>	<p>(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.</p>

	Command or Action	Purpose
		<p>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.</p> <p>If all retries fail, the next normal transfer occurs after the configured transfer-interval time.</p>
Step 10	<p>retain <i>minutes</i></p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-bulk-tr)# retain 60</pre>	<p>(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.</p> <p>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.</p>
Step 11	<p>enable</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-bulk-tr)# enable</pre>	<p>Begins the bulk statistics data collection and transfer process for this configuration.</p> <ul style="list-style-type: none"> 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	<p>commit <i>minutes</i></p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-bulk-tr)# retain 60</pre>	<p>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.</p>

	Command or Action	Purpose
		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/dumppdcm
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 8

Configuring Flexible Command Line Interface

This module describes how to configure and use flexible command line interface (CLI) configuration groups.

- [Flexible CLI Configuration Groups, on page 85](#)
- [Flexible Configuration Restrictions, on page 85](#)
- [Configuring a Configuration Group, on page 87](#)
- [Verifying the Configuration of Configuration Groups, on page 89](#)
- [Regular Expressions in Configuration Groups, on page 91](#)
- [Configuration Examples for Flexible CLI Configuration, on page 102](#)

Flexible CLI Configuration Groups

Flexible command line interface (CLI) configuration groups provide the ability to minimize repetitive configurations by defining a series of configuration statements in a configuration group, and then applying this group to multiple hierarchical levels in the router configuration tree.

Flexible CLI configuration groups utilize regular expressions that are checked for a match at multiple submodes of the configuration tree based on where the group is applied within the hierarchy. If a match is found at a configuration submode, the corresponding configuration defined in the group is inherited within the matched submode.

Flexible CLI configuration groups also provide an auto-inheritance feature. Auto-inheritance means that any change done to a CLI configuration group is automatically applied to the configuration in any matched submodes that have an apply-group at that hierarchical level. This allows you to make a configuration change or addition once, and have it applied automatically in multiple locations, depending on where you have applied the flexible CLI configuration group.

Flexible Configuration Restrictions

Note these restrictions while using flexible configuration groups:

- Flexible CLI configuration groups are not supported in administration configurations and corresponding apply-groups are not supported in administration configurations.
- Use of preconfigured interfaces in configuration groups is not supported.
- Downgrading from an image that supports configuration groups to an image that does not support them is not supported.

- Access lists, quality of service and route policy configurations do not support the use of configuration groups. Configurations such as these are not valid:

```
group g-not-supported
  ipv4 access-list ...
  !
  ipv6 access-list ...
  !
  ethernet-service access-list ...
  !
  class-map ...
  !
  policy-map ...
  !
  route-policy ...
  !
end-group
```

You can, however, reference such configurations, as shown in this example:

```
group g-reference-ok
  router bgp 6500
  neighbor 7::7
  remote-as 65000
  bfd fast-detect
  update-source Loopback300
  graceful-restart disable
  address-family ipv6 unicast
    route-policy test1 in
    route-policy test2 out
  soft-reconfiguration inbound always
  !
  !
  !
  interface Bundle-Ether1005
    bandwidth 10000000
    mtu 9188
    service-policy output input_1
    load-interval 30
  !
end-group
```

- Some regular expressions are not supported within groups. For example, ‘?’, ‘|’ and ‘\$,’ are not supported within groups. Also some characters such as /d and /w are not supported.

- The choice operator “|” to express multiple match expressions within a regular expression is not supported. For example, these expressions are not supported:

Gig.*|Gig.*\..*—To match on either Gigabit Ethernet main interfaces or Gigabit Ethernet sub-interfaces.

Gig.*0/0/0/[1-5]|Gig.*0/0/0/[10-20]—To match on either Gig.*0/0/0/[1-5] or Gig.*0/0/0/[10-20].

'TenGigE.*|HundredGigE.*—To match on either TenGigE.* or HundredGigE.*.

- Commands that require a node identifier for the **location** keyword are not supported. For example, this configuration is not supported:

```
lpts pifib hardware police location 0/RP0/CPU0
```


- Overlapping regular expressions within a configuration group for the same configuration are not supported. For example:

```
group G-INTERFACE
interface 'gig.*a.*'
    mtu 1500
!
interface 'gig.*e.* '
    mtu 2000
!
end-group

interface gigabitethernet0/0/0/* ---- where * is 0 to 79 or 0 to 39
    apply-group G-INTERFACE
```

This configuration is not permitted because it cannot be determined whether the `interface GigabitEthernet0/0/0/*` configuration inherits `mtu 1500` or `mtu 2000`. Both expressions in the configuration group match `GigabitEthernet0/0/0/*`.

- Up to eight configuration groups are permitted on one `apply-group` command.

Configuring a Configuration Group

A configuration group includes a series of configuration statements that can be used in multiple hierarchical levels in the router configuration tree. By using regular expressions in a configuration group, you can create generic commands that can be applied in multiple instances.

Use this task to create and use a configuration group.



Note Flexible CLI configurations are not available through the XML interface.

SUMMARY STEPS

1. **configure**
2. **group** *group-name*
3. Enter configuration commands, starting from global configuration mode. Use regular expressions for interface names and other variable instances.
4. **end-group**
5. **apply-group**

DETAILED STEPS

Step 1 **configure**

Example:

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

Enters global configuration mode.

Step 2 `group group-name`**Example:**

```
RP/0/RP0/CPU0:router(config)# group g-interf
```

Specifies a name for a configuration group and enters group configuration mode to define the group. The *group-name* argument can have up to 32 characters and cannot contain any special characters.

Step 3 Enter configuration commands, starting from global configuration mode. Use regular expressions for interface names and other variable instances.**Example:**

```
RP/0/RP0/CPU0:router(config)# group g-interf
RP/0/RP0/CPU0:router(config-GRP)# interface 'GigabitEthernet.*'
RP/0/RP0/CPU0:router(config-GRP-if)# mtu 1500
```

Specifies the configuration statements that you want included in this configuration group.

For more information regarding the use of regular expressions, see [Configuration Group Inheritance with Regular Expressions: Example, on page 100](#). This example is applicable to all Gigabit Ethernet interfaces.

Step 4 `end-group`**Example:**

```
RP/0/RP0/CPU0:router(config-GRP-if)# end-group
```

Completes the configuration of a configuration group and exits to global configuration mode.

Step 5 `apply-group`**Example:**

```
RP/0/RP0/CPU0:router(config)# interface GigabitEthernet0/2/0/0
RP/0/RP0/CPU0:router(config-if)# apply-group g-interf
```

Adds the configuration of the configuration group into the router configuration applicable at the location that the group is applied. Groups can be applied in multiple locations, and their effect depends on the location and context.

The MTU value from the group `g-interf` is applied to the interface `GigabitEthernet0/2/0/0`. If this group is applied in global configuration mode, the MTU value is inherited by all Gigabit Ethernet interfaces that do not have an MTU value configured.

Simple Configuration Group: Example

This example shows how to use configuration groups to add a global configuration to the system:

```
RP/0/RP0/CPU0:router(config)# group g-logging
RP/0/RP0/CPU0:router(config-GRP)# logging trap notifications
RP/0/RP0/CPU0:router(config-GRP)# logging console debugging
RP/0/RP0/CPU0:router(config-GRP)# logging monitor debugging
RP/0/RP0/CPU0:router(config-GRP)# logging buffered 10000000
RP/0/RP0/CPU0:router(config-GRP)# end-group
```

```
RP/0/RP0/CPU0:router(config)# apply-group g-logging
```

When this configuration is committed, all commands contained in the g-logging configuration group are committed.

Configuration Group Applied to Different Places: Example

Configuration groups can be applied to different places, and their effect depends on the context within which they are applied. Consider this configuration group:

```
RP/0/RP0/CPU0:router(config)# group g-interfaces
RP/0/RP0/CPU0:router(config-GRP)# interface 'GigabitEthernet.*'
RP/0/RP0/CPU0:router(config-GRP-if)# mtu 1500
RP/0/RP0/CPU0:router(config-GRP-if)# exit
RP/0/RP0/CPU0:router(config-GRP)# interface 'GigabitEthernet.*'
RP/0/RP0/CPU0:router(config-GRP-if)# mtu 1000
RP/0/RP0/CPU0:router(config-GRP-if)# exit
RP/0/RP0/CPU0:router(config-GRP)# interface 'GigabitEthernet.*'
RP/0/RP0/CPU0:router(config-GRP-if)# mtu 2000
RP/0/RP0/CPU0:router(config-GRP-if)# end-group
```

This group can be applied to Gigabit Ethernet interface and in each instance the applicable MTU is applied. For instance, in this example, the Gigabit Ethernet interface is configured to have an MTU of 1000:

```
RP/0/RP0/CPU0:router(config)# interface GigabitEthernet0/2/0/0
RP/0/RP0/CPU0:router(config-if)# apply-group g-interfaces
RP/0/RP0/CPU0:router(config-if)# ipv4 address 2.2.2.2 255.255.255.0
```

In this example, the Gigabit Ethernet interface is configured to have an MTU of 1500:

```
RP/0/RP0/CPU0:router(config)# interface GigabitEthernet0/2/0/0
RP/0/RP0/CPU0:router(config-if)# apply-group g-interfaces
RP/0/RP0/CPU0:router(config-if)# ipv4 address 3.3.3.3 255.255.255.0
```

The same configuration group is used in both cases, but only the applicable configuration statements are used.

Verifying the Configuration of Configuration Groups

Use this task to verify the router configuration using configuration groups:

SUMMARY STEPS

1. **show running-config group** *[group-name]*
2. **show running-config**
3. **show running-config inheritance**

4. show running-config interface x/y/z inheritance *detail*

DETAILED STEPS

	Command or Action	Purpose
Step 1	show running-config group [<i>group-name</i>] Example: <pre>RP/0/RP0/CPU0:router# show running-config group group g-int-ge interface 'GigabitEthernet.*' mtu 1000 negotiation auto ! end-group</pre>	Displays the contents of a specific or all configured configuration groups.
Step 2	show running-config Example: <pre>RP/0/RP0/CPU0:router# show running-config group G-INTERFACE-MTU interface 'GigabitEthernet.*' mtu 1500 ! end-group interface interface GigabitEthernet0/4/1/0 apply-group G-INTERFACE-MTU ! interface interface GigabitEthernet0/4/1/1 apply-group G-INTERFACE-MTU mtu 2000 !</pre>	Displays the running configuration. Any applied groups are displayed. There is no indication as to whether these configuration groups affect the actual configuration or not. In this example, although the group G-INTERFACE-MTU is applied to interface GigabitEthernet0/4/1/1, the configured MTU value is 2000 and not 1500. This happens if the command mtu 2000 is configured directly on the interface. An actual configuration overrides a configuration group configuration if they are the same.
Step 3	show running-config inheritance Example: <pre>RP/0/RP0/CPU0:router# show running-config inheritance . . group G-INTERFACE-MTU interface 'GigabitEthernet.*' mtu 1500 ! end-group . . interface interface GigabitEthernet0/4/1/0 ## Inherited from group G-INTERFACE-MTU mtu 1500 ! interface interface GigabitEthernet0/4/1/1</pre>	Displays the inherited configuration where ever a configuration group has been applied.

	Command or Action	Purpose
	<pre>mtu 2000 !</pre>	
Step 4	<p>show running-config interface x/y/z inheritance detail</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router# show running-config interface interface GigabitEthernet0/4/1/0 inheritance detail interface interface GigabitEthernet0/4/1/0 ## Inherited from group G-INTERFACE-MTU mtu 1500</pre>	Displays the inherited configuration for a specific configuration command.

Regular Expressions in Configuration Groups

Regular expressions are used in configuration groups to make them widely applicable. Portable Operating System Interface for UNIX (POSIX) 1003.2 regular expressions are supported in the names of configuration statements. Single quotes must be used to delimit a regular expression.



Note Not all POSIX regular expressions are supported.

Regular Expressions for Interface Identifiers

Configuration groups do not accept exact interface identifiers. You must use a regular expression to identify a group of interfaces that are applicable to the configuration group. The regular expression `.*` is not allowed. You must begin the regular expression for an interface identifier with an unambiguous word, followed by the regular expression. For example, to configure Gigabit Ethernet interfaces, use the regular expression `'GigabitEthernet.*'`.

To display a list of available interface types for your router configuration, enter **interface ?** at the configuration group prompt:

```
RP/0/RP0/CPU0:router(config-GRP)# interface ?

ATM                'RegExp': ATM Network Interface(s)
BVI                'RegExp': Bridge-Group Virtual Interface
Bundle-Ether       'RegExp': Aggregated Ethernet interface(s)
GigabitEthernet    'RegExp': GigabitEthernet/IEEE 802.3 interface(s)
IMA                'RegExp': ATM Network Interface(s)
Loopback           'RegExp': Loopback interface(s)
MgmtEth            'RegExp': Ethernet/IEEE 802.3 interface(s)
Multilink          'RegExp': Multilink network interface(s)
Null               'RegExp': Null interface
PW-Ether           'RegExp': PWHE Ethernet Interface
PW-IW              'RegExp': PWHE VC11 IP Interworking Interface
Serial             'RegExp': Serial network interface(s)
tunnel-ip          'RegExp': GRE/IPinIP Tunnel Interface(s)
```

```
tunnel-mte      'RegExp': MPLS Traffic Engineering P2MP Tunnel interface(s)
tunnel-te       'RegExp': MPLS Traffic Engineering Tunnel interface(s)
tunnel-tp       'RegExp': MPLS Transport Protocol Tunnel interface
```



Note Although you are required to enter only enough characters for the interface type to be unique, it is recommended that you enter the entire phrase. All interface types used in regular expressions are case-sensitive.

To specify a subinterface, prefix the expression with the characters `\.` (backslash period). For example, use `interface 'GigabitEthernet.*\..*'` to configure all Gigabit Ethernet subinterfaces.

You can specify Layer 2 transport interfaces or point-to-point interfaces as shown in these examples:

```
group g-l2t
  interface 'Gi.*\..*' l2transport
  .
end-group
group g-ptp
  interface 'Gi.*\..*' point-to-point
  .
end-group
```

Regular Expressions for an OSPF Configuration

Exact router process names and OSPF areas cannot be used. You must use a regular expression to specify a process name or group of OSPF areas. To specify that the OSPF area can be either a scalar value or an IP address, use the regular expression `'.*'`, as in this example:

```
group g-ospf
  router ospf '.*'
  area '.*'
  mtu-ignore enable
  !
end-group
```

To specify that the OSPF area must be an IP address, use the expression `'\.'` as in this example:

```
group g-ospf-ipaddress
  router ospf '.*\..*\..*\..*.*'
  area '.*'
  passive enable
  !
end-group
```

To specify that the OSPF area must be a scalar value, use the expression `'1.*'`, as in this example:

```
group g-ospf-match-number
  router ospf '.*'
  area '1.*'
  passive enable
```

```
!
!
end-group
```

Regular Expressions for a BGP AS

Exact BGP AS values cannot be used in configuration groups. Use a regular expression to specify either AS plain format, or AS dot format as in the format X.Y. To match AS plain format instances, use a simple regular expression. To match AS dot format instances, use two regular expressions separated by a dot, as shown in this example:

```
group g-bgp
router bgp '*'.'*'
address-family ipv4 unicast
!
!
end-group
```

Regular Expressions for ANCP

Exact Access Node Control Protocol (ANCP) sender-name identifiers cannot be used in configuration groups. Because the sender name argument can be either an IP address or a MAC address, you must specify in the regular expression which one is being used. Specify an IP address as `'.*\..*\..*\..*'`; specify a MAC address as `'.*\..*\..*'`.

Resolving to a Uniform Type

Regular expressions must resolve to a uniform type. This is an example of an illegal regular expression:

```
group g-invalid
interface \.*'
  bundle port-priority 10
!
interface \.*Ethernet.*'
  bundle port-priority 10
!
end-group
```

In this example, the **bundle** command is supported for interface type GigabitEthernet but not for interface type 'FastEthernet'. The regular expressions `\.*'` and `\.*Ethernet.*'` match both GigabitEthernet and FastEthernet types. Because the **bundle** command is not applicable to both these interface types, they do not resolve to a uniform type and therefore the system does not allow this configuration.



Note If the system cannot determine from the regular expression what the configuration should be, the expression is not considered valid.



Note The regular expression `\.*'` is not allowed when referring to an interface identifier. You must begin the regular expression for an interface identifier with an unambiguous word, followed by the regular expression. Refer to *Regular Expressions for Interface Identifiers* in this section for more information.

Overlapping Regular Expressions

Regular expressions are used in names of configuration statements within a configuration group. This permits inheritance by the configuration when applied to matching names. Single quotes are used to delimit the regular expression. Overlapping regular expression within a configuration group for the same configuration is permitted.

The example, given below, illustrates the process of creating and applying multiple configuration groups:

```
RP/0//CPU0:router(config)#group FB_flexi_snmp
RP/0//CPU0:router(config-GRP)# snmp-server vrf '.*'
RP/0//CPU0:router(config-GRP-snmp-vrf)# host 1.1.1.1 traps version 2c group_1
RP/0//CPU0:router(config-GRP-snmp-vrf)# host 1.1.1.1 informs version 2c group_1
RP/0//CPU0:router(config-GRP-snmp-vrf)# context group_1

RP/0//CPU0:router(config-GRP-snmp-vrf)#
RP/0//CPU0:router(config-GRP-snmp-vrf)#commit

RP/0//CPU0:router(config-GRP-snmp-vrf)#root
RP/0//CPU0:router(config)#
RP/0//CPU0:router(config)#snmp-server vrf vrf1
RP/0//CPU0:router(config-snmp-vrf)#snmp-server vrf vrf10
RP/0//CPU0:router(config-snmp-vrf)#!
RP/0//CPU0:router(config-snmp-vrf)#snmp-server vrf vrf100
RP/0//CPU0:router(config-snmp-vrf)#
RP/0//CPU0:router(config-snmp-vrf)#commit

RP/0//CPU0:router(config-snmp-vrf)#root
RP/0//CPU0:router(config)#
RP/0//CPU0:router(config)#apply-group FB_flexi_snmp
RP/0//CPU0:router(config)#do sh running-config group
group FB_flexi_snmp
  snmp-server vrf '.*'
  host 1.1.1.1 traps version 2c group_1
  host 1.1.1.1 informs version 2c group_1
  context group_1
!
end-group
apply-group FB_flexi_snmp
snmp-server vrf vrf1
!
snmp-server vrf vrf10
!
snmp-server vrf vrf100
!
RP/0//CPU0:ios#show running-config inheritance detail

group FB_flexi_snmp
  snmp-server vrf '.*'
  host 1.1.1.1 traps version 2c group_1
  host 1.1.1.1 informs version 2c group_1
  context group_1
!
end-group
snmp-server vrf vrf1
## Inherited from group FB_flexi_snmp
host 1.1.1.1 traps version 2c group_1
## Inherited from group FB_flexi_snmp
host 1.1.1.1 informs version 2c group_1
## Inherited from group FB_flexi_snmp
context group_1
!
```



```

snmp-server vrf vrf10
## Inherited from group FB_flexi_snmp
host 1.1.1.1 traps version 2c group_1
## Inherited from group FB_flexi_snmp
host 1.1.1.1 informs version 2c group_1
## Inherited from group FB_flexi_snmp
context group_1
!
snmp-server vrf vrf100
## Inherited from group FB_flexi_snmp
host 1.1.1.1 traps version 2c group_1
## Inherited from group FB_flexi_snmp
host 1.1.1.1 informs version 2c group_1
## Inherited from group FB_flexi_snmp
context group_1

```

The example given below demonstrates the regular expression. In this example `snmp-server vrf '.*'` and `snmp-server vrf '[\w]+'` are two different regular expressions.

```

group FB_flexi_snmp
snmp-server vrf '.*'
host 1.1.1.1 traps version 2c group_1
host 1.1.1.1 informs version 2c group_1
context group_1
!
snmp-server vrf '[\w]+'
host 2.2.2.2 traps version 2c group_2
host 2.2.2.2 informs version 2c group_2
context group_2
!
end-group

```

This individual regular expression gets combined to all the three expressions - `snmp-server vrf vrf1`, `snmp-server vrf vrf10` and `snmp-server vrf vrf100` as given below.

```

apply-group FB_flexi_snmp
snmp-server vrf vrf1
!
snmp-server vrf vrf10
!
snmp-server vrf vrf100
!

```

In a configuration group, there can be instances of regular expressions overlap. In such cases, the regular expression with the highest priority is activated and inherited, when applied. It has that regular expression, which comes first in the lexicographic order that has the highest priority.

The following example shows how to use overlapping regular expressions and how the expression with higher priority is applied:

```

group FB_flexi_snmp
snmp-server vrf '.*'

```

```

host 1.1.1.1 traps version 2c group_1
host 1.1.1.1 informs version 2c group_1
context group_1
!
snmp-server vrf '[\w]+'

host 2.2.2.2 traps version 2c group_2
host 2.2.2.2 informs version 2c group_2
context group_2
!
end-group

```

The expression shown below has the highest priority:

```

group FB_flexi_snmp

snmp-server vrf '.*'

host 1.1.1.1 traps version 2c group_1
host 1.1.1.1 informs version 2c group_1
context group_1

```

The examples given above, show two different regular expression `snmp-server vrf '.*'` and `snmp-server vrf '[\w]+'`.

The expression below, shows how these two expressions get merged together:

```

apply-group FB_flexi_snmp

snmp-server vrf vrf1
!
snmp-server vrf vrf10
!
snmp-server vrf vrf100
!

```

Any change in a regular expression with lower priority will not affect the inheritance.

Any changes made to an existing regular expression, which is of less (non-top) priority, it will not have any effect on the inheritance.

```

snmp-server vrf '[\w]+'

host 2.2.2.2 traps version 2c group_2
host 2.2.2.2 informs version 2c group_2
context group_2

```

The expression with the higher priority gets inherited, as shown below:

```

group FB_flexi_snmp

snmp-server vrf '.*'

```

```

host 1.1.1.1 traps version 2c group_1

host 1.1.1.1 informs version 2c group_1

context group_1

```

Apply Groups Priority Inheritance

Priority governs inheritance.



Note From the Cisco IOS XR, Release 6.3.1 onwards, you are able to enter the Flexible CLI config group definition, **apply-group** and **exclude-group** command in any order as long as the entire commit has all the group definitions needed.

Apply groups priority inheritance helps flexible configuration groups to handle common configuration statements between groups. When multiple configuration groups have common configuration statements, the inheritance priority is such that the configuration statements present in inner groups have precedence over those configuration statements present in outer groups. In case of tiebreakers, the priority is assigned in accordance to the lexicographical order of regular expressions. User defined order of commands are not accepted.

For example, a configuration statement in configuration group ONE has precedence over another group. A configuration statement in configuration group SEVEN is used only if it does not exist in any other group. Within a configuration group, inheritance priority is the longest match.

```

apply-group SIX SEVEN
router ospf 0
apply-group FOUR FIVE
area 0
apply-group THREE
interface GigabitEthernet0/0/0/0
apply-group ONE TWO

!
!
!

```

The above example shows two scenarios. The inner most group (**apply-group ONE TWO**) has the highest priority. Case 1

The first scenario shows which group gets the priority. The example states which group is applied between different configuration groups (different groups with nothing in common). While applying group one (ONE TWO), all the seven groups matches the interface `interface GigabitEthernet0/0/0/0` is applied.

Case 2

Here, when all have the same (common) configuration, group one will be active. That is `apply-group ONE TWO` is active. If group ONE is deleted, then group TWO will be active.

Configuration Examples Using Regular Expressions

Configuration Group with Regular Expression: Example

This example shows the definition of a configuration group for configuring Gigabit Ethernet interfaces with ISIS routing parameters, using regular expressions for the exact interface:

```
RP/0/RP0/CPU0:router(config)# group g-isis-gige
RP/0/RP0/CPU0:router(config-GRP)# router isis '.*'
RP/0/RP0/CPU0:router(config-GRP-isis)# interface 'GigabitEthernet.*'
RP/0/RP0/CPU0:router(config-GRP-isis-if)# lsp-interval 20
RP/0/RP0/CPU0:router(config-GRP-isis-if)# hello-interval 40
RP/0/RP0/CPU0:router(config-GRP-isis-if)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-GRP-isis-if-af)# metric 10
RP/0/RP0/CPU0:router(config-GRP-isis-if-af)# end-group
RP/0/RP0/CPU0:router(config)#
```

To illustrate the use of this configuration group, assume that you want to configure these Gigabit Ethernet interfaces with the ISIS routing parameters:

```
router isis green
interface GigabitEthernet0/0/0/0
  lsp-interval 20
  hello-interval 40
  address-family ipv4 unicast
  metric 10
!
!
interface GigabitEthernet0/0/0/1
  lsp-interval 20
  hello-interval 40
  address-family ipv4 unicast
  metric 10
!
!
interface GigabitEthernet0/0/0/2
  lsp-interval 20
  hello-interval 40
  address-family ipv4 unicast
  metric 10
!
!
interface GigabitEthernet0/0/0/3
  lsp-interval 20
  hello-interval 40
  address-family ipv4 unicast
  metric 10
!
!
!
```

There are three possible ways to use the configuration group to configure these interfaces. The first is by applying the group within the interface configuration, as shown here:

```
router isis green
interface GigabitEthernet0/0/0/0
  apply-group g-isis-gige
```

```
!  
!  
interface GigabitEthernet0/0/0/1  
    apply-group g-isis-gige  
!  
!  
interface GigabitEthernet0/0/0/2  
    apply-group g-isis-gige  
!  
!  
interface GigabitEthernet0/0/0/3  
    apply-group g-isis-gige  
!  
!
```

In this situation, only the interfaces to which you apply the configuration group inherit the configuration.

The second way to configure these interfaces using the configuration group is to apply the configuration group within the **router isis** configuration, as shown here:

```
router isis green  
    apply-group g-isis-gige  
interface GigabitEthernet0/0/0/0  
!  
interface GigabitEthernet0/0/0/1  
!  
interface GigabitEthernet0/0/0/2  
!  
interface GigabitEthernet0/0/0/3  
!  
!
```

In this way, any other Gigabit Ethernet interfaces that you configure in the ISIS green configuration also inherit these configurations.

The third way to configure these interfaces using the configuration group is to apply the group at the global level as shown here:

```
    apply-group g-isis-gige  
router isis green  
interface GigabitEthernet0/0/0/0  
!  
interface GigabitEthernet0/0/0/1  
!  
interface GigabitEthernet0/0/0/2  
!  
interface GigabitEthernet0/0/0/3  
!  
!
```

In this example, the configuration of the group is applied to all Gigabit Ethernet interfaces configured for ISIS.

Configuration Group Inheritance with Regular Expressions: Example

Local Configuration Has Precedence Over Configuration Group

An explicit configuration takes precedence over a configuration applied from a configuration group. For example, assume that this configuration is running on the router:

```
router ospf 100
  packet-size 1000
!
```

You configure this configuration group, apply it, and commit it to the configuration.

```
RP/0/RP0/CPU0:router(config)# group g-ospf
RP/0/RP0/CPU0:router(config-GRP)# router ospf '.*'
RP/0/RP0/CPU0:router(config-GRP-ospf)# nsf cisco
RP/0/RP0/CPU0:router(config-GRP-ospf)# packet-size 3000
RP/0/RP0/CPU0:router(config-GRP-ospf)# end-group

RP/0/RP0/CPU0:router(config)# apply-group g-ospf
```

The result is effectively this configuration:

```
router ospf 100
  packet-size 1000
  nsf cisco
```

Note that `packet-size 3000` is not inherited from the configuration group because the explicit local configuration has precedence.

Compatible Configuration Is Inherited

The configuration in the configuration group must match the configuration on the router to be inherited. If the configuration does not match, it is not inherited. For example, assume that this configuration is running on the router:

```
router ospf 100
  auto-cost disable
!
```

You configure this configuration and commit it to the configuration.

```
RP/0/RP0/CPU0:router(config)# group g-ospf
RP/0/RP0/CPU0:router(config-GRP)# router ospf '.*'
RP/0/RP0/CPU0:router(config-GRP-ospf)# area '.*'
RP/0/RP0/CPU0:router(config-GRP-ospf-ar)# packet-size 2000
RP/0/RP0/CPU0:router(config-GRP-ospf)# end-group

RP/0/RP0/CPU0:router(config)# apply-group g-ospf

RP/0/RP0/CPU0:router(config)# router ospf 200
RP/0/RP0/CPU0:router(config-ospf)# area 1
```

The result is effectively this configuration:

```
router ospf 100
  auto-cost disable

router ospf 200
  area 1
  packet-size 2000
```

The packet size is inherited by the ospf 200 configuration, but not by the ospf 100 configuration because the area is not configured.

Layer 2 Transport Configuration Group: Example

This example shows how to configure and apply a configuration group with Layer 2 transport subinterfaces:

```
RP/0/RP0/CPU0:router(config)# group g-l2trans-if
RP/0/RP0/CPU0:router(config-GRP)# interface 'TenGigE.*\.*' l2transport
RP/0/RP0/CPU0:router(config-GRP)# mtu 1514
RP/0/RP0/CPU0:router(config-GRP)# end-group

RP/0/RP0/CPU0:router(config)# interface TenGigE0/0/0/0.1 l2transport
RP/0/RP0/CPU0:router(config-if)# apply-group g-l2trans-if
```

When this configuration is committed, the Ten Gigabit Ethernet interface 0/0/0/0.1 inherits the 1514 MTU value. This is the output displayed from the **show running-config inheritance** command for the Ten Gigabit Ethernet interface:

```
interface TenGigE0/0/0/0.1 l2transport
  ## Inherited from group g-l2trans-if
  mtu 1514
!
```

Configuration Group Precedence: Example

When similar configuration statements are contained in multiple configuration groups, groups applied in inner configuration modes take precedence over groups applied in outer modes. This example shows two configuration groups that configure different cost values for OSPF.

```
RP/0/RP0/CPU0:router(config)# group g-ospf2
RP/0/RP0/CPU0:router(config-GRP)# router ospf '.*'
RP/0/RP0/CPU0:router(config-GRP-ospf)# area '.*'
RP/0/RP0/CPU0:router(config-GRP-ospf-ar)# cost 2
RP/0/RP0/CPU0:router(config-GRP-ospf-ar)# end-group

RP/0/RP0/CPU0:router(config)# group g-ospf100
RP/0/RP0/CPU0:router(config-GRP)# router ospf '.*'
RP/0/RP0/CPU0:router(config-GRP-ospf)# area '.*'
RP/0/RP0/CPU0:router(config-GRP-ospf-ar)# cost 100
RP/0/RP0/CPU0:router(config-GRP-ospf-ar)# end-group
```

If these configuration groups are applied as follows, the cost 2 specified in g-ospf2 is inherited by OSPF area 0 because the group is applied in a more inner configuration mode. In this case, the configuration in group g-ospf100 is ignored.

```
RP/0/RP0/CPU0:router(config)# router ospf 0
RP/0/RP0/CPU0:router(config-ospf)# apply-group g-ospf100
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# apply-group g-ospf2
```

Changes to Configuration Group are Automatically Inherited: Example

When you make changes to a configuration group that is committed and applied to your router configuration, the changes are automatically inherited by the router configuration. For example, assume that this configuration is committed:

```
group g-interface-mtu
  interface 'GigabitEthernet.*'
    mtu 1500
  !
end-group

interface POS0/4/1/0
  apply-group g-interface-mtu
  !
```

Now you change the configuration group as in this example:

```
RP/0/RP0/CPU0:router(config)# group g-interface-mtu
RP/0/RP0/CPU0:router(config-GRP)# interface 'GigabitEthernet.*'
RP/0/RP0/CPU0:router(config-GRP-if)# mtu 2000
RP/0/RP0/CPU0:router(config-GRP-if)# end-group
```

When this configuration group is committed, the MTU configuration for interface GigabitEthernet0/4/1/0 is automatically updated to 2000.

Configuration Examples for Flexible CLI Configuration

Basic Flexible CLI Configuration: Example

This example shows that the Media Access Control (MAC) accounting configuration from the gd21 configuration group is applied to all Gigabit Ethernet interfaces in slot 2, ports 1 to 9.

1. Configure the configuration group that configures MAC accounting:

```
RP/0/RP0/CPU0:router# show running group gd21

group gd21
  interface 'GigabitEthernet0/0/0/2[1-9]'
  description general interface inheritance check
```



```
load-interval 30
mac-accounting ingress
mac-accounting egress
!
end-group
```

2. Check that the corresponding apply-group is configured in global configuration or somewhere in the hierarchy:

```
RP/0/RP0/CPU0:router# show running | in apply-group gd21

Building configuration...
apply-group gd21
```

3. Check the concise local view of the configuration of some of the interfaces:

```
RP/0/RP0/CPU0:router# show running interface

interface GigabitEthernet0/0/0/21
!
interface GigabitEthernet0/0/0/22
!
```

4. Verify that the match and inheritance occur on these interfaces:

```
RP/0/RP0/CPU0:router# show running-config inheritance interface

interface GigabitEthernet0/0/0/21
## Inherited from group gd21
description general interface inheritance check
## Inherited from group gd21
load-interval 30
## Inherited from group gd21
mac-accounting ingress
## Inherited from group gd21
mac-accounting egress
!
Interface GigabitEthernet0/0/0/22
## Inherited from group gd21
description general interface inheritance check
## Inherited from group gd21
load-interval 30
## Inherited from group gd21
mac-accounting ingress
## Inherited from group gd21
mac-accounting egress
!
!
```

5. Verify that the inherited configuration actually takes effect:

```
RP/0/RP0/CPU0:router# show mac-accounting GigabitEthernet0/0/0/21

GigabitEthernet0/0/0/21
  Input (96 free)
    6c9c.ed35.90fd: 1271 packets, 98426 bytes
    Total: 1271 packets, 98426 bytes
  Output (96 free)
    6c9c.ed35.90fd: 774 packets, 63265 bytes
```

Total: 774 packets, 63264 bytes

Interface MTU Settings for Different Interface Types: Example

This example shows that an MTU value is configured on different interface types.

1. Configure an interface MTU configuration group and apply this group:

```
RP/0/RP0/CPU0:router# show running group l2tr

group l2tr
interface 'GigabitEthernet0/0/0/3.*'
mtu 1500
!
interface 'GigabitEthernet0/0/0/9\..*'
mtu 1400
!
interface 'GigabitEthernet0/0/0/9\..*' l2transport
mtu 1400
!
end-group

RP/0/RP0/CPU0:router# show running | inc apply-group

Building configuration...

apply-group l2tr
```

2. Check the concise view and the inheritance view of the various interfaces:

```
RP/0/RP0/CPU0:router# show running interface gigabitEthernet0/0/0/30

interface GigabitEthernet0/0/0/30
!
RP/0/RP0/CPU0:router# show running interface gigabitEthernet0/0/0/30 inheritance detail

interface GigabitEthernet0/0/0/30
## Inherited from group l2tr
mtu 1500
!

RP/0/RP0/CPU0:router# show running interface gigabitEthernet0/0/0/9.800

interface GigabitEthernet0/0/0/9.800
encapsulation dot1q 800
!

RP/0/RP0/CPU0:router# show running interface gigabitEthernet0/0/0/9.800 inheritance detail

interface GigabitEthernet0/0/0/9.800
## Inherited from group l2tr
mtu 1400
encapsulation dot1q800
!

RP/0/RP0/CPU0:router# show running interface gigabitEthernet0/0/0/9.250
```

```
interface GigabitEthernet0/0/0/9.250 l2transport
 encapsulation dot1q 250
!
```

```
RP/0/RP0/CPU0:router# show running interface gigabitEthernet0/0/0/9.800 inheritance
detail
```

```
interface GigabitEthernet0/0/0/9.250 l2transport
 encapsulation dot1q250
## Inherited from group l2tr
mtu 1400
!
```

3. Verify that the correct values from the group do take effect:

```
RP/0/RP0/CPU0:router# show interface gigabitEthernet 0/0/0/30
```

```
GigabitEthernet0/0/0/30 is down, line protocol is down
Interface state transitions: 0
Hardware is GigabitEthernet, address is 0026.9824.ee56 (bia 0026.9824.ee56)
Internet address is Unknown
MTU 1500 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)
 reliability 255/255, txload 0/255, rxload 0/255
Encapsulation ARPA,
Full-duplex, 1000Mb/s, link type is force-up
output flow control is off, input flow control is off
loopback not set,
Last input never, output never
Last clearing of "show interface" counters never
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
 0 packets input, 0 bytes, 0 total input drops
 0 drops for unrecognized upper-level protocol
Received 0 broadcast packets, 0 multicast packets
 0 runs, 0 giants, 0 throttles, 0 parity
 0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
 0 packets output, 0 bytes, 0 total output drops
Output 0 broadcast packets, 0 multicast packets
 0 output errors, 0 underruns, 0 applique, 0 resets
 0 output buffer failures, 0 output buffers swapped out
```

```
RP/0/RP0/CPU0:router# show interface gigabitEthernet 0/0/0/9.801
```

```
GigabitEthernet0/0/0/9.801 is up, line protocol is up
Interface state transitions: 1
Hardware is VLAN sub-interface(s), address is 0026.9824.ee41
Internet address is Unknown
MTU 1400 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)
 reliability 255/255, txload 0/255, rxload 0/255
Encapsulation 802.1Q Virtual LAN, VLAN Id 801, loopback not set,
Last input never, output never
Last clearing of "show interface" counters never
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
 0 packets input, 0 bytes, 0 total input drops
 0 drops for unrecognized upper-level protocol
Received 0 broadcast packets, 0 multicast packets
 0 packets output, 0 bytes, 0 total output drops
Output 0 broadcast packets, 0 multicast packets
```

```
RP/0/RP0/CPU0:router# show interface gigabitEthernet 0/0/0/9.250

GigabitEthernet0/0/0/9.250 is up, line protocol is up
Interface state transitions: 1
Hardware is VLAN sub-interface(s), address is 0026.9824.ee41
Layer 2 Transport Mode
MTU 1400 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)
  reliability Unknown, txload Unknown, rxload Unknown
Encapsulation 802.1Q Virtual LAN,
  Outer Match: Dot1Q VLAN 250
  Ethertype Any, MAC Match src any, dest any
loopback not set,
Last input never, output never
Last clearing of "show interface" counters never
  0 packets input, 0 bytes
  0 input drops, 0 queue drops, 0 input errors
  0 packets output, 0 bytes

  0 output drops, 0 queue drops, 0 output errors
```

ACL Referencing: Example

This example shows how to reference access-lists on a number of interfaces using configuration groups.

1. Configure the configuration group and apply-group:

```
RP/0/RP0/CPU0:router# show running group acref

group acref
interface 'GigabitEthernet0/0/0/3.*'
  ipv4 access-group adem ingress
  ipv4 access-group adem egress
!
end-group

RP/0/RP0/CPU0:router# show running | inc apply-group

Building configuration...

apply-group isis l2tr isis2 mpp bundle1 acref
```

2. Check the concise and inheritance view of the matching configurations:

```
RP/0/RP0/CPU0:router# show running interface gigabitEthernet 0/0/0/30

interface GigabitEthernet0/0/0/30
!

RP/0/RP0/CPU0:router# show running interface GigabitEthernet 0/0/0/30 inheritance detail

interface GigabitEthernet0/0/0/30
## Inherited from group l2tr
mtu 1500
## Inherited from group acref
ipv4 access-group adem ingress
## Inherited from group acref
```

```

    ipv4 access-group adem egress
    !

RP/0/RP0/CPU0:router# show running interface gigabitEthernet 0/0/0/31

interface GigabitEthernet0/0/0/31
    !

RP/0/RP0/CPU0:router# show running interface GigabitEthernet 0/0/0/31 inheritance detail

interface GigabitEthernet0/0/0/31
    ## Inherited from group l2tr
    mtu 1500
    ## Inherited from group acref
    ipv4 access-group adem ingress
    ## Inherited from group acref
    ipv4 access-group adem egress

```

3. Check that the ACL group configuration actually got configured by using a traffic generator and watching that denied traffic is dropped.

Local Configuration Takes Precedence: Example

This example illustrates that local configurations take precedence when there is a discrepancy between a local configuration and the configuration inherited from a configuration group.

1. Configure a local configuration in a configuration submode with an access list:

```

RP/0/RP0/CPU0:router# show running interface gigabitEthernet 0/0/0/39

interface GigabitEthernet0/0/0/39
    ipv4 access-group smany ingress
    ipv4 access-group smany egress
    !

RP/0/RP0/CPU0:router# show running interface gigabitEthernet 0/0/0/38

interface GigabitEthernet0/0/0/38
    !

RP/0/RP0/CPU0:router# show running ipv4 access-list smany

ipv4 access-list smany
    10 permit ipv4 any any
    !

RP/0/RP0/CPU0:router# show running ipv4 access-list adem

ipv4 access-list adem
    10 permit ipv4 21.0.0.0 0.255.255.255 host 55.55.55.55
    20 deny ipv4 any any
    !

```

2. Configure and apply the access list group configuration:

```

RP/0/RP0/CPU0:router# show running group acref

group acref

```

```

interface 'GigabitEthernet0/0/0/3.*'
  ipv4 access-group adem ingress
  ipv4 access-group adem egress
!
end-group

RP/0/RP0/CPU0:router# show running | inc apply-group

Building configuration...
apply-group isis l2tr isis2 mpp bundle1 acref

```

3. Check the concise and inheritance views for the matching interface where the access list reference is configured locally:

```

RP/0/RP0/CPU0:router# show running interface gigabitEthernet 0/0/0/39

interface GigabitEthernet0/0/0/39
  ipv4 access-group smany ingress
  ipv4 access-group smany egress
!

RP/0/RP0/CPU0:router# show running interface gigabitEthernet 0/0/0/39 inheritance detail

interface GigabitEthernet0/0/0/39
  ## Inherited from group l2tr
  mtu 1500
  ipv4 access-group smany ingress
  ipv4 access-group smany egress      << no config inherited, local config prioritized
!

RP/0/RP0/CPU0:router# show running interface gigabitEthernet 0/0/0/38

interface GigabitEthernet0/0/0/38
!

RP/0/RP0/CPU0:router# show running interface gigabitEthernet 0/0/0/38 inheritance detail

interface GigabitEthernet0/0/0/38
  ## Inherited from group l2tr
  mtu 1500
  ## Inherited from group acref
  ipv4 access-group adem ingress
  ## Inherited from group acref
  ipv4 access-group adem egress
!

```

4. Use a traffic generator to verify that the traffic pattern for interface GigabitEthernet0/0/0/39 gets acted on by the access list in the local configuration (smany) and not according to the inherited referenced access list (adem).

ISIS Hierarchical Configuration: Example

This example illustrates inheritance and priority handling with two ISIS groups using an ISIS configuration.

1. Configure the local ISIS configuration:

```
RP/0/RP0/CPU0:router# show running router isis
```

```
router isis vink
net 49.0011.2222.2222.2222.00
address-family ipv4 unicast
  mpls traffic-eng level-1-2
  mpls traffic-eng router-id Loopback0
redistribute connected
!
interface Bundle-Ether1
  address-family ipv4 unicast
!
!
interface Bundle-Ether2
!
interface Loopback0
!
interface TenGigE0/2/0/0.3521
  address-family ipv4 unicast
!
!
interface TenGigE0/2/0/0.3522
  address-family ipv4 unicast
!
!
interface TenGigE0/2/0/0.3523
  address-family ipv4 unicast
!
!
interface TenGigE0/2/0/0.3524
  address-family ipv4 unicast
!
!
interface TenGigE0/2/0/0.3525
  address-family ipv4 unicast
!
!
interface TenGigE0/2/0/0.3526
!
interface TenGigE0/2/0/0.3527
!
interface TenGigE0/2/0/0.3528
!
interface TenGigE0/2/0/1
  address-family ipv4 unicast
!
!
!
```

2. Configure two ISIS groups and apply these to the configuration:

```
RP/0/RP0/CPU0:router# show running group isis
```

```
group isis
router isis '.*'
  address-family ipv4 unicast
  mpls traffic-eng level-1-2
  mpls traffic-eng router-id Loopback0
  redistribute connected
  redistribute ospf 1 level-1-2
!
```

```

interface 'TenGig.*'
  lsp-interval 40
  hello-interval 15
  address-family ipv4 unicast
  metric 50
!
!
interface 'Bundle-Ether.*'
  address-family ipv4 unicast
  metric 55
!
!
!
end-group

RP/0/RP0/CPU0:router# show running group isis2

group isis2
router isis '.*'
!
router isis '^(\vink)'
  address-family ipv4 unicast
!
  interface '^(\Ten)Gig.*'
!
  interface '^(\Ten)Gig.*'
    address-family ipv4 unicast
    metric 66
!
!
!
end-group

RP/0/RP0/CPU0:router# show running | inc apply-group

Building configuration...

apply-group isis l2tr isis2 mpp bundle1 ahref

```

3. Check the inheritance view of the ISIS configuration:

```

RP/0/RP0/CPU0:router# show running router isis inheritance detail

router isis vink
net 49.0011.2222.2222.00
address-family ipv4 unicast
  mpls traffic-eng level-1-2
  mpls traffic-eng router-id Loopback0
  redistribute connected
  ## Inherited from group isis
  redistribute ospf 1 level-1-2
!
interface Bundle-Ether1
  address-family ipv4 unicast
  ## Inherited from group isis
  metric 55
!
!
interface Bundle-Ether2
  ## Inherited from group isis
  address-family ipv4 unicast
  ## Inherited from group isis

```



```
metric 55
!
!
interface Loopback0
!
interface TenGigE0/2/0/0.3521
## Inherited from group isis
lsp-interval 40
## Inherited from group isis
hello-interval 15
address-family ipv4 unicast
## Inherited from group isis
metric 50
!
!
interface TenGigE0/2/0/0.3522
## Inherited from group isis
lsp-interval 40
## Inherited from group isis
hello-interval 15
address-family ipv4 unicast
## Inherited from group isis
metric 50
!
!
interface TenGigE0/2/0/0.3523
## Inherited from group isis
lsp-interval 40
## Inherited from group isis
hello-interval 15
address-family ipv4 unicast
## Inherited from group isis
metric 50
!
!
interface TenGigE0/2/0/0.3524
## Inherited from group isis
lsp-interval 40
## Inherited from group isis
hello-interval 15
address-family ipv4 unicast
## Inherited from group isis
metric 50
!
!
interface TenGigE0/2/0/0.3525
## Inherited from group isis
lsp-interval 40
## Inherited from group isis
hello-interval 15
address-family ipv4 unicast
## Inherited from group isis
metric 50
!
!
interface TenGigE0/2/0/0.3526
## Inherited from group isis
lsp-interval 40
## Inherited from group isis
hello-interval 15
## Inherited from group isis
address-family ipv4 unicast
## Inherited from group isis
metric 50
```

```

!
!
interface TenGigE0/2/0/0.3527
## Inherited from group isis
lsp-interval 40
## Inherited from group isis
hello-interval 15
## Inherited from group isis
address-family ipv4 unicast
## Inherited from group isis
metric 50
!
!
interface TenGigE0/2/0/0.3528
## Inherited from group isis
lsp-interval 40
## Inherited from group isis
hello-interval 15
## Inherited from group isis
address-family ipv4 unicast
## Inherited from group isis
metric 50
!
!
interface TenGigE0/2/0/1
## Inherited from group isis
lsp-interval 40
## Inherited from group isis
hello-interval 15
address-family ipv4 unicast
## Inherited from group isis
metric 50
!
!
!

```

4. Verify the actual functionality:

```

RP/0/RP0/CPU0:router# show isis interface TenGigE0/2/0/0.3528 | inc Metric

Metric (L1/L2):          50/50

```

OSPF Hierarchy: Example

This example illustrates hierarchical inheritance and priority. The configuration that is lower in hierarchy gets the highest priority.

1. Configure a local OSPF configuration:

```

RP/0/RP0/CPU0:router# show running router ospf

router ospf 1
  apply-group go-c
  nsr
  router-id 121.121.121.121
  nsf cisco
  redistribute connected
  address-family ipv4 unicast

```

```

area 0
  apply-group go-b
  interface GigabitEthernet0/0/0/0
    apply-group go-a
  !
  interface GigabitEthernet0/0/0/1
  !
  interface GigabitEthernet0/0/0/3
  !
  interface GigabitEthernet0/0/0/4
  !
  interface GigabitEthernet0/0/0/21
    bfd minimum-interval 100
    bfd fast-detect
    bfd multiplier 3
  !
  interface TenGigE0/2/0/0.3891
  !
  interface TenGigE0/2/0/0.3892
  !
  interface TenGigE0/2/0/0.3893
  !
  interface TenGigE0/2/0/0.3894
  !
!
!
router ospf 100
!
router ospf 1000
!
router ospf 1001
!

```

2. Configure a configuration group and apply it in a configuration submode:

```
RP/0/RP0/CPU0:router# show running group go-a
```

```

group go-a
  router ospf '*'
  area '*'
    interface 'Gig.*'
      cost 200
    !
  !
!
end-group

```

```
RP/0/RP0/CPU0:router# show running group go-b
```

```

group go-b
  router ospf '*'
  area '*'
    interface 'Gig.*'
      cost 250
    !
  !
!
end-group

```

```
RP/0/RP0/CPU0:router# show running group go-c
```

```

group go-c
  router ospf '*'

```

```

area '*'
  interface 'Gig.*'
    cost 300
  !
!
!

```

```
end-group
```

3. Check the inheritance view and verify that the apply-group in the lowest configuration submode gets the highest priority:

```
RP/0/RP0/CPU0:router# show running router ospf 1 inheritance detail
```

```

router ospf 1
  nsr
  router-id 121.121.121.121
  nsf cisco
  redistribute connected
  address-family ipv4 unicast
  area 0
    interface GigabitEthernet0/0/0/0
      ## Inherited from group go-a
      cost 200                                << apply-group in lowest submode gets highest priority
    !
    interface GigabitEthernet0/0/0/1
      ## Inherited from group go-b
      cost 250
    !
    interface GigabitEthernet0/0/0/3
      ## Inherited from group go-b
      cost 250
    !
    interface GigabitEthernet0/0/0/4
      ## Inherited from group go-b
      cost 250
    !
    interface GigabitEthernet0/0/0/21
      bfd minimum-interval 100
      bfd fast-detect
      bfd multiplier 3
      ## Inherited from group go-b
      cost 250
    !
    interface TenGigE0/2/0/0.3891
    !
    interface TenGigE0/2/0/0.3892
    !
    interface TenGigE0/2/0/0.3893
    !
    interface TenGigE0/2/0/0.3894
    !
  !
!

```

4. Check the functionality of the cost inheritance through the groups:

```
RP/0/RP0/CPU0:router# show ospf 1 interface GigabitEthernet 0/0/0/0
```

```

GigabitEthernet0/0/0/0 is up, line protocol is up
  Internet Address 1.0.1.1/30, Area 0
  Process ID 1, Router ID 121.121.121.121, Network Type BROADCAST, Cost: 200

```

```

Transmit Delay is 1 sec, State DR, Priority 1, MTU 1500, MaxPktSz 1500
Designated Router (ID) 121.121.121.121, Interface address 1.0.1.1
No backup designated router on this network
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
Non-Stop Forwarding (NSF) enabled
  Hello due in 00:00:02
Index 5/5, flood queue length 0
Next 0(0)/0(0)
Last flood scan length is 1, maximum is 40
Last flood scan time is 0 msec, maximum is 7 msec
LS Ack List: current length 0, high water mark 0
Neighbor Count is 1, Adjacent neighbor count is 0
Suppress hello for 0 neighbor(s)
Multi-area interface Count is 0

```

Link Bundling Usage: Example

This example shows how to configure interface membership in a bundle link:

1. Configure the configuration groups:

```

RP/0/RP0/CPU0:router# show running group bundle1

group bundle1
  interface 'GigabitEthernet0/1/0/1[1-6]'
    bundle id 1 mode active
  !
end-group

RP/0/RP0/CPU0:router# show running | inc apply-group

Building configuration...

apply-group isis l2tr isis2 mpp bundle1

```

2. Check the local configuration:

```

RP/0/RP0/CPU0:router# show running interface gigabitEthernet 0/1/0/11

interface GigabitEthernet0/1/0/11
!

RP/0/RP0/CPU0:router# show running interface Bundle-Ether1

interface Bundle-Ether1
  ipv4 address 108.108.1.1 255.255.255.0
  bundle maximum-active links 10
  bundle minimum-active links 5
!

```

3. Check the inheritance configuration view:

```

RP/0/RP0/CPU0:router# show running interface GigabitEthernet 0/1/0/11 inheritance detail

interface GigabitEthernet0/1/0/11

```

```
## Inherited from group bundle1
bundle id 1 mode active
!
```

4. Check that the inheritance configuration took effect:

```
RP/0/RP0/CPU0:router# show interface Bundle-Ether1

Bundle-Ether1 is up, line protocol is up
Interface state transitions: 1
Hardware is Aggregated Ethernet interface(s), address is 0024.f71f.4bc3
Internet address is 108.108.1.1/24
MTU 1514 bytes, BW 6000000 Kbit (Max: 6000000 Kbit)
  reliability 255/255, txload 0/255, rxload 0/255
Encapsulation ARPA,
Full-duplex, 6000Mb/s
loopback not set,
ARP type ARPA, ARP timeout 04:00:00
  No. of members in this bundle: 6
    GigabitEthernet0/1/0/11      Full-duplex  1000Mb/s      Active
    GigabitEthernet0/1/0/12      Full-duplex  1000Mb/s      Active
    GigabitEthernet0/1/0/13      Full-duplex  1000Mb/s      Active
    GigabitEthernet0/1/0/14      Full-duplex  1000Mb/s      Active
    GigabitEthernet0/1/0/15      Full-duplex  1000Mb/s      Active
    GigabitEthernet0/1/0/16      Full-duplex  1000Mb/s      Active
Last input 00:00:00, output 00:00:00
Last clearing of "show interface" counters never
5 minute input rate 8000 bits/sec, 1 packets/sec
5 minute output rate 3000 bits/sec, 1 packets/sec
  2058 packets input, 1999803 bytes, 426 total input drops
    0 drops for unrecognized upper-level protocol
  Received 1 broadcast packets, 2057 multicast packets
    0 runts, 0 giants, 0 throttles, 0 parity
  0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
  1204 packets output, 717972 bytes, 0 total output drops
  Output 2 broadcast packets, 1202 multicast packets
  0 output errors, 0 underruns, 0 applique, 0 resets
  0 output buffer failures, 0 output buffers swapped out
  0 carrier transitions
```



CHAPTER 9

Managing Router Hardware

This chapter describes about clearing the memory and partitions of an RP or a line card before an RMA (Return Merchandise Authorization).

- [Clear the Memory and the Partitions of a Card, on page 117](#)
- [System Logs during RSP Switchover, on page 120](#)

Clear the Memory and the Partitions of a Card

Users can clear the memory and the partitions of an RP or a line card before an RMA (Return Merchandise Authorization). Clearing the memory and partitions of the card is performed when the card is defective and has to be returned.

When a line card or an RP is identified for an RMA, the user might want to remove the card from the chassis. However, the service personnel may not be available onsite to remove the card immediately. By clearing the memory and partitions of the card, the users can clear the RP or the line card and power-off the card and also let it remain in the slot.

After clearing the memory, do not reload the card or the chassis until the card is removed from the slot. This is because reloading will reboot the card or the chassis resulting in restoring the data that was erased.

In a dual RP system, the reset of the standby RP must be executed from the active RP. Once the standby RP has been cleaned, the standby RP will be shut down to prevent resync with the active RP.

Prerequisites

XR VM and the System Admin VM must be operational.



Note Do not perform an admin process restart, card reload, or an FPD upgrade while clearing the memory and partitions of the card.

Commands

Run the following commands from the XR VM to clear the memory and the partitions of the card:

- **show zapdisk locations**- displays the locations where the memory and the partition can be cleared.
- **zapdisk start location <location-id>** - clears the memory and the partition from the specified location.

The following steps explain how to clear the memory or the partition of the card:

1. Display the Locations to Clear the Memory - Use the **show zapdisk locations** command to display the locations to be cleared.

The following example shows how to display the location:

<! Display the Locations to Clear the Memory !>

```
Router# show zapdisk locations
0/RP1      Fully qualified location specification
0/2        Fully qualified location specification
0/6        Fully qualified location specification
all        all locations
```

```
Router#conf t
Router(config)#logging console disable
Router(config)#commit
Router(config)#end
```

2. Clear the Memory or Partition - Use the **zapdisk start location** command to clear the memory or partition.

The following example shows how to clear the memory or partition:

<! Clear the Memory or Partition !>

```
Router#zapdisk start location 0/2
Action on designated location is in progress, please monitor admin syslog.
Action on designated location is in progress, please monitor admin syslog.
```

```
Router#zapdisk start location 0/6
Action on designated location is in progress, please monitor admin syslog.
Action on designated location is in progress, please monitor admin syslog.
```

```
Router#zapdisk start location 0/RP1
Action on designated location is in progress, please monitor admin syslog.
Action on designated location is in progress, please monitor admin syslog.
```

3. Verify that the memory and the partition is cleared - Use **show logging**, **show platform**, **show controller card**, and **show reboot-history card location** commands to verify if the memory and partitions are cleared.

The following example shows how to verify if the memory and partitions are successfully cleared:

<!Verification!>

```
sysadmin-vm:0_RP0# show controller card-mgr event-history brief location 0/2
```

Card Event History for: 0/2

Card Event History as seen by Master (0/RP0)

Current State: **ZAPDISK_POWERED_ON**

DATE	TIME (UTC)	STATE	EVENT
03/04	22:26:13.400	ZAPDISK_RESET	ev_dml_power_up_ok
03/04	22:26:02.630	SYSADMIN_VM_GOING_DOWN	ev_zapdisk_req
03/04	22:25:46.660	CARD_READY	ev_sysadmin_vm_shutdown
03/04	21:58:14.842	OIR_INSERT_NOTIF	if_card_local_init_done
03/04	21:58:14.841	WAIT_CARD_INFO	ev_card_info_synced
03/04	21:57:57.219	WAIT_SYSADMIN_VM_READY	ev_sysadmin_vm_booted
03/04	21:57:45.305	HOST_OS_RUNNING	ev_sysadmin_vm_started
03/04	21:57:24.371	BOOTLDR_STARTED	ev_host_os_started


```

03/04 21:56:04.619 CARD_POWERED_ON      ev_bootldr_started
03/04 21:55:58.212 CARD_IN_RESET      ev_dml_power_up_ok
03/04 21:55:45.397 IMAGE_INSTALLED     ev_ios_install_reset
03/04 21:55:44.896 INSTALLING_IMAGE    ev_ios_install_done
03/04 21:54:53.045 WAIT_FIRST_EVENT   ev_ios_install_started
03/04 21:54:53.043 IDLE                      ev_present

sysadmin-vm:0_RP0# show controller card-mgr event-history brief location 0/6
Card Event History for: 0/6

Card Event History as seen by Master (0/RP0)
Current State: ZAPDISK_POWERED_ON

DATE      TIME (UTC)      STATE                      EVENT
-----
03/04 22:26:14.309 ZAPDISK_RESET            ev_dml_power_up_ok
03/04 22:26:03.722 SYSADMIN_VM_GOING_DOWN  ev_zapdisk_req
03/04 22:25:49.563 CARD_READY              ev_sysadmin_vm_shutdown
03/04 22:00:32.071 OIR_INSERT_NOTIF       if_card_local_init_done
03/04 22:00:32.070 WAIT_CARD_INFO         ev_card_info_synced
03/04 22:00:10.314 WAIT_SYSADMIN_VM_READY  ev_sysadmin_vm_booted
03/04 21:59:57.999 HOST_OS_RUNNING        ev_sysadmin_vm_started
03/04 21:59:35.271 BOOTLDR_STARTED        ev_host_os_started
03/04 21:58:18.244 CARD_POWERED_ON        ev_bootldr_started
03/04 21:58:11.836 CARD_IN_RESET          ev_dml_power_up_ok
03/04 21:57:59.122 IMAGE_INSTALLED        ev_ios_install_reset
03/04 21:57:58.521 INSTALLING_IMAGE       ev_ios_install_done
03/04 21:54:53.045 WAIT_FIRST_EVENT       ev_ios_install_started
03/04 21:54:53.043 IDLE                      ev_present

Aborted: by user
sysadmin-vm:0_RP0# show controller card-mgr event-history brief location 0/RP1
Card Event History for: 0/RP1

Card Event History as seen by Master (0/RP0)
Current State: ZAPDISK_POWERED_ON

DATE      TIME (UTC)      STATE                      EVENT
-----
03/04 22:26:24.730 ZAPDISK_RESET            ev_dml_power_up_ok
03/04 22:26:04.503 HOST_GOING_DOWN         ev_zapdisk_req
03/04 22:26:00.677 SYSADMIN_VM_GOING_DOWN  ev_host_shutdown_started
03/04 22:25:54.770 CARD_READY              ev_sysadmin_vm_shutdown
03/04 21:57:28.878 OIR_INSERT_NOTIF       if_card_local_init_done
03/04 21:57:28.878 WAIT_CARD_INFO         ev_card_info_synced
03/04 21:57:11.443 WAIT_SYSADMIN_VM_READY  ev_sysadmin_vm_booted
03/04 21:56:59.228 HOST_OS_RUNNING        ev_sysadmin_vm_started
03/04 21:56:31.882 BOOTING_IOS_IMAGE       ev_host_os_started
03/04 21:56:26.466 BOOTING_IOS_IMAGE       ev_boot_kernel
03/04 21:56:12.834 CARD_POWERED_ON        ev_bootldr_ssd_boot
03/04 21:56:09.730 CARD_IN_RESET          ev_dml_power_up_ok
03/04 21:55:48.701 IMAGE_INSTALLED        ev_ios_install_reset
03/04 21:55:47.700 INSTALLING_IMAGE       ev_ios_install_done
03/04 21:54:53.046 WAIT_FIRST_EVENT       ev_ios_install_started

Aborted: by user
sysadmin-vm:0_RP0# show logging | i card_mgr
0/RP0/ADMIN0:Mar 4 22:26:03.240 : card_mgr[3211]: %DRIVER-CARD_MGR-5-ZAPDISK_STARTED :
Card cleanup started for location 0/2
0/RP0/ADMIN0:Mar 4 22:26:04.332 : card_mgr[3211]: %DRIVER-CARD_MGR-5-ZAPDISK_STARTED :
Card cleanup started for location 0/6
0/RP0/ADMIN0:Mar 4 22:26:04.503 : card_mgr[3211]: %DRIVER-CARD_MGR-5-ZAPDISK_STARTED :
Card cleanup started for location 0/RP1
sysadmin-vm:0_RP0# show reboot-history card location 0/2
Card Reboot History for 0/2

```

```

0
Reason Code 22
Reason      "ZAPDISK by user request"
Src Location 0/RP0
Src Name     card_mgr
sysadmin-vm:0_RP0# show reboot-history card location 0/6

Card Reboot History for 0/6
0
Reason Code 22
Reason      "ZAPDISK by user request"
Src Location 0/RP0
Src Name     card_mgr
sysadmin-vm:0_RP0# show reboot-history card location 0/RP
Card Reboot History for 0/RP1
0
Reason Code 22
Reason      "ZAPDISK by user request"
Src Location 0/RP0
Src Name     card_mgr

sysadmin-vm:0_RP0# show reboot-history card location 0/RP1
Card Reboot History for 0/RP1
0
Reason Code 22
Reason      "ZAPDISK by user request"
Src Location 0/RP0
Src Name     card_mgr

```

4. Power-Down the Card - Shut down the card.

System Logs during RSP Switchover

In the event of an RSP switchover, the router logs the following syslog messages:

```

RP/0/0/1/CPU0:Feb 19 09:08:00.655 UTC: rmf_svr[436]: %HA-REDCON-6-GO_ACTIVE : this card going
active
RP/1/1/CPU0:Mar 8 11:43:29.041 UTC: rmf_svr[147]: %HA-REDCON-6-GO_STANDBY : this card going
standby, location RP/1/1/CPU0

```



CHAPTER 10

Configuring Network Time Protocol

Network Time Protocol (NTP) is a protocol designed to time-synchronize devices within a network. Cisco IOS XR software implements NTPv4. NTPv4 retains backwards compatibility with the older versions of NTP, including NTPv3 and NTPv2 but excluding NTPv1, which has been discontinued due to security vulnerabilities.

- [Prerequisites for Implementing NTP on Cisco IOS XR Software, on page 121](#)
- [Information About Implementing NTP, on page 121](#)
- [Configuration Examples for Implementing NTP, on page 139](#)
- [Configuring NTP server inside VRF interface, on page 142](#)

Prerequisites for Implementing NTP on Cisco IOS XR Software

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 NTP

NTP synchronizes timekeeping among a set of distributed time servers and clients. This synchronization allows events to be correlated when system logs are created and other time-specific events occur.

NTP uses the User Datagram Protocol (UDP) as its transport protocol. All NTP communication uses Coordinated Universal Time (UTC). An NTP network usually receives its time from an authoritative time source, such as a radio clock or an atomic clock attached to a time server. NTP distributes this time across the network. NTP is extremely efficient; no more than one packet per minute is necessary to synchronize two machines to within a millisecond of each other.

NTP uses the concept of a “stratum” to describe how many NTP “hops” away a machine is from an authoritative time source. A “stratum 1” time server typically has an authoritative time source (such as a radio or atomic clock, or a GPS time source) directly attached, a “stratum 2” time server receives its time via NTP from a “stratum 1” time server, and so on.

NTP avoids synchronizing to a machine whose time may not be accurate, in two ways. First, NTP never synchronizes to a machine that is not synchronized itself. Second, NTP compares the time reported by several machines and does not synchronize to a machine whose time is significantly different than the others, even if its stratum is lower. This strategy effectively builds a self-organizing tree of NTP servers.

The Cisco implementation of NTP does not support stratum 1 service; in other words, it is not possible to connect to a radio or atomic clock (for some specific platforms, however, you can connect a GPS time-source device). We recommend that time service for your network be derived from the public NTP servers available in the IP Internet.

If the network is isolated from the Internet, the Cisco implementation of NTP allows a machine to be configured so that it acts as though it is synchronized via NTP, when in fact it has determined the time using other means. Other machines can then synchronize to that machine via NTP.

Several manufacturers include NTP software for their host systems, and a publicly available version for systems running UNIX and its various derivatives is also available. This software also allows UNIX-derivative servers to acquire the time directly from an atomic clock, which would subsequently propagate time information along to Cisco routers.

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

The Cisco implementation of NTP supports three ways that a networking device can obtain NTP time information on a network:

- By polling host servers
- By listening to NTP broadcasts
- By listening to NTP multicasts
- By using a peer-to-peer relationship.

In a LAN environment, NTP can be configured to use IP broadcast or multicast messages. As compared to polling, IP broadcast or multicast messages reduce configuration complexity, because each machine can simply be configured to send or receive broadcast or multicast messages. However, the accuracy of timekeeping is marginally reduced because the information flow is one-way only.

An NTP broadcast client listens for broadcast messages sent by an NTP broadcast server at a designated IPv4 address. The client synchronizes the local clock using the first received broadcast message.

An NTP multicast server periodically sends a message to a designated IPv4 or IPv6 local multicast group address. An NTP multicast client listens on this address for NTP messages.

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

When multiple sources of time (VINES, hardware clock, manual configuration) are available, NTP is always considered to be more authoritative. NTP time overrides the time set by any other method.

Preventing Issues due to GPS Week Number Rollover (WNRO)

- If there are no GPS sources in the NTP source chain or server chain, there is no impact of GPS Week Number Rollover (WNRO).
- GPS WNRO affects only the system clock and not user traffic.
- Contact your GPS manufacturer to fix the GPS source for this condition.

To mitigate impact of GPS sources that are subject to GPS WNRO perform the following optional workarounds:

- If the GPS source has been identified to be a cause of potential disruption on April 6, 2019 (or after), configure `ntp master` in the Cisco that is device connected to this source, and its clock on the Stratum 1 device to preventively isolate it. This configuration enables the device to present its own clock for synchronization to downstream NTP clients.



Note The usage of `ntp master` command as mentioned above is only a workaround to this condition. Use this command until the GPS source-related conditions are resolved, and to prevent the distribution of incorrect clock values throughout the network.

- Configure multiple NTP servers (ideally 4, but more than 3) at Stratum 2 level of the network, to enable NTP clients at Stratum 2 level to get clock from more than one Stratum 1 server. This way, WNRO affected Stratum 1 servers are staged to be marked as ‘false ticker’ or ‘outlier’ clock sources as compared to other non-WNRO affected Stratum 1 servers.



Note To configure day light saving time (DST) on your IOS XR 64-bit device, select the appropriate country and city. The device will automatically update the DST based on the internal mappings at kernel level. The *DST* keyword is not available in the configuration CLI, since manual configuration of DST is not supported on IOS XR 64-bit devices.

Configuring Poll-Based Associations



Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

You can configure the following types of poll-based associations between the router and other devices (which may also be routers):

- Client mode
- Symmetric active mode

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

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

When a networking device is operating in the symmetric active mode, it polls its assigned time-serving hosts for the current time and it responds to polls by its hosts. Because this is a peer-to-peer relationship, the host also retains time-related information about the local networking device that it is communicating with. This mode should be used when there are several mutually redundant servers that are interconnected via diverse

network paths. Most stratum 1 and stratum 2 servers on the Internet today adopt this form of network setup. Use the **peer** command to individually specify the time-serving hosts that you want your networking device to consider synchronizing with and to set your networking device to operate in the symmetric active mode.

When the router polls several other devices for the time, the router selects one device with which to synchronize.



Note To configure a peer-to-peer association between the router and another device, you must also configure the router as a peer on the other device.

You can configure multiple peers and servers, but you cannot configure a single IP address as both a peer and a server at the same time.

To change the configuration of a specific IP address from peer to server or from server to peer, use the **no** form of the **peer** or **server** command to remove the current configuration before you perform the new configuration. If you do not remove the old configuration before performing the new configuration, the new configuration does not overwrite the old configuration.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **server** *ip-address* [**vrf** *vrf*] [**version** *number*] [**key** *key-id*] [**minpoll** *interval*] [**maxpoll** *interval*] [**source** *type interface-path-id*] [**prefer**] [**burst**] [**iburst**]
4. **peer** *ip-address* [**vrf** *vrf*] [**version** *number*] [**key** *key-id*] [**minpoll** *interval*] [**maxpoll** *interval*] [**source** *type interface-path-id*] [**prefer**]
5. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	ntp Example: RP/0/RP0/CPU0:router(config)# ntp	Enters NTP configuration mode.
Step 3	server <i>ip-address</i> [vrf <i>vrf</i>] [version <i>number</i>] [key <i>key-id</i>] [minpoll <i>interval</i>] [maxpoll <i>interval</i>] [source <i>type interface-path-id</i>] [prefer] [burst] [iburst] Example:	Forms a server association with another system. This step can be repeated as necessary to form associations with multiple devices.

	Command or Action	Purpose
	RP/0/RP0/CPU0:router(config-ntp)# server 172.16.22.44	
Step 4	<p>peer <i>ip-address</i> [vrf <i>vrf</i>] [version number] [key <i>key-id</i>] [minpoll interval] [maxpoll interval] [source type <i>interface-path-id</i>] [prefer]</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-ntp)# peer 192.168.22.33 source tengige 0/0/0/1</pre>	<p>Forms a peer association with another system. This step can be repeated as necessary to form associations with multiple systems.</p> <p>Note To complete the configuration of a peer-to-peer association between the router and the remote device, the router must also be configured as a peer on the remote device.</p>
Step 5	<p>Use one of the following commands:</p> <ul style="list-style-type: none"> • end • commit <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-ntp)# end</pre> <p>or</p> <pre>RP/0/RP0/CPU0:router(config-ntp)# commit</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring Broadcast-Based NTP Associates

In a broadcast-based NTP association, an NTP server propagates NTP broadcast packets throughout a network. Broadcast clients listen for the NTP broadcast packets propagated by the NTP server and do not engage in any polling.

Broadcast-based NTP associations should be used when time accuracy and reliability requirements are modest and if your network is localized and has a large number of clients (more than 20). Broadcast-based NTP associations also are recommended for use on networks that have limited bandwidth, system memory, or CPU resources. Time accuracy is marginally reduced in broadcast-based NTP associations because information flows only one way.

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

Use the **broadcast** command to set your networking device to send NTP broadcast packets.



Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. (Optional) **broadcastdelay** *microseconds*
4. **interface** *type interface-path-id*
5. **broadcast client**
6. **broadcast** [**destination** *ip-address*] [**key** *key-id*] [**version** *number*]
7. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	ntp Example: RP/0/RP0/CPU0:router(config)# ntp	Enters NTP configuration mode.
Step 3	(Optional) broadcastdelay <i>microseconds</i> Example: RP/0/RP0/CPU0:router(config-ntp)# broadcastdelay 5000	Adjusts the estimated round-trip delay for NTP broadcasts.
Step 4	interface <i>type interface-path-id</i> Example: RP/0/RP0/CPU0:router(config-ntp)# interface POS 0/1/0/0	Enters NTP interface configuration mode.

	Command or Action	Purpose
Step 5	broadcast client Example: <pre>RP/0/RP0/CPU0:router(config-ntp-int)# broadcast client</pre>	Configures the specified interface to receive NTP broadcast packets. Note Go to the next step to configure the interface to send NTP broadcast packets.
Step 6	broadcast [destination ip-address] [key key-id] [version number] Example: <pre>RP/0/RP0/CPU0:router(config-ntp-int)# broadcast destination 10.50.32.149</pre>	Configures the specified interface to send NTP broadcast packets. Note Go to previous step to configure the interface to receive NTP broadcast packets.
Step 7	Use one of the following commands: <ul style="list-style-type: none"> • end • commit Example: <pre>RP/0/RP0/CPU0:router(config-ntp-int)# end</pre> or <pre>RP/0/RP0/CPU0:router(config-ntp-int)# commit</pre>	Saves configuration changes. <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring NTP Access Groups



Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

The access list-based restriction scheme allows you to grant or deny certain access privileges to an entire network, a subnet within a network, or a host within a subnet. NTP communication consists of time requests and control queries. A *time request* is a request for time synchronization from an NTP server. A *control query* is a request for configuration information from an NTP server.

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

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

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

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

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **access-group** {**peer** | **query-only** | **serve** | **serve-only**} *access-list-name*
4. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	ntp Example: RP/0/RP0/CPU0:router(config)# ntp	Enters NTP configuration mode.
Step 3	access-group { peer query-only serve serve-only } <i>access-list-name</i> Example: RP/0/RP0/CPU0:router(config-ntp)# access-group peer access1	Creates an access group and applies a basic IPv4 or IPv6 access list to it.
Step 4	Use one of the following commands: <ul style="list-style-type: none"> • end • commit Example: RP/0/RP0/CPU0:router(config-ntp)# end	Saves configuration changes. <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: Uncommitted changes found, commit them before

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

Configuring NTP Authentication

This task explains how to configure NTP authentication.



Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

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

The authentication process begins from the moment an NTP packet is created. A message authentication code (MAC) is computed using the MD5 Message Digest Algorithm and the MAC is embedded into an NTP synchronization packet. The NTP synchronization packet together with the embedded MAC and key number are transmitted to the receiving client. If authentication is enabled and the key is trusted, the receiving client computes the MAC in the same way. If the computed MAC matches the embedded MAC, the system is allowed to sync to the server that uses this key in its packets.

After NTP authentication is properly configured, your networking device only synchronizes with and provides synchronization to trusted time sources.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **authenticate**
4. **authentication-key** *key-number* **md5** [**clear** | **encrypted**] *key-name*
5. **trusted-key** *key-number*

6. Use one of the following commands:

- **end**
- **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: <pre>RP/0/RP0/CPU0:router# configure</pre>	Enters global configuration mode.
Step 2	ntp Example: <pre>RP/0/RP0/CPU0:router(config)# ntp</pre>	Enters NTP configuration mode.
Step 3	authenticate Example: <pre>RP/0/RP0/CPU0:router(config-ntp)# authenticate</pre>	Enables the NTP authentication feature.
Step 4	authentication-key <i>key-number</i> md5 [clear encrypted] <i>key-name</i> Example: <pre>RP/0/RP0/CPU0:router(config-ntp)# authentication-key 42 md5 clear key1</pre>	Defines the authentication keys. <ul style="list-style-type: none"> • Each key has a key number, a type, a value, and, a name. Currently the only key type supported is md5.
Step 5	trusted-key <i>key-number</i> Example: <pre>RP/0/RP0/CPU0:router(config-ntp)# trusted-key 42</pre>	Defines trusted authentication keys. <ul style="list-style-type: none"> • If a key is trusted, this router only synchronizes to a system that uses this key in its NTP packets.
Step 6	Use one of the following commands: <ul style="list-style-type: none"> • end • commit Example: <pre>RP/0/RP0/CPU0:router(config-ntp)# end</pre> or <pre>RP/0/RP0/CPU0:router(config-ntp)# commit</pre>	Saves configuration changes. <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Disabling NTP Services on a Specific Interface

NTP services are disabled on all interfaces by default.

NTP is enabled globally when any NTP commands are entered. You can selectively prevent NTP packets from being received through a specific interface by turning off NTP on a given interface.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. Use one of the following commands:
 - **no interface** *type interface-path-id*
 - **interface** *type interface-path-id* **disable**
4. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	ntp Example: RP/0/RP0/CPU0:router(config)# ntp	Enters NTP configuration mode.
Step 3	Use one of the following commands: <ul style="list-style-type: none"> • no interface <i>type interface-path-id</i> 	Disables NTP services on the specified interface.

	Command or Action	Purpose
	<ul style="list-style-type: none"> • interface <i>type interface-path-id</i> disable <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-ntp)# no interface pos 0/0/0/1</pre> <p>or</p> <pre>RP/0/RP0/CPU0:router(config-ntp)# interface POS 0/0/0/1 disable</pre>	
Step 4	<p>Use one of the following commands:</p> <ul style="list-style-type: none"> • end • commit <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-ntp)# end</pre> <p>or</p> <pre>RP/0/RP0/CPU0:router(config-ntp)# commit</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring the Source IP Address for NTP Packets

By default, the source IP address of an NTP packet sent by the router is the address of the interface through which the NTP packet is sent. Use this procedure to set a different source address.



Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

SUMMARY STEPS

1. **configure**
2. **ntp**

3. **source** *type interface-path-id*
4. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: <pre>RP/0/RP0/CPU0:router# configure</pre>	Enters global configuration mode.
Step 2	ntp Example: <pre>RP/0/RP0/CPU0:router(config)# ntp</pre>	Enters NTP configuration mode.
Step 3	source <i>type interface-path-id</i> Example: <pre>RP/0/RP0/CPU0:router(config-ntp)# source POS 0/0/0/1</pre>	Configures an interface from which the IP source address is taken. Note This interface is used for the source address for all packets sent to all destinations. If a source address is to be used for a specific association, use the source keyword in the peer or server command shown in Configuring Poll-Based Associations, on page 123 .
Step 4	Use one of the following commands: <ul style="list-style-type: none"> • end • commit Example: <pre>RP/0/RP0/CPU0:router(config-ntp)# end</pre> or <pre>RP/0/RP0/CPU0:router(config-ntp)# commit</pre>	Saves configuration changes. <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.

	Command or Action	Purpose
		<ul style="list-style-type: none"> Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring the System as an Authoritative NTP Server

You can configure the router to act as an authoritative NTP server, even if the system is not synchronized to an outside time source.



Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **master** *stratum*
4. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# <code>configure</code>	Enters global configuration mode.
Step 2	ntp Example: RP/0/RP0/CPU0:router(config)# <code>ntp</code>	Enters NTP configuration mode.
Step 3	master <i>stratum</i> Example: RP/0/RP0/CPU0:router(config-ntp)# <code>master 9</code>	Makes the router an authoritative NTP server. Note Use the master command with caution. It is very easy to override valid time sources using this command, especially if a low stratum number is configured. Configuring multiple machines in the same network with the master command can cause instability in time keeping if the machines do not agree on the time.

	Command or Action	Purpose
Step 4	<p>Use one of the following commands:</p> <ul style="list-style-type: none"> • end • commit <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-ntp)# end</pre> <p>or</p> <pre>RP/0/RP0/CPU0:router(config-ntp)# commit</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

FQDN for NTP Server

NTP on Cisco IOS XR Software supports configuration of servers and peers using their Fully Qualified Domain Names (FQDN). While configuring, the FQDN is resolved via DNS into its corresponding IPv4 or IPv6 address and is stored in the running-configuration of the system. NTP supports FQDN for both IPv4 and IPv6 protocols. You can configure FQDN on default vrf.

Configure FQDN for NTP server

Configuration Example for FQDN on NTP Server on Default VRF

Use the **ntp server** command with the FQDN name to configure FQDN on default VRF. You don't need to specify VRF name. In the following example, time.cisco.com is the FQDN.

```
Router#configure
Router(config)#ntp server time.cisco.com
Router(config)#commit
```



Note When you are configuring FQDN over default VRF, you don't need to specify VRF name.

Running Configuration

Use the **show running-config ntp** command to see the ntp running configuration.

```
Router#show running-config ntp
ntp
server 10.48.59.212
!
```

Verification

Use the **show ntp associations** command to verify that an NTP association has come up.

```
Router#show ntp associations

      address          ref clock      st  when  poll reach  delay  offset  disp
~10.48.59.212      173.38.201.67      2   42   128    3  196.06  -14.25  3949.4
* sys_peer, # selected, + candidate, - outlayer, x falseticker, ~ configured
```

Updating the Hardware Clock

On devices that have hardware clocks (system calendars), you can configure the hardware clock to be periodically updated from the software clock. This is advisable for devices using NTP, because the time and date on the software clock (set using NTP) is more accurate than the hardware clock. The time setting on the hardware clock has the potential to drift slightly over time.



Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **update-calendar**
4. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	ntp Example: RP/0/RP0/CPU0:router(config)# ntp	Enters NTP configuration mode.
Step 3	update-calendar Example:	Configures the router to update its system calendar from the software clock at periodic intervals.

	Command or Action	Purpose
	RP/0/RP0/CPU0:router(config-ntp)# update-calendar	
Step 4	<p>Use one of the following commands:</p> <ul style="list-style-type: none"> • end • commit <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-ntp)# end</pre> <p>or</p> <pre>RP/0/RP0/CPU0:router(config-ntp)# commit</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Verifying the Status of the External Reference Clock

This task explains how to verify the status of NTP components.



Note The commands can be entered in any order.

SUMMARY STEPS

1. **show ntp associations [detail] [location node-id]**
2. **show ntp status [location node-id]**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>show ntp associations [detail] [location node-id]</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router# show ntp associations</pre>	Displays the status of NTP associations.

	Command or Action	Purpose
Step 2	show ntp status [location node-id] Example: RP/0/RP0/CPU0:router# show ntp status	Displays the status of NTP.

Examples

The following is sample output from the **show ntp associations** command:

```
RP/0/RP0/CPU0:router# show ntp associations
```

```
Tue Oct 7 11:22:46.839 JST
```

```

      address      ref clock      st  when  poll reach  delay  offset  disp
*~192.168.128.5    10.81.254.131    2    1    64  377    7.98  -0.560  0.108
+~dead:beef::2 vrf testAA
                  171.68.10.80    3    20   64  377    6.00  -2.832  0.046
* sys_peer, # selected, + candidate, - outlayer, x falseticker, ~ configured

```

```
RP/0/RP0/CPU0:router# show ntp associations
```

```

      address      ref clock      st  when  poll reach  delay  offset  disp
+~127.127.1.1      127.127.1.1      5    5  1024   37    0.0    0.00  438.3
*~172.19.69.1      172.24.114.33    3   13  1024    1    2.0   67.16  0.0
* master (syncd), # master (unsyncd), + selected, - candidate, ~ configured

```

The following is sample output from the **show ntp status** command:

```
RP/0/RP0/CPU0:router# show ntp status
```

```
Tue Oct 7 11:22:54.023 JST
```

```

Clock is synchronized, stratum 3, reference is 192.168.128.5
nominal freq is 1000.0000 Hz, actual freq is 1000.2725 Hz, precision is 2**24
reference time is CC95463C.9B964367 (11:21:48.607 JST Tue Oct 7 2008)
clock offset is -1.738 msec, root delay is 186.050 msec
root dispersion is 53.86 msec, peer dispersion is 0.09 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is -0.0002724105 s/s
system poll interval is 64, last update was 66 sec ago

```

```
RP/0/RP0/CPU0:router# show ntp status
```

```

Clock is synchronized, stratum 4, reference is 172.19.69.1
nominal freq is 1000.0000 Hz, actual freq is 999.9988 Hz, precision is 2**26
reference time is C54C131B.9EECF6CA (07:26:19.620 UTC Mon Nov 24 2008)
clock offset is 66.3685 msec, root delay is 7.80 msec
root dispersion is 950.04 msec, peer dispersion is 3.38 msec

```

Configuration Examples for Implementing NTP

Configuring Poll-Based Associations: Example

The following example shows an NTP configuration in which the router's system clock is configured to form a peer association with the time server host at IP address 192.168.22.33, and to allow the system clock to be synchronized by time server hosts at IP address 10.0.2.1 and 172.19.69.1:

```
ntp
  server 10.0.2.1
  peer 192.168.22.33

  server 172.19.69.1
```

Configuring Broadcast-Based Associations: Example

The following example shows an NTP client configuration in which interface 0/2/0/0 is configured to receive NTP broadcast packets, and the estimated round-trip delay between an NTP client and an NTP broadcast server is set to 2 microseconds:

```
ntp
  interface tengige 0/2/0/0
    broadcast client
  exit
  broadcastdelay 2
```

The following example shows an NTP server configuration where interface 0/2/0/2 is configured to be a broadcast server:

```
ntp
  interface tengige 0/2/0/2
    broadcast
```

Configuring Multicast-Based Associations: Example

The following example shows an NTP multicast client configuration where 10-Gigabit Ethernet interface 0/1/1/0 is configured to be a multicast client and to join the default multicast group (IPv4 address 224.0.1.1):

```
ntp interface TenGigE 0/1/1/0
  multicast client
```

The following example shows an NTP multicast server configuration where 10-Gigabit Ethernet interface 0/1/1/0 is configured to be a multicast server:

```
ntp interface TenGigE 0/1/1/0
```

```
multicast destination 224.0.1.1
```

Configuring NTP Access Groups: Example

The following example shows a NTP access group configuration where the following access group restrictions are applied:

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

```
ntp
 peer 10.1.1.1
 peer 10.1.1.1
 peer 10.2.2.2
 peer 10.3.3.3
 peer 10.4.4.4
 peer 10.5.5.5
 peer 10.6.6.6
 peer 10.7.7.7
 peer 10.8.8.8
 access-group peer peer-acl
 access-group serve serve-acl
 access-group serve-only serve-only-acl
 access-group query-only query-only-acl
 exit
ipv4 access-list peer-acl
 10 permit ip host 10.1.1.1 any
 20 permit ip host 10.8.8.8 any
 exit
ipv4 access-list serve-acl
 10 permit ip host 10.4.4.4 any
 20 permit ip host 10.5.5.5 any
 exit
ipv4 access-list query-only-acl
 10 permit ip host 10.2.2.2 any
 20 permit ip host 10.3.3.3 any
 exit
ipv4 access-list serve-only-acl
 10 permit ip host 10.6.6.6 any
 20 permit ip host 10.7.7.7 any
 exit
```

Configuring NTP Authentication: Example

The following example shows an NTP authentication configuration. In this example, the following is configured:

- NTP authentication is enabled.

- Two authentication keys are configured (key 2 and key 3).
- The router is configured to allow its software clock to be synchronized with the clock of the peer (or vice versa) at IP address 10.3.32.154 using authentication key 2.
- The router is configured to allow its software clock to be synchronized with the clock by the device at IP address 10.32.154.145 using authentication key 3.
- The router is configured to synchronize only to systems providing authentication key 3 in their NTP packets.

```
ntp
authenticate
authentication-key 2 md5 encrypted 06120A2D40031D1008124
authentication-key 3 md5 encrypted 1311121E074110232621
trusted-key 3
server 10.3.32.154 key 3
peer 10.32.154.145 key 2
```

Disabling NTP on an Interface: Example

The following example shows an NTP configuration in which 0/2/0/0 interface is disabled:

```
ntp
interface tengige 0/2/0/0
disable
exit
authentication-key 2 md5 encrypted 06120A2D40031D1008124
authentication-key 3 md5 encrypted 1311121E074110232621
authenticate
trusted-key 3
server 10.3.32.154 key 3
peer 10.32.154.145 key 2
```

Configuring the Source IP Address for NTP Packets: Example

The following example shows an NTP configuration in which Ethernet management interface 0/0/CPU0/0 is configured as the source address for NTP packets:

```
ntp
authentication-key 2 md5 encrypted 06120A2D40031D1008124
authentication-key 3 md5 encrypted 1311121E074110232621
authenticate
trusted-key 3
server 10.3.32.154 key 3
peer 10.32.154.145 key 2
source MgmtEth0/0/CPU0/0
```

Configuring the System as an Authoritative NTP Server: Example

The following example shows a NTP configuration in which the router is configured to use its own NTP master clock to synchronize with peers when an external NTP source becomes unavailable:

```
ntp
 master 6
```

Updating the Hardware Clock: Example

The following example shows an NTP configuration in which the router is configured to update its hardware clock from the software clock at periodic intervals:

```
ntp
 server 10.3.32.154
 update-calendar
```

Configuring NTP server inside VRF interface

This task explains how to configure NTP server inside VRF interface.



Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **vrf** *vrf-name*
4. **source** *interface-type interface-instance*
5. Use one of the following commands:
 - **end**
 - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.

	Command or Action	Purpose
Step 2	ntp Example: RP/0/RP0/CPU0:router(config)# ntp	Enters NTP configuration mode.
Step 3	vrf vrf-name Example: RP/0/RP0/CPU0:router(config)# ntp vrf Customer_A	Specify name of a VRF (VPN- routing and forwarding) instance to configure.
Step 4	source interface-type interface-instance Example: RP/0/RP0/CPU0:router(config)# ntp vrf Customer_A source bvi 70	Configures an interface from which the IP source address is taken. This allows IOS-XR to respond to NTP queries on VRF interfaces, in this case the source is BVI. Note This interface is used for the source address for all packets sent to all destinations. If a source address is to be used for a specific association, use the source keyword in the peer or server command shown in Configuring Poll-Based Associations, on page 123 .
Step 5	Use one of the following commands: <ul style="list-style-type: none"> • end • commit Example: RP/0/RP0/CPU0:router(config-ntp)# end or RP/0/RP0/CPU0:router(config-ntp)# commit	Saves configuration changes. <ul style="list-style-type: none"> • When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.



CHAPTER 11

Configuring Precision Time Protocol

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.

This module contains the following topics:

- [PTP Overview, on page 145](#)
- [ITU-T Telecom Profiles for PTP, on page 157](#)
- [Configuring PTP , on page 161](#)
- [Configuration Examples, on page 172](#)

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 6: 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.

Network Element	Description
Ordinary Clock (OC)	<p>An ordinary clock is a 1588 clock with a single PTP port that can operate in one of the following modes:</p> <ul style="list-style-type: none"> • 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. • 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.
Boundary Clock (BC)	<p>The device participates in selecting the best master clock and can act as the master clock if no better clocks are detected.</p> <p>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.</p>
Transparent Clock (TC)	<p>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.</p>

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-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 [Configuring PTP Hybrid Mode](#) is described in a subsequent section in this chapter.

Time of Day (ToD) Support

The router receives GPS ToD messages in serial ASCII stream through the RS422 interface in any of the following formats:

- NTP Type 4
- Cisco
- NMEA - GPZDA



Note You can refer to the below support information in context of the current release and see relevant *Release Notes* for more information on supported features and hardware.

Port States

State machine indicates the behavior of each port. The possible states are:

State	Description
INIT	Port is not ready to participate in PTP.

State	Description
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 NCS 5500 Series Router:

- NCS55-RP does not support PTP
- NC55-18H18F line card does not support PTP
- SyncE is not supported on a 1GE copper SFP.
- SyncE is not supported on 25 GE or 100 GE interfaces when they are used in 1G mode.
- 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 MACSec.
- G.8273.2 Class-A performance is met if CFP2-DCO is configured on either Slave or Master port on the node.
- Transparent Clock is not supported.
- PTP over MPLS is not supported.



Note

- We recommend you to configure, and enable Frequency Synchronization selection input on two interfaces per line card.
- For link aggregation, you must configure and enable Frequency Synchronization selection input on a single bundle member.

PTP Support Information

This table lists different types of support information related to PTP:

Transport Media	<ul style="list-style-type: none"> • UDP over IPv4 • Ethernet
Messages	<ul style="list-style-type: none"> • Signaling • Announce • Sync • Follow-up • Delay-request • Delay-response • Management
Transport Modes	<ul style="list-style-type: none"> • Unicast: This is the default mode. All packets are sent as unicast messages. Unicast is applicable only for PTP over IP profiles. • Multicast: All packets are sent as multicast messages. Multicast is the only mode for PTP over ethernet profiles.

PTP Hardware Support Matrix



Note The table also contains support details of upcoming releases. You can read this table in context of the current release and see relevant *Release Notes* for more information on supported features and hardware.

This table provides a detailed information on the timing features that are supported on the following hardware variants.

Hardware Variant	Features	Cisco IOS XR Release	Comments
NC57-48Q2D-S NC57-48Q2D-SE-S	G8275.1	Release 7.10.1	<p>With this release, SyncE and PTP Class-C, Class-B performance is supported on 1G, 10G, 25G, 40G and 100G port speeds.</p> <p>On 50G and 400G ports speeds, only timing functionality is supported.</p> <p>PTP support is available on compatible mode.</p> <p>PTP with Class-C is not achieved with macsec on any interface speed.</p> <p>Note For 1G Class C port speed, only port 32 and 40 are supported. It is not recommended to plug in 1G optics to ports greater than or equal to port 32.</p>
NC57-36H6D-S	G8265.1	Release 7.10.1	<p>With this release, timing support for PTP and SyncE is extended to 4x10G and 4x25G breakout ports of NC57-36H6D-S in native mode.</p> <p>Class B and Class C performances are supported on 4x10G and 4x25G breakout ports in native mode. Route Processor: NC55-RP2-E</p>
	G8275.1		
	G8275.2		
	Default Profile		

Hardware Variant	Features	Cisco IOS XR Release	Comments
NC57-36H-SE	G8265.1	Release 7.10.1	<p>With this release, timing support for PTP and SyncE is extended to 4x10G breakout port of NC57-36H-SE is in native mode.</p> <p>Class B performance is supported on 4x10G breakout port in native mode.</p> <p>Route Processor: NC55-RP2-E</p>
	G8275.1		
	G8275.2		
	Default Profile		
NCS-57C1-48Q6-SYS	G.8265.1	Release 7.10.1	<p>G.8273.2 Class C is supported on 400G interfaces with the following optics modules:</p> <ul style="list-style-type: none"> • Cisco QSFPDD 400G FR4 Pluggable Optics Module • Cisco QSFPDD 400G LR4 Pluggable Optics Module
	G.8275.1		
	G.8275.2		
	Default Profile		
G.8275.1	Release 7.8.1	Release 7.7.1	
G.8275.2	Release 7.8.1		
Default Profile	Release 7.8.1		
NCS-57C3-MODS-SYS NCS-57C3-MODS-SYS	PTP Virtual Port and APTS		
NCS-57B1-6D24-SYS	PTP Virtual Port and APTS	Release 7.7.1	
NCS-57C1-48Q6-SYS	Default profile	Release 7.5.1	
	G.8265.1	Release 7.5.1	
	G.8275.1	Release 7.5.1	
	G.8275.2	Release 7.5.1	
RP:NC57-MOD-RP-2E with NCS573-MODS-SYS and NCS-573-MOD-SYS	G.8275.1	Release 7.4.1	
	G.8273.2	Release 7.4.1	
	GNSS	Release 7.4.1	

Hardware Variant	Features	Cisco IOS XR Release	Comments
NCS-57B1-5DSE-SYS	Default profile	Release 7.3.1	
NCS-57B1-6D24-SYS	G.8265.1	Release 7.3.1	
	G.8275.1	Release 7.3.1	
	G.8275.2	Release 7.3.1	
RP: NC55-RP2-E Line card: NC57-36H6D-S	G.8275.1	Release 7.3.2	<ul style="list-style-type: none"> • Release 7.3.2 - Supports Compatible Mode only • Release 7.7.1 - Supports both Native and Compatible mode.
	G.8273.2	Release 7.3.2	<ul style="list-style-type: none"> • Release 7.3.2 - Supports Compatible Mode only • Release 7.7.1 - Supports both Native and Compatible mode.
RP:NC55-RP-E with Line cards: NC55-MOD-A-S and NC55-32T16Q4H-AT	BITS	Release 7.1.1	
	G8275.1	Release 7.1.1	For the profile G8275.1 NC55-32T16Q4H-AT supports only T-BC and does not support T-GM. 25G/100G/40G is supported from IOSXR release 7.2.2 onwards.
	G8273.2	Release 7.1.1	Class B
RP:NC55-RP2-E with Line cards: NC55-MOD-A-S and NC55-32T16Q4H-AT	BITS	Release 7.1.1	
	G.8275.1	Release 7.1.1	For the profile G8275.1 NC55-32T16Q4H-AT supports only T-BC and does not support T-GM. 25G/100G/40G is supported from IOSXR release 7.2.2 onwards.
	G.8273.2	Release 7.1.1	Class B

Hardware Variant	Features	Cisco IOS XR Release	Comments
RP:NC55-RP2-E with Line card:NC55-32T16Q4H-AT	BITS	Release 7.1.1	
	G.8275.1	Release 7.1.1	For the profile G.8275.1 NC55-32T16Q4H-AT supports only T-BC and does not support T-GM. 25G/100G/40G is supported from IOSXR release 7.2.2 onwards.
	G.8273.2	Release 7.1.1	Class C
NCS-55A1-36H-SE-S	G.8265.1	Release 7.0.1	
	G.8275.1	Release 7.0.1	
	G.8275.2	Release 7.0.1	
	G.8273.2	Release 7.0.1	Class B
NCS-55A1-36H-S	G.8265.1	Release 7.0.1	
	G.8275.1	Release 7.0.1	
	G.8275.2	Release 7.0.1	
	G.8273.2	Release 7.0.1	Class B
NCS-55A1-24Q6H-S NCS-55A1-24Q6H-SS	G.8265.1	Release 6.6.25	
	G.8275.1	Release 6.6.25	
	G.8275.2	Release 6.6.25	From Release 7.7.1, support is available for PTP over IPv6 for ports 10G-25G and 40G-100G
	G.8273.2	Release 6.6.25	Class B
NCS-55A1-48Q6H	G.8265.1	Release 6.6.25	
	G.8275.1	Release 6.6.25	
	G.8275.2	Release 6.6.25	
	G.8273.2	Release 6.6.25	Class B
NCS-55A1-24H	G.8265.1	Release 6.5.2	
	G.8275.1	Release 6.5.2	
	G.8275.2	Release 6.5.2	
	G.8273.2	Release 6.5.2	Class B

Hardware Variant	Features	Cisco IOS XR Release	Comments
NCS55A2-MOD	G.8265.1	Release 6.5.1	
	G.8275.1	Release 6.5.1	
	G.8275.2	Release 6.5.1	
	G.8273.2	Release 6.5.1	Class B
RP:NC55-RP-E Linecard:NC55-MOD-A-S	BITS	Release 6.5.1	SyncE is not supported on 25GE or 100GE interfaces, when they are used in 1G mode.
	G.8265.1	Release 6.5.1	
	G.8275.1	Release 6.5.1	
	G.8275.2	Release 6.5.1	This profile is supported from Release 6.5.1 for Ipv4.
	G.8273.2	Release 6.5.1	Class B
RP:NC55-RP-E Linecard:NC55-36X100G-A-SE	G.8273.2	Release 6.3.2	Class B
	BITS	Release 6.3.2	SyncE is not supported on 25GE or 100GE interfaces, when they are used in 1G mode.
	G.8265.1	Release 6.3.2	
	G.8275.1	Release 6.3.2	
	G.8275.2	NA	
	G.8273.2	Release 6.3.2	Class B
NCS5501-SE	G.8265.1	Release 6.3.2	
	G.8275.1	Release 6.3.2	Class B
	G.8275.2	Release 6.3.2	
	GNSS External	Release 6.3.2	

Hardware Variant-Specific Behaviour

The line card NC55-32T16Q4H-AT displays the following behaviour when configured for PTP:

- The timing features are supported on all ports of the line cards.
- The NC55-RP2-E does not support PTP on the 1588 Port.

- To configure Class C for the profile G.8273.2 when you use NC55-RP2-E with line card NC55-32T16Q4H-AT, follow the example below:

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)#frequency synchronization
RP/0/RP0/CPU0:router(config-freqsync)#timing-accuracy enhanced
RP/0/RP0/CPU0:router(config-freqsync)#commit
```



Note Cisco NCS 5500 Series Routers support 64 PTP clients at 64 PPS sync packet rate.

Timing features are supported on the following MPAs:

- NC55-MPA-2TH-S
- NC55-MPA-1TH2H-S
- NC55-MPA-1TH2H-HD-S
- NC55-MPA-4H-S
- NC55-MPA-4H-HD-S
- NC55-MPA-12T-S

Breakout Timing Support

PTP Profiles 8275.1 and 8275.2 are supported on breakout ports on the following hardware PIDs:

Table 7: Breakout Timing Support Hardware Matrix

Hardware PID	Client Port	Server Port
NCS-55A1-36H-S	100G	25G Breakout
NCS-55A1-36H-S	100G	10G Breakout
NCS-55A1-48Q6H	10G	25G Breakout
NCS-55A1-48Q6H	100G	25G Breakout
NCS55A1-24Q6H-S	1G	25G Breakout
NCS55A1-24Q6H-S	10G	25G Breakout
NCS55A1-24Q6H-S	100G	25G Breakout
NCS-5501-SE	1G	10G Breakout
NCS-5501-SE	1G	25G Breakout
NCS-5501-SE	10G	10G Breakout
NCS-5501-SE	10G	25G Breakout

Hardware PID	Client Port	Server Port
NC57-36H6D-S	25G	25G Breakout
NC57-36H6D-S	25G	10G Breakout
NC57-36H6D-S	10G	25G Breakout
NC57-36H6D-S	10G	10G Breakout
NC57-36H-SE	10G	10G Breakout



Note The server ports 100G and 40G are used as breakout for 4x25G and 4x10G respectively. The client ports are used as direct ports of different port speeds as presented in the table, *Breakout Timing Support Hardware Matrix*.

Slow Tracking

Under normal configured conditions, any change in offset triggers an immediate reaction in the servo. With the Slow Tracking feature enabled, the servo corrects the phase offset based on the configured value. If the phase offset exceeds the acceptable range, servo goes into Holdover state. In such a condition, the Slow Tracking feature becomes inactive and the servo corrects itself to the latest offset and goes into Phase locked state. Slow Tracking becomes active again.



Note

- The supported slow tracking rate range is from 8-894 nanoseconds per second and must be in multiples of 8.
- This feature is active only when servo is in Phase locked mode.

```
Router:~# config
ptp
clock
domain 24
profile g.8275.1 clock-type T-BC
!
profile profile1
multicast target-address ethernet 01-1B-19-00-00-00
transport ethernet
sync frequency 16
clock operation one-step
announce frequency 8
delay-request frequency 16
!
physical-layer-frequency
servo-slow-tracking 16
!
```

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. The default domain number is 4.
- Port Numbers: All PTP port numbers can only be one (1) because all clocks in this profile network are Ordinary Clocks.

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.

G.8275.1

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:

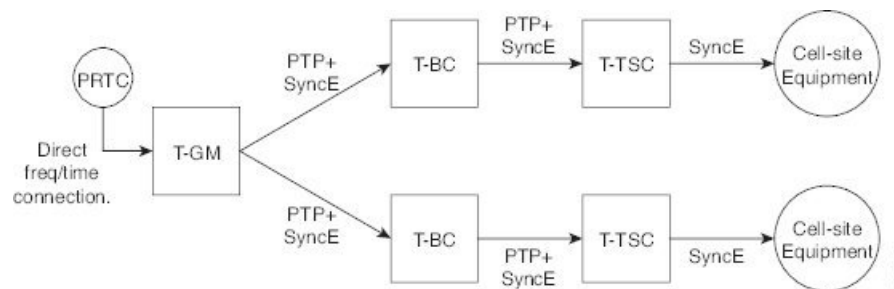
- **Telecom Grandmaster (T-GM):** Provides timing for other network devices and does not synchronize its local clock to other network devices.
 - **Telecom Time Slave Clock (T-TSC):** A slave clock synchronizes its local clock to another PTP clock, but does not provide PTP synchronization to any other 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 4: A Sample G.8275.1 Topology



Route Processor Fail Over

The Route processor fail over (RPFO) or stateful switchover (SSO) feature is supported on the Profile G.8275.1 on the Telecom Boundary Clock. Over a switchover, the time error might jump to a high value after losing lock with the T-GM clock. With this feature enabled, the time error will not increase by more than 400 nanoseconds over a switchover.



Note You must wait for a specific time duration to lapse to build holdover.

G.8275.2

The G.8275.2 is a PTP profile for use in telecom networks where phase or time-of-day synchronization is required. It differs from G.8275.1 in that it is not required that each device in the network participates in the PTP protocol. Also, G.8275.2 uses PTP over IPv4 in unicast mode.

The G.8275.2 profile is based on the partial timing support from the network. Hence nodes using G.8275.2 are not required to be directly connected.

The G.8275.2 profile is used in mobile cellular systems that require accurate synchronization of time and phase. For example, the fifth generation (5G) of mobile telecommunications technology.



Note G.8275.2 profile is supported on Cisco NCS 5500 Series Routers. However, the performance standards of this profile is not aligned with any of the ITU-T standards because performance specifications for G.8275.2 profile has not yet been made available by ITU-T.

For more information on hardware that supports G.8275.2 profile configurations, refer to [PTP Support Information](#) section in this chapter.

Features of G.8275.2 profile are:

- *Clock Selection*: G.8275.2 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



Note See ITU-T G.8275.2 document to determine the valid values for Clock Class parameter.

- *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 unicast transport mode.
- *Packet Rates*:
 - Synchronization/Follow-Up—minimum is one packet-per-second and maximum of 128 packets-per-second.

- Packet rate for Announce packets—minimum of one packet-per-second and maximum of eight packets-per-second.
- Delay-Request/Delay-Response packets—minimum is one packet-per-second and maximum of 128 packets-per-second
- *Transport Mechanism*: G.8275.2 profile supports only IPv4 PTP transport mechanism.
- *Mode*: G.8275.2 profile supports transport of data packets only in unicast mode.
- *Clock Type*: G.8275.2 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.8275.2 profile network ranges from 44 to 63. The default domain number is 44.

Starting from Release 7.2.1, PTP Multi-profile is supported for the below combination of PTP profiles:

- G8275.1
- G8275.2

Configuring PTP

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 tasks you need to configure PTP on Cisco IOS XR software.



Note When a subinterface is configured with encapsulation default or untag configuration, you must configure PTP on that subinterface, instead of the main interface.

Configuring Global G.8275.1 Profile

This below 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.8275.1 profile as an example:



Note Prior to Cisco IOS XR Software Release 6.3.3, the default PTP timers for G2875.1 were not set to standard values. This could lead to interoperability issues with other routers running the timers with updated values. Hence, to prevent such issues arising due to difference in packet rates, you must explicitly configure the **announce interval** value to 8, **sync frequency** value to 16 and **delay-request frequency** value to 16 while configuring global g.2875.1 profile.

```
RP/0/RP0/CPU0:router# config terminal
RP/0/RP0/CPU0:router(config)# ptp
RP/0/RP0/CPU0:router(config-ptp)# clock
RP/0/RP0/CPU0:router(config-ptp-clock)# domain 24
RP/0/RP0/CPU0:router(config-ptp-clock)# profile g.8275.1 clock-type T-BC
RP/0/RP0/CPU0:router(config-ptp-clock)# exit
RP/0/RP0/CPU0:router(config-ptp)# profile slave
RP/0/RP0/CPU0:router(config-ptp-profile)# multicast target-address ethernet 01-1B-19-00-00-00
RP/0/RP0/CPU0:router(config-ptp-profile)# transport ethernet
RP/0/RP0/CPU0:router(config-ptp-profile)# sync frequency 16
RP/0/RP0/CPU0:router(config-ptp-profile)# announce frequency 8
RP/0/RP0/CPU0:router(config-ptp-profile)# delay-request frequency 16
RP/0/RP0/CPU0:router(config-ptp-profile)# exit
RP/0/RP0/CPU0:router(config-ptp)# profile master
RP/0/RP0/CPU0:router(config-ptp-profile)# multicast target-address ethernet 01-1B-19-00-00-00
RP/0/RP0/CPU0:router(config-ptp-profile)# transport ethernet
RP/0/RP0/CPU0:router(config-ptp-profile)# sync frequency 16
RP/0/RP0/CPU0:router(config-ptp-profile)# announce frequency 8
RP/0/RP0/CPU0:router(config-ptp-profile)# delay-request frequency 16
RP/0/RP0/CPU0:router(config-ptp-profile)# exit
RP/0/RP0/CPU0:router(config-ptp)# physical-layer-frequency
RP/0/RP0/CPU0:router(config-ptp)# log
RP/0/RP0/CPU0:router(config-ptp-log)# servo events
RP/0/RP0/CPU0:router(config-ptp-log)# commit
```

Verification

To display the configured PTP profile details, use **show run ptp** command.

```
RP/0/RP0/CPU0: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.

```
RP/0/RP0/CPU0:router# configure terminal
RP/0/RP0/CPU0:router(config)# interface HundredGigE0/0/0/0
RP/0/RP0/CPU0:router(config-if)# ptp
RP/0/RP0/CPU0:router(config-if-ptp)# profile master
RP/0/RP0/CPU0:router(config-if-ptp)# port state master-only
RP/0/RP0/CPU0:router(config-if-ptp)# commit
```

Verification

To verify the port state details, use **show run interface** *interface-name* command.

```
RP/0/RP0/CPU0: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.

```
RP/0/RP0/CPU0:router# configure terminal
RP/0/RP0/CPU0:router(config)# interface HundredGigE0/0/0/1
RP/0/RP0/CPU0:router(config-if)# ptp
RP/0/RP0/CPU0:router(config-if-ptp)# profile slave
RP/0/RP0/CPU0:router(config-if-ptp)# port state slave-only
RP/0/RP0/CPU0:router(config-if-ptp)# commit
```

Verification

To verify the port state details, use **show run interface** *interface-name* command.

```
RP/0/RP0/CPU0: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. By default, the hybrid mode is used, regardless of the physical-layer-frequency configuration.
- G.8275.2 PTP profile supports both hybrid mode and non-hybrid mode. By default, the non-hybrid mode is used. Hybrid mode is used only when the physical-layer-frequency is configured.

To configure PTP Hybrid mode:

1. Configure Global Frequency Synchronization

```
RP/0/RP0/CPU0:router(config)# frequency synchronization
RP/0/RP0/CPU0: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.

```
RP/0/RP0/CPU0:router(config)# interface GigabitEthernet 0/0/0/0
RP/0/RP0/CPU0:router(config-if)# frequency synchronization
RP/0/RP0/CPU0:router(config-if-freqsync)# selection input
RP/0/RP0/CPU0:router(config-if-freqsync)# time-of-day-priority 100
RP/0/RP0/CPU0:router(config-if-freqsync)# commit
```

3. Configure Global PTP. To configure PTP as source for ToD, use ToD priority values in the range from 1 (highest priority) to 254 (lowest priority). Use frequency from the physical layer.

```
RP/0/RP0/CPU0:router(config)# ptp
RP/0/RP0/CPU0:router(config-ptp)# physical-layer-frequency
RP/0/RP0/CPU0:router(config-ptp)# time-of-day priority 1
RP/0/RP0/CPU0:router(config)# commit
```

4. Configure PTP Interface. To enable this interface as a PTP Master, use **master** command in ptp-interface configuration mode.

```
RP/0/RP0/CPU0:router(config)# interface GigabitEthernet 0/0/0/2
RP/0/RP0/CPU0:router(config-if)# ipv4 address 10.0.0.1/24
RP/0/RP0/CPU0:router(config-if)# ptp
RP/0/RP0/CPU0:router(config-if-ptp)# master ipv4 10.0.0.2
RP/0/RP0/CPU0:router(config-if-ptp)# commit
```

Verifying PTP Hybrid Mode

```
RP/0/RP0/CPU0:router # show frequency synchronization selection
```

```
Tue Feb  6 06:34:17.627 UTC
Node 0/0/CPU0:
=====
Selection point: ETH_RXMUX (1 inputs, 1 selected)
Last programmed 3d23h ago, and selection made 3d23h ago
Next selection points
  SPA scoped      : None
  Node scoped     : None
  Chassis scoped: T0-SEL-B 1588-SEL
  Router scoped  : None
```

```

Uses frequency selection
S  Input                               Last Selection Point      QL  Pri  Status
==
1  GigabitEthernet0/0/0/2             n/a    PRC      1  Available
Selection point: LC_TX_SELECT (1 inputs, 1 selected)
Last programmed 3d23h ago, and selection made 3d23h ago
Next selection points
  SPA scoped      : None
  Node scoped     : None
  Chassis scoped: None
  Router scoped  : None
Uses frequency selection
Used for local line interface output
S  Input                               Last Selection Point      QL  Pri  Status
==
7  GigabitEthernet0/0/0/2             0/RP0/CPU0 T0-SEL-B 1    PRC  1  Available
Node 0/RP0/CPU0:
=====
Selection point: T0-SEL-B (3 inputs, 1 selected)
Last programmed 1d00h ago, and selection made 00:36:33 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  GigabitEthernet0/0/0/2 0/0/CPU0 ETH_RXMUX 1    PRC  1  Locked
   PTP [0/RP0/CPU0] n/a    SEC  254  Available
   Internal0 [0/RP0/CPU0] n/a    SEC  255  Available
Selection point: 1588-SEL (2 inputs, 1 selected)
Last programmed 3d23h ago, and selection made 00:36:33 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  GigabitEthernet0/0/0/2 0/0/CPU0 ETH_RXMUX 1    PRC  1  Locked
   Internal0 [0/RP0/CPU0] n/a    SEC  255  Available
Selection point: CHASSIS-TOD-SEL (2 inputs, 1 selected)
Last programmed 1d00h ago, and selection made 1d00h ago
Next selection points
  SPA scoped      : None
  Node scoped     : None
  Chassis scoped: None
  Router scoped  : None
PRC      1  Locked
SEC  255  Available
Last Selection Point
QL Pri Status
Uses time-of-day selection
S  Input                               Last Selection Point      Pri  Time  Status
==
1  PTP [0/RP0/CPU0] n/a    100  Yes   Available
   GigabitEthernet0/0/0/2 0/RP0/CPU0 T0-SEL-B 1    100  No    Available

```

Configuring PTP Telecom Profile Interface

This procedure describes the steps involved to create an interface for PTP ITU-T Telecom Profiles.



Note

- It is also possible to make these definitions within a global PTP profile and attach them to the interface using the profile command in PTP interface configuration mode.

1. To configure an interface, use **interface** *type interface-path-id* command in the configuration mode.

```
RP/0/RP0/CPU0:router(config)# interface gigabitethernet 0/1/0/1
```

2. To enter the PTP configuration mode for the given interface, use **ptp** command in the interface configuration mode.

```
RP/0/RP0/CPU0:router(config-if)# ptp
```

3. To configure a PTP profile (or specify a previously defined profile), use **profile** *name* command in the ptp-interface configuration mode.



Note

Any additional commands entered in ptp-interface configuration mode overrides the global profile settings.

```
RP/0/RP0/CPU0:router(config-if-ptp)# profile slave
```

4. To configure frequency for Sync or Delay-request messages for the given ptp interface, use **sync frequency** *rate* command or **delay-request frequency** *rate* command appropriately in the ptp-interface configuration mode. The valid configurable values are **2, 4, 8, 16, 32, 64 or 128**.

```
RP/0/RP0/CPU0:router(config-if-ptp)# sync frequency 128
```

```
RP/0/RP0/CPU0:router(config-if-ptp)# delay-request frequency 128
```

5. To configure duration for different PTP messages, use one of the following commands in the ptp-interface configuration mode: **announce grant-duration** *duration*, **sync grant-duration** *duration*, or **delay-response grant-duration** *duration*. The duration value can be between **60 and 1000 seconds**.



Note

This duration value represents the length of grant that is requested by a port in Slave state and represents the maximum grant-duration allowed when the port is in Master state.

```
RP/0/RP0/CPU0:router(config-if-ptp)# announce grant-duration 120
```

```
RP/0/RP0/CPU0:router(config-if-ptp)# sync grant-duration 120
```

```
RP/0/RP0/CPU0:router(config-if-ptp)# delay-response grant-duration 120
```


6. To configure a timeout value, length of time by when a PTP message must be received (before PTSF-lossSync is raised), use one of the following commands in the ptp-interface configuration mode: **sync timeout** *timeout* or **delay-response timeout** *timeout*. The timeout value can be between **100 to 10000 micro seconds**.

```
RP/0/RP0/CPU0:router(config-if-ptp)# sync timeout 120
```

```
RP/0/RP0/CPU0:router(config-if-ptp)# delay-response timeout 120
```

7. To configure a response for unicast-grant invalid-request, use **unicast-grant invalid-request {reduce | deny}** command. The response for requests with unacceptable parameters would either be denied or granted with reduced parameters.

```
RP/0/RP0/CPU0:router(config-if-ptp)# unicast-grant
invalid-request reduce
```

8. To configure IPv4 address for a PTP master, use **master ipv4 ip-address** command in the ptp-interface configuration mode.

```
RP/0/RP0/CPU0:router(config-if-ptp)# master ipv4 1.7.1.2
```

9. To override the clock-class received in Announce messages from the specified Master, use **clock-class class** command in the ptp-master-interface configuration mode. The class values can range from **0 to 255**.

```
RP/0/RP0/CPU0:router(config-if-ptp-master)# clock-class 2
```

Verification

To display the PTP interface details, use **show ptp interfaces brief** command.

```
RP/0/RP0/CPU0:router# show ptp interfaces brief
```

```
Fri Feb 9 11:16:45.248 UTC
```

Intf Name	Port Number	Port State	Encap	Line State	Mechanism
Gi0/1/0/0	1	Slave	IPv4	up	1-step DRRM
Gi0/0/0/40	2	Master	IPv4	up	1-step DRRM

To verify the configured profile details, use **show run interface interface-name** command.

```
RP/0/RP0/CPU0:router# show run interface Gi0/0/0/33
```

```
Wed Feb 28 11:49:16.940 UTC
```

```
interface GigabitEthernet0/0/0/33
```

```
ptp
```

```
profile slave
```

```
transport ipv4
```

```
sync frequency 64
```

```
clock operation one-step
```

```
delay-request frequency 64
```

```
!
```

```
physical-layer-frequency
```

```
!
```

```
ipv4 address 21.1.1.2 255.255.255.0
```

```
frequency synchronization
```

```
selection input
```

```
priority 5
```

```
wait-to-restore 0
!
```

Configuring PTP Telecom Profile Clock

This procedure describes the steps involved to configure PTP clock and its settings to be consistent with ITU-T Telecom Profiles for Frequency.

1. To enter the PTP configuration mode, use **ptp** command in the configuration mode.

```
RP/0/RP0/CPU0:router(config)# ptp
```

2. To enter the PTP-clock configuration mode, use **clock** command in the ptp-configuration mode.

```
RP/0/RP0/CPU0:router(config-ptp)# clock
```

3. To configure the domain-number for a PTP profile, use **domain number** command in the ptp-configuration mode. The allowed domain number range for G.8265.1 profile is between **4 and 23** and the range for G.8275.1 profile is between **24 and 43**.

```
RP/0/RP0/CPU0:router(config-ptp)# domain 24
```

4. To exit the ptp-clock configuration mode, use **exit** command.

```
RP/0/RP0/CPU0:router(config-ptp-clock)# exit
```

5. To configure the desired telecom profile and the clock type for the profile, use **clock profile {g.8275.1 | g.8275.2} clock-type {T-GM | T-BC | T-TSC}** command in the ptp configuration mode. For **g.8265.1** clock profile, clock type is either master or slave.



Note The **clock-selection telecom-profile** and **clock-advertisement telecom-profile** commands are deprecated from Release 6.1.2. They are replaced by the **clock profile** command.

```
RP/0/RP0/CPU0:router(config-ptp)# clock profile g.8275.1 clock-type T-GM
```

Verification

To display the configured PTP clock profile details, use **show run ptp** command.

```
RP/0/RP0/CPU0:router# show run ptp
ptp
clock
  domain 24
  profile g.8275.1 clock-type T-GM
  timescale PTP
  time-source GPS
  clock-class 6
!
profile master
  transport ethernet
  sync frequency 16
```

```

announce interval 1
delay-request frequency 16
!
profile master1
transport ethernet
sync frequency 64
announce interval 1
delay-request frequency 64
!

```

To verify that PTP has been enabled on the router and the device is in LOCKED Phase, use **show ptp platform servo** command.

```

RP/0/RP0/CPU0:router # show ptp platform servo

Fri Feb  9 11:16:54.568 UTC
Servo status: Running
Servo stat_index: 2
Device status: PHASE_LOCKED
Servo log level: 0
Phase Alignment Accuracy: 1 ns
Sync timestamp updated: 111157
Sync timestamp discarded: 0
Delay timestamp updated: 111157
Delay timestamp discarded: 0
Previous Received Timestamp T1: 1518155252.263409770  T2: 1518155252.263410517  T3:
1518155252.287008362  T4: 1518155252.287009110
Last Received Timestamp T1: 1518155252.325429435  T2: 1518155252.325430194  T3:
1518155252.348938058  T4: 1518155252.348938796
Offset from master:  0 secs, 11 nsecs
Mean path delay    :  0 secs, 748 nsecs
setTime():2  stepTime():1  adjustFreq():10413  adjustFreqTime():0
Last setTime: 1.000000000 flag:1  Last stepTime:-736216, Last adjustFreq:465

```

Configuring PTP Delay Asymmetry

Table 8: Feature History Table

Feature Name	Release Information	Description
PTP Delay Asymmetry	Release 7.3.1	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. The delay-symmetry command is introduced for this feature.

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

```

Configuration Examples

Slave Configuration Example

The following example shows a PTP slave configuration:

```
interface TenGigE 0/1/0/5
 ptp
  profile slave
  transport ipv4
  port state slave-only
  master ipv4 1.7.1.2
  !
  announce interval 1
  !
  ipv4 address 1.7.1.1 255.255.255.0
  !
```

Master Configuration Example

This example shows a PTP master configuration:

```
ptp
 profile master
 transport ipv4
 announce interval 1
 !
 ipv4 address 1.7.1.2 255.255.255.0
 !
```

PTP Hybrid Mode Configuration Example

This example shows the configuration of PTP hybrid mode:

```
ptp
 time-of-day priority 10
 !
 interface GigabitEthernet0/1/1/0
  ptp
   transport ipv4
   port state slave-only
   master ipv4 1.7.1.2
   !
   sync frequency 64
   announce interval 1
   delay-request frequency 64
   !
 interface GigabitEthernet 0/1/0/1
  ipv4 address 1.7.1.2 255.255.255.0
  speed 100
```

```
frequency synchronization
selection input
priority 10
wait-to-restore 0
ssm disable
time-of-day-priority 100
!
```

ITU-T Telecom Profile Examples:

G.8265.1 Profile Configuration Examples

Master Global Configuration:

```
ptp
clock
domain 4
profile g.8265.1
!
profile master
transport ipv4
sync frequency 16
announce interval 1
delay-request frequency 16
interface gi 0/2/0/4
ptp
profile master
transport ipv4
clock operation two-step
!
ipv4 address 17.1.1.1/24
```

Slave Global Configuration:

```
ptp
clock
domain 4
profile g.8265.1
!
profile slave
transport ipv4
sync frequency 16
announce interval 1
delay-request frequency 16
interface gi 0/1/0/0
ptp
profile slave
transport ipv4
Master ipv4 18.1.1.1
port state slave-only
!
clock operation two-step
!
ipv4 address 18.1.1.2/24
```

Configuring With Clock Type as T-Boundary Clock (T-BC)

```

ptp
 clock
 domain 4
 profile g.8265.1
 !
 profile master
 transport ipv4
 sync frequency 16
 announce interval 1
 delay-request frequency 16
 exit
 profile slave
 transport ipv4
 sync frequency 16
 announce interval 1
 delay-request frequency 16
 exit
interface gi 0/2/0/4
 ptp
  profile slave
  transport ipv4
  Master ipv4 17.1.1.1
  port state slave-only
  !
  clock operation two-step
  !
ipv4 address 17.1.1.2/24
interface gi 0/2/0/0
 ptp
  profile master
  transport ipv4
  clock operation two-step
  !
ipv4 address 18.1.1.1/24

```

G.8275.1 Profile Configuration Examples

Master Global Configuration:

```

ptp
 clock
 domain 24
 profile g.8275.1
 !
 profile master
 transport ethernet
 sync frequency 16
 announce frequency 8
 delay-request frequency 16
interface gi 0/2/0/4
 ptp
  profile master
  transport ethernet
  multicast target-address ethernet 01-1B-19-00-00-00
  !

```

Slave Global Configuration:

```

ptp

```



```
clock
domain 24
profile g.8275.1 clock-type T-TSC
!
profile slave
transport ethernet
sync frequency 16
announce frequency 8
delay-request frequency 16
interface gi 0/1/0/0
ptp
profile slave
transport ethernet
multicast target-address ethernet 01-1B-19-00-00-00
!
```

Configuring With Clock Type as T-Boundary Clock (T-BC)

```
ptp
clock
domain 24
profile g.8275.1 clock-type T-BC
!
profile master
transport ethernet
sync frequency 16
announce frequency 8
delay-request frequency 16
exit
profile slave
transport ethernet
sync frequency 16
announce frequency 8
delay-request frequency 16
exit
interface gi 0/2/0/4
ptp
profile slave
transport ethernet
multicast target-address ethernet 01-1B-19-00-00-00
!
interface gi 0/2/0/0
ptp
profile master
transport ethernet
multicast target-address ethernet 01-1B-19-00-00-00
```

G.8275.2 Profile Configuration Examples

Master Global Configuration:

```
ptp
clock
domain 44
profile g.8275.2 clock-type T-GM
!
profile master
transport ipv4
sync frequency 64
announce frequency 8
```

```

    unicast-grant invalid-request deny
    delay-request frequency 64
    !
    !

interface GigabitEthernet0/2/0/11
 ptp
  profile master
  !
  ipv4 address 17.1.1.1/24

```

Slave Global Configuration:

```

ptp
 clock
  domain 44
  profile g.8275.2 clock-type T-TSC
  !
 profile slave
  transport ipv4
  port state slave-only
  sync frequency 64
  announce frequency 8
  unicast-grant invalid-request deny
  delay-request frequency 64
  !
 log
  servo events
  best-master-clock changes
  !
  !
interface GigabitEthernet0/2/0/12
 ptp
  profile slave
  master ipv4 18.1.1.1
  !
  !
  ipv4 address 18.1.1.2/24
  !

```

Configuring With Clock Type as T-Boundary Clock (T-BC)

```

ptp
 clock
  domain 44
  profile g.8275.2 clock-type T-BC
  !
 profile slave
  transport ipv4
  port state slave-only
  sync frequency 64
  announce frequency 8
  unicast-grant invalid-request deny
  delay-request frequency 64
  !
 profile master
  transport ipv4
  sync frequency 64
  announce frequency 8
  unicast-grant invalid-request deny
  delay-request frequency 64
  !

```

```

log
  servo events
  best-master-clock changes
!
!

interface GigabitEthernet0/2/0/11
  ptp
  profile master
  !
  ipv4 address 18.1.1.1/24
  !

interface GigabitEthernet0/2/0/12
  ptp
  profile slave
  master ipv4 17.1.1.1
  !
  !
  ipv4 address 17.1.1.2/24
  !

```

Configure E-SyncE on Primary and Secondary Interface

Primary Interface

The following example shows how you can configure global sync on primary interface:

```

Router#configure terminal
Router(config)#frequency synchronization
Router(config-freqsync)#quality itu-t option 1
Router(config-freqsync)#clock-identity mac-address aaaa.bbbb.cccc
Router(config-freqsync)#clock-interface timing-mode system
Router(config-freqsync)#commit

```

The following example shows how you can configure sync on primary interface:

```

Router#configure terminal
Router(config)# interface HundredGigE0/0/0/11
Router(config-if)# frequency synchronization
Router(config-if)# quality transmit exact itu-t option 1 ePRTC
Router(config-if)# commit

```

Secondary Interface

The following example shows how you can configure global sync on secondary interface:

```

Router#configure terminal
Router(config)#frequency synchronization
Router(config-freqsync)#quality itu-t option 1
Router(config-freqsync)#clock-interface timing-mode system
Router(config-freqsync)#commit

```

The following example shows how you can configure sync on secondary interface:

```

Router#configure terminal
Router(config)# interface HundredGigE0/0/0/10
Router(config-if)# frequency synchronization
Router(config-if-freqsync)# selection input
Router(config-if-freqsync)# priority 10
Router(config-if-freqsync)# wait-to-restore 0
Router(config-if-freqsync)# commit

```



Note If timing mode system is not configured, the major alarm T4 PLL is in FREERUN mode is raised. This alarm has no functional impact to the system behavior.

Verification

Use the **show frequency synchronization** command if e-synce is configured.

```

Routerr#show frequency synchronization interfaces br
Flags: > - Up           D - Down           S - Assigned for selection
        d - SSM Disabled x - Peer timed out i - Init state
        s - Output squelched

Fl  Interface           Qlrcv Qluse Pri Qlsnd Output driven by
====
>S  HundredGigE0/0/0/13  ePRTC ePRTC 31 ePRTC HundredGigE0/0/0/18
>S  HundredGigE0/0/0/18  ePRTC ePRTC 30 DNU  HundredGigE0/0/0/18
RP/0/RP0/CPU0:Shadowtower#sh frequency synchronization selection
Node 0/RP0/CPU0:
=====
Selection point: T0-SEL-B (3 inputs, 1 selected)
Last programmed 02:41:55 ago, and selection made 02:41:04 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
Used for local clock interface output
S  Input           Last Selection Point           QL Pri Status
==
33 HundredGigE0/0/0/18  0/RP0/CPU0 ETH_RXMUX 33  ePRTC 30 Locked
   HundredGigE0/0/0/13  0/RP0/CPU0 ETH_RXMUX 22  ePRTC 31 Available
   Internal0 [0/RP0/CPU0] n/a                          SEC 255 Available

Selection point: 1588-SEL (3 inputs, 1 selected)
Last programmed 02:41:55 ago, and selection made 02:41:04 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  Internal0 [0/RP0/CPU0] n/a                          SEC 255 Freerun
   HundredGigE0/0/0/18  0/RP0/CPU0 ETH_RXMUX 33  ePRTC 30 Available
   HundredGigE0/0/0/13  0/RP0/CPU0 ETH_RXMUX 22  ePRTC 31 Available

Selection point: CHASSIS-TOD-SEL (1 inputs, 1 selected)
Last programmed 02:41:44 ago, and selection made 02:41:44 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  HundredGigE0/0/0/18  0/RP0/CPU0 T0-SEL-B 33    100 No  Available

```

```
Selection point: ETH_RXMUX (2 inputs, 2 selected)
Last programmed 02:41:55 ago, and selection made 02:41:55 ago
Next selection points
  SPA scoped      : None
  Node scoped     : T0-SEL-B 1588-SEL
  Chassis scoped  : None
  Router scoped   : None
Uses frequency selection
```

S	Input	Last Selection Point	QL	Pri	Status
33	HundredGigE0/0/0/18	n/a	ePRTC	30	Available
22	HundredGigE0/0/0/13	n/a	ePRTC	31	Available



CHAPTER 12

Configuring Smart Licensing

This module describes the configuration related to the Smart Licensing.

Table 9: Feature History for Smart License

Release	Modification
Release 6.3.2	Smart Licensing was introduced.
Release 6.5.2	Flexible Consumption License Model was introduced.
Release 7.4.1	YANG Data Models for Smart Licensing was introduced.

This module contains the following topics:

- [What is Smart Licensing?, on page 181](#)
- [What is Flexible Consumption Model?, on page 182](#)
- [How Does Smart Licensing Work?, on page 188](#)
- [What is Cisco Smart Software Manager?, on page 189](#)
- [Configuring Smart Licensing, on page 191](#)
- [Registering and Activating Your Router, on page 198](#)
- [Verifying the Smart Licensing Configuration , on page 203](#)
- [Specific License Reservation, on page 210](#)
- [Configuring Specific License Reservation, on page 211](#)
- [To cancel, verify, remove, and return SLR, on page 213](#)
- [Configuration Examples for SLR, on page 215](#)

What is Smart Licensing?

Smart Licensing is a cloud-based, flexible software licensing model that enables you to activate and manage Cisco software licenses across their organization. Smart Licensing solution allows you to easily track the status of your license and software usage trends. Cisco Smart Licensing establishes a pool of licenses or entitlements that can be used across the entire organization in a flexible and automated manner. Smart Licensing helps simplify four core functions:

- **Purchase**—Creates a Smart Account (and optionally, your Virtual Account). Licenses are added to your Smart Account and are immediately available for use.

- **Install**—Register your product with your Smart Account using an account-based Registration Token. Thereafter, the entire process is automatic. Product Activation Keys (PAKs) and license files are no longer needed.
- **Management**—Make changes to license consumption by updating your configuration; any license change is automatically reflected in your Smart Account. You can share licenses in your Virtual Account through the license pooling option. License pools (logical grouping of licenses) can reflect your organization structure. Smart Licensing solution also offers Cisco Smart Software Manager, a centralized portal that enables you to manage all your Cisco software licenses from one centralized website.
- **Visibility and Asset Management**—Cisco Smart Software Manager (CSSM) portal offers an integrated view of the licenses you own and have deployed. You can use this data to make better purchase decisions, based on your consumption.

What is Flexible Consumption Model?

Table 10: Feature History Table

Feature Name	Release Information	Feature Description
Flexible Consumption Model on NC57-48Q2D-S and NC57-48Q2D-SE-S	Release 7.10.1	Flexible Consumption Model on NC57-48Q2D-S and NC57-48Q2D-SE-S Line cards.
Support for Flexible Consumption Model (FCM) licensing on the chassis NCS-57B1-6D24-SYS and NCS-57B1-5DSE-SYS.	Release 7.3.1	Support for FCM licensing is now extended to the following chassis: <ul style="list-style-type: none"> • NCS-57B1-6D24-SYS • NCS-57B1-5DSE-SYS

The Flexible Consumption Model (FCM) provides the capability and flexibility to purchase software capacity as needed. FCM delivers the following:

- **Pay-as-you-grow**—Enables you to lower initial costs and add more capacity over time.
- **Simplify operations**—FCM delivers the carrier-class IOS-XR software feature set with two software suites, Essentials and Advantage, that simplifies license management.
- **Utilize capital efficiently**—License pooling enables an efficient way to share licenses across the network.

To enable Flexible Consumption model licensing on routers running Cisco IOS XR :

```
Router(config)# license smart flexible-consumption enable
Router(config)# commit
```

To verify the Flexible Consumption Model configuration:

```
Device# show running-config license smart flexible-consumption enable
```

The following tables provide information about FCM licenses for NCS 5500 and NCS 5700 Series routers:

Table 11: Flexible Consumption Model Licensing Usage Pattern

License Name	Hardware Supported	Consumption Pattern
<p>Essential Licenses:</p> <ul style="list-style-type: none"> • ESS-100G-RTU-1 • ESS-400G-RTU-1 <p>Advantage Licenses:</p> <ul style="list-style-type: none"> • ADV-100G-RTU-1 • ADV-400G-RTU-1 	<ul style="list-style-type: none"> • Routers with fixed chassis unit: NCS-5501-SYS, NCS-5501-SE-SYS, NCS-5502-SYS, NCS-5502-SE-SYS, NCS-55A1-24H-SYS, NCS-55A1-36H-S-SYS, NCS-55A2-MOD-HD-S-SYS, NCS-55A2-MOD-SE-S-SYS, NCS-55A2-MOD-S-SYS, NCS-55A1-36H-SE-S-SYS, NCS-55A1-48Q6H-SYS, NCS-55A1-24Q6H-SS-SYS, NCS-57B1-6D24-SYS, and NCS-57B1-5DSE-SYS • Routers with modular chassis unit: NCS-5504-SYS, NCS-5508-SYS, NCS-5516-SYS • Line cards: NC-57-18DD-SE, NC-57-24DD, NC-55-32T16Q4H-A, NC-57-36H-SE, NC-55-36X100G, NC-55-36X100GA-SE, NC-55-18H18F, NC-55-24H12F-SE, NC-55-24X100G-SE, NC-55-6X200-DWDM-S, NC-55-MOD-A-S, NC-55-MOD-A-SE-S, and NC-55-36X100G-S 	<p>The number of essential or advantage licenses that are consumed depends on the number of active ports and is reported on per chassis basis.</p>

License Name	Hardware Supported	Consumption Pattern
<p>Hardware Tracking Licenses that support chassis:</p> <ul style="list-style-type: none"> • NCS-5501-TRK • NCS-5501-SE-TRK • NCS-5502-TRK • NCS-5502-SE-TRK • NCS-5504-TRK • NCS-5508-TRK • NCS-5516-TRK • NCS-55A1-24H-TRK • NCS-55A1-36H-TRK • NCS-55A1-36HS-TRK • NCS-55A1-48Q6H-TRK • NCS-55A2-MOD-TRK • NCS-55A2-MODH-TRK • NCS-55A2-MODS-TRK • NCS-55A1-24Q6-TRK • NCS-55A1-24QX-TRK • NC55-32T16Q-LC-TRK • NCS-57B1-6D24-SYS • NCS-57B1-5DSE-SYS 	<p>These Tracking licenses are named on the basis of the hardware supported. For example, NCS-5501-TRK licenses support NCS 5501 systems.</p>	<p>The number of licenses that are consumed depends on the number of chassis in use.</p>

License Name	Hardware Supported	Consumption Pattern
Hardware Tracking Licenses that support line cards: <ul style="list-style-type: none"> • NC55-36H-LC-TRK • NC55-36HSE-LC-TRK • NC55-18HF-LC-TRK • NC55-24H12-LC-TRK • NC55-24HSE-LC-TRK • NC55-DWDM-LC-TRK • NC55-MOD-A-SE-TRK • NC55-MOD-A-TRK • NC55-36HS-LC-TRK • NC55-18D12THSE-LC-TRK • NC55-24D-LC-TRK 	These Tracking licenses are named on the basis of the line card supported. For example, NC55-36H-LC-TRK licenses support NC-55-36X100G line cards.	The number of licenses that are consumed depends on the number of line cards in use.

Software Innovation Access

Table 12: Feature History Table

	Release Information	Feature Description
Software Innovation Access (SIA) Entitlement	Release 7.3.1	SIA license grants you access to the latest software upgrades which contain new features, bug fixes, and security enhancements for devices on your network. Also, it enables the consumption of Advantage and Essential Right-to-Use (RTU) licenses on your device, and allows portability of these RTU licenses from one device to another.

Overview

Software Innovation Access (SIA) subscription, a type of FCM licensing, provides access to the latest software upgrades and features for your network. SIA licenses enable the consumption of Right-to-Use (RTU) licenses for your devices to access software innovation and avail support for your devices throughout the term of subscription.

The benefits of SIA subscription are:

- **Access to software innovation:** SIA subscription provides access to continuous software upgrades which contain latest features, security enhancements, and bug fixes for all your devices at a network level.

- **Pooling of licenses:** SIA subscription enables Right-to-Use (RTU) licenses to be shared across your FCM network from a common license pool through the virtual account.
- **Protects your investment:** SIA subscription enables the portability of perpetual RTU licenses purchased for your current device to a next-generation router when you expand or upgrade your network.

The initial term of a SIA subscription is for a term of three years. You can renew the subscription by contacting your Cisco account representative. An equal number of SIA licenses and corresponding RTU licenses are required to enjoy the benefits, and ensure that your network is in compliance. There are two types of SIA licenses available:

- : To utilize Advantage RTU licenses, you need Advantage SIA licenses.
- : Essential SIA licenses are required to utilise Essential RTU on your device.

If your device is in a state of SIA Out-of-Compliance (OOC) the benefits cease.

SIA Out-of-compliance (OOC) state

When your device is in a SIA Out-of-compliance state, support for major software version upgrades in your network devices are restricted. However, you can continue to perform minor updates, SMU installations, and RPM installations, and continue to utilize the RTU licenses without support for porting.

A device can get into a SIA Out-of-compliance (OOC) state in the following instances:

- The SIA License *EVAL* period of 90 days has expired.
- The number of SIA licenses consumed has exceeded the number of SIA licenses purchased. This can also occur when the RTU licenses consumed is higher than the number of SIA licenses purchased.
- The term of the SIA license has expired and you haven't renewed the subscription.
- The license authorization status is:
 - Not Authorized: The license authorization code installed doesn't contain sufficient counts for the request. This can occur when you attempt to use more licenses than the licenses available in your Virtual Account.
 - Authorization expired: The device hasn't been able to connect to CSSM for an extended period, due to which the authorization status couldn't be verified.



Note The CSSM smart license hierarchy applies to the Right-to-Use (RTU) license only. Therefore, if there is an insufficient RTU 100G license, CSSM can convert the RTU 400G license into four RTU 100G licenses. This is not applicable for SIA license.

In order to bring your device to In-Compliance state, perform one of the following steps:

- Register your device with CSSM if the SIA license *EVAL* period has expired.
- If the SIA license has expired or the number of SIA licenses consumed is more than the number of SIA licenses purchased, contact your Cisco Account Representative to purchase or renew the required licenses.
- If the authorization code has insufficient counts for the request, generate the code with sufficient counts.
- If the authorization has expired, connect the device with CSSM.

When the device enters an OOC state, a grace period of 90 days (cumulative of all the previous occurrences) begins. During this period, SIA license benefits can still be availed. The system attempts to renew the authorization period by connecting with the CSSM during the grace period, or even after the grace period has expired. If an attempt isn't successful, it remains in an OOC state. If the attempt is successful, a new authorization period begins and the device is In-Compliance.

Verification

To verify the device compliance status, use the **show license platform summary** command:

Examples

Status: In-Compliance

```
Router# show license platform summary
Fri Feb 12 20:33:51.784 UTC
Collection: LAST: Fri Feb 12 2021 20:33:46 UTC
NEXT: Fri Feb 12 2021 20:35:46 UTC
Reporting: LAST: Fri Feb 12 2021 20:33:46 UTC
NEXT: Fri Feb 12 2021 20:35:46 UTC
SIA Status: In Compliance
Count
Feature/Area Entitlement Last Next
=====
FCM NCS 5500 Core & Agg Netw SW&Autom Ess(100G) 1.0 Per 1 0
FCM Core & Aggr Essentials SIA per 100G 1 0
FCM 5501 SE Base Hardware Tracking PID 1 0
```

Status: Out-of-Compliance (Grace Period Remaining)

```
Router# show license platform summary
Mon Mar 30 04:01:50.405 IST
Collection: LAST: Mon Mar 30 2020 04:01:12 IS
NEXT: Mon Mar 30 2020 04:02:12 IST
Reporting: LAST: Mon Mar 30 2020 04:01:12 IST
NEXT: Mon Mar 30 2020 04:03:12 IST
*****IMPORTANT*****
SIA Status: Out of Compliance(Remaining Grace Period: 89 days, 23 hours)
Number of SIA license(s) used is more than available.
SW Upgrade will be allowed as SIA Grace Period is remaining
*****
Feature/Area Entitlement Count
Last Next
=====
FCM NCS 5500 Core & Agg Netw SW&Autom Ess(100G) 1.0 2 0
FCM Essential SIA License 2 0
FCM Total NC-55-18H18F LCs in the system 1 0
FCM 5508 Base Hardware Tracking PID 1 0
```

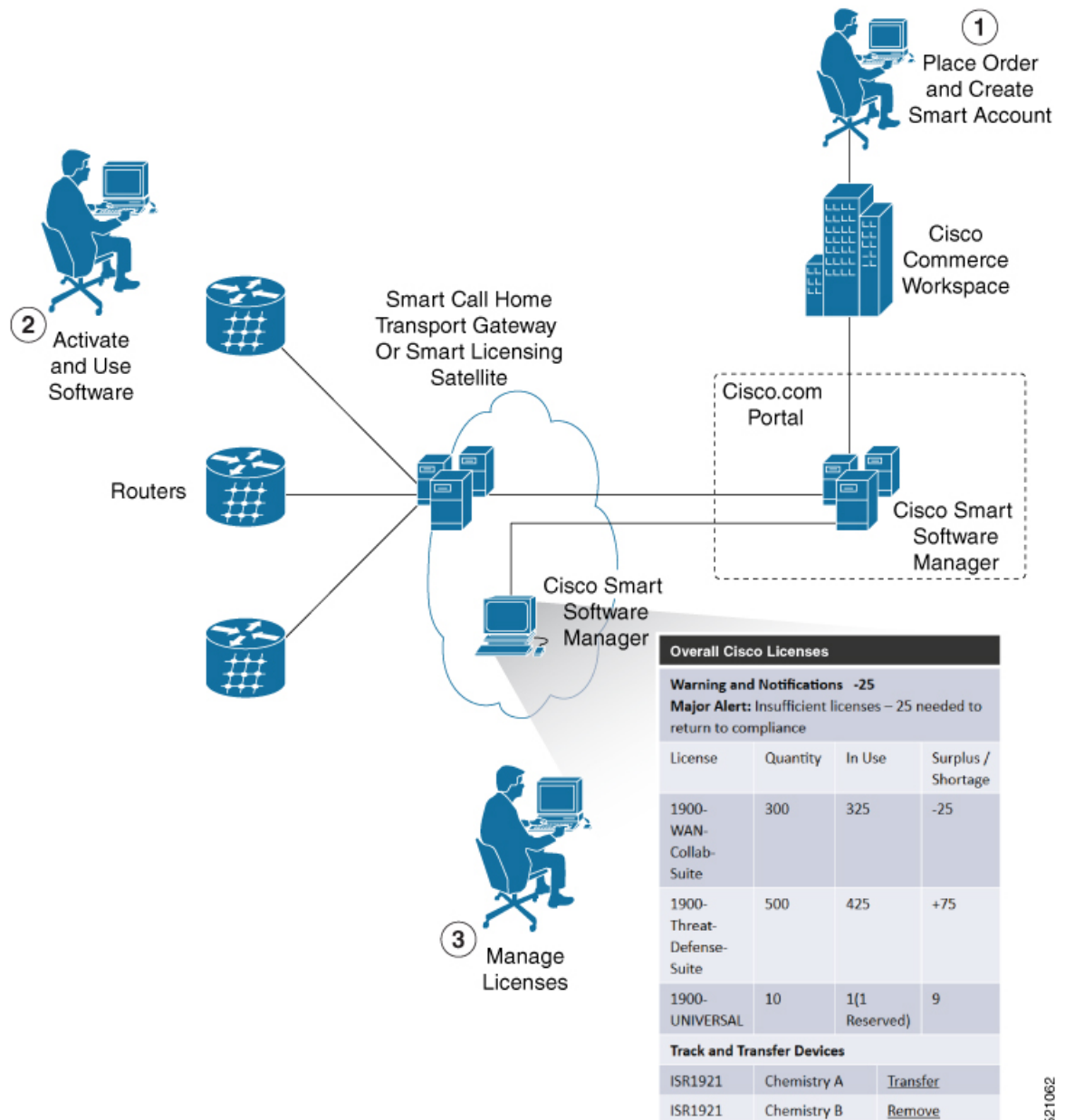
Status: Out-of-Compliance (Grace Period expired and upgrades are blocked)

```
Router# show license platform summary
Mon Mar 30 04:41:02.036 IST
Collection: LAST: Mon Mar 30 2020 04:40:12 IST
NEXT: Mon Mar 30 2020 04:41:12 IST
Reporting: LAST: Mon Mar 30 2020 04:39:12 IST
NEXT: Mon Mar 30 2020 04:41:12 IST
*****IMPORTANT*****
SIA Status: Out of Compliance(Grace Period Expired)
SW Upgrades are blocked as SIA license(s) are in Not Authorized state
*****
Count
```

Feature/Area	Entitlement	Last	Next
FCM	NCS 5500 Core & Agg Netw SW&Autom Ess(100G) 1.0	2	2
FCM	Essential SIA License	2	2

How Does Smart Licensing Work?

Figure 5: Smart Licensing - Workflow



521062

1. Place Order and Create Smart Account—You must have a Smart Account to set up Smart Licensing.
 - a. Go to <https://software.cisco.com/>.

- b. Under the **Administration** section, click **Get a Smart Account or Request Access to an Existing Smart Account**.
 - c. Verify or enter your Cisco.com profile details to complete creating a Smart Account.
2. Activate and Use Software—Register your product. For more information, see the *Registering your Router* section. You can use either of the following options to communicate with the CSSM:
 - Smart Call Home—The Smart Call Home feature is automatically configured. Smart Call Home is used by Smart Licensing as a medium for communication with the CSSM. You can use this feature to page a network support engineer, email a Network Operations Center, or 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. For more information on Smart Call Home feature, see the [Smart Call Home Deployment Guide](#).
 - Smart Licensing CSSM On-Prem—The Smart licensing on-premise option provides an on-premises collector that can be used to consolidate and manage Smart license usage, as well as facilitate communications back to the CSSM at Cisco.com.
3. Manage Licenses—You can manage and view reports about your overall license usage in the Smart Software Manager portal.

What is Cisco Smart Software Manager?

Cisco Smart Software Manager enables you to manage all of your Cisco Smart software licenses from one centralized website. With Cisco Smart Software Manager, you organize and view your licenses in groups called virtual accounts (collections of licenses and product instances). The Cisco Smart Software Manager allows you to:

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

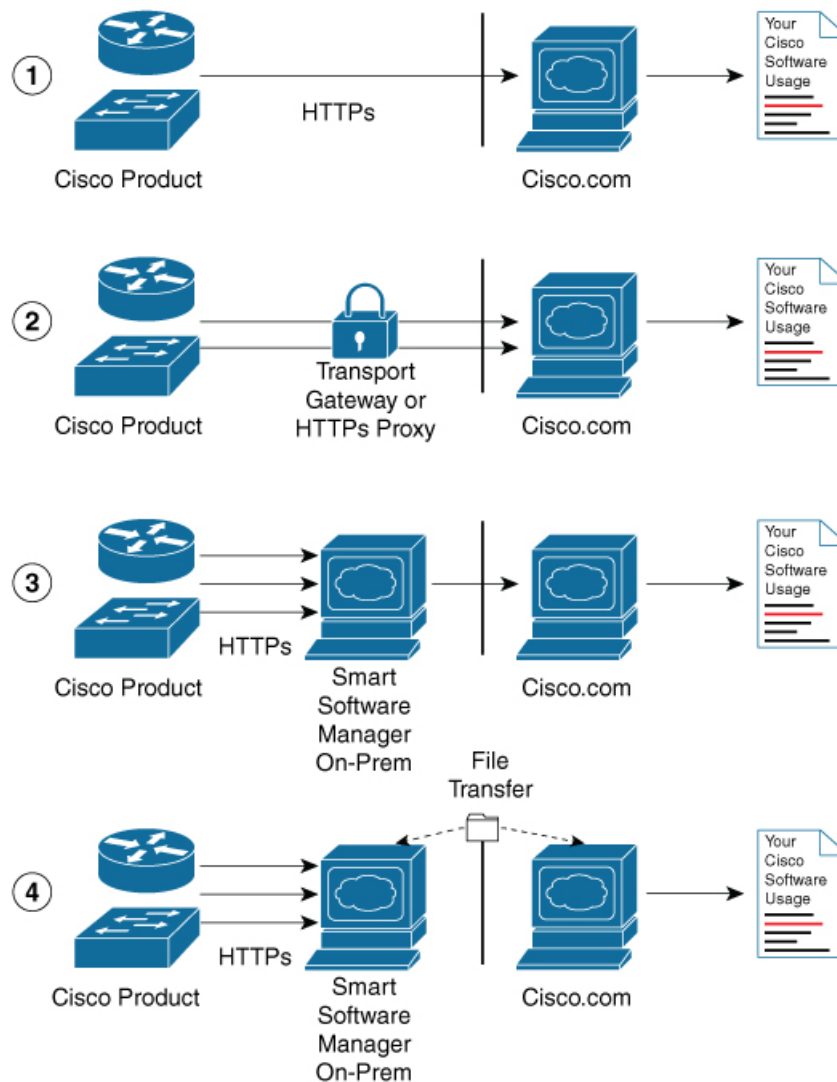
To access the Cisco Smart Software Manager:

- Go to <https://software.cisco.com>.
- Under the **License** section, click **Smart Software Licensing**.

Smart Licensing Deployment Options

The following illustration shows the various options available for deploying Smart Licensing:

Figure 6: Smart Licensing Deployment Options



1. Direct cloud access—In this method, Cisco products send usage information directly over the internet to CSSM on <http://www.cisco.com>; no additional components are needed for deployment.
2. Direct cloud access through an HTTPs proxy—In direct cloud access through an HTTPs proxy deployment method, Cisco products send usage information over the internet through a proxy server—either a Smart Call Home Transport Gateway or off-the-shelf Proxy (such as Apache) to CSSM on <http://www.cisco.com>.
3. Mediated access through an on-premises collector-connected—In mediated access through an on-premises collector-connected deployment method, Cisco products send usage information to a locally connected collector, which acts as a local license authority. Periodically, the information is exchanged to keep the databases in synchronization.
4. Mediated access through an on-premises collector-disconnected—In the mediated access through an on-premises collector-disconnected deployment method, Cisco products send usage information to a local disconnected collector, which acts as a local license authority. Exchange of human-readable information is performed occasionally (once a month) to keep the databases in synchronization.

Options 1 and 2 provide easy deployment options, whereas options 3 and 4 provide secure environment deployment options.



Note Smart Software On-Premise provides support for options 3 and 4.

The communication between Cisco devices and CSSM is facilitated by the Smart Call Home software.

Configuring Smart Licensing

Prerequisites for Configuring Smart Licensing

Ensure that you have completed the following activities on Cisco Smart Software Manager:

- Set up a Cisco Smart Account. For more information, see the *How Smart Licensing Works* section in this document.
- Set up Virtual Account or accounts. For more information, see the *Virtual Accounts* section in the [Smart Software Manager Help](#).
- Create user roles in the **Users** tab in the **Manage Smart Account** page. Provide the appropriate user access rights.
- Accept the Smart Software Licensing Agreement on Cisco Smart Software Manager to register your router.
- Have a layer 3 connection set up on your router.
- Configure a valid DNS and proper time on the router to connect CSSM or CSSM On-Prem.

Setting up the Router for Smart Licensing

Table 13: Three-step Roadmap to Set up the Router for Smart Licensing

Activity	Communication Connection Options		
Step 1—Configure Communications	See the <i>Configuring a Direct Cloud Connection</i> section.	See the <i>Configuring a Connection through a HTTP Proxy</i> section.	See the <i>Connecting to CSSM On-Premise</i> section.
Step 2—Register and Activate	See the <i>Registering and Activating your Router</i> section.		
Step 3—Verify the Configuration	See the <i>Verifying your Smart Licensing Configuration</i> section.		

Configuring a Communications Connection Between the Router and Cisco Smart Software Manager

Configuring a Direct Cloud Connection

In this deployment option, the **configure call-home profile** is configured by default. Use the **show call-home profile all** command to check the profile status.

Call Home service provides email-based and web-based notification of critical system events to Cisco Smart Software Manager.

To configure and enable Call Home service:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. **service active**
4. **contact-email-addr** *email-address*
5. **profile CiscoTAC-1**
6. **destination transport-method** **http**
7. **destination address** **http** *url*
8. **active**
9. **no destination transport-method** **email**
10. **commit**
11. **exit**
12. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 2	call-home Example: Router(config)# call-home	Enters Call Home configuration mode.
Step 3	service active Example: Router(config-call-home)# service active	Activates Call Home service.
Step 4	contact-email-addr <i>email-address</i> Example: Router(config-call-home)# contact-email-addr username@example.com	Assigns the provided email address. You can enter up to 200 characters in email address format. Note Spaces are not allowed in the email address.

	Command or Action	Purpose
Step 5	profile CiscoTAC-1 Example: <pre>Router(config-call-home)# profile CiscoTAC-1</pre>	Enables the CiscoTAC-1 profile to be used with the Call Home service. By default, the CiscoTAC-1 profile is disabled.
Step 6	destination transport-method http Example: <pre>Router(config-call-home-profile)# destination transport-method http</pre>	Enables the Call Home service through an HTTP connection.
Step 7	destination address http url Example: <pre>Router(config-call-home-profile)# destination address http https://tools.cisco.com/its/service/odde/services/DDCEService</pre>	Connects the router to the Cisco Smart Software Manager.
Step 8	active Example: <pre>Router(config-call-home-profile)# active</pre>	Enables the destination profile.
Step 9	no destination transport-method email Example: <pre>Router(config-call-home-profile)# no destination transport-method email</pre>	Disables the email option for the Call Home service.
Step 10	commit Example: <pre>Router(config-call-home-profile)# commit</pre>	Commits the configuration.
Step 11	exit Example: <pre>Router(config-call-home-profile)# exit</pre>	Exits the Call Home destination profile configuration mode and returns to the Call Home configuration mode.
Step 12	exit Example: <pre>Router(config-call-home)# exit Router(config)#</pre>	Exits the Call Home configuration mode and returns to the global configuration mode.

Configuring a Connection Through an HTTP Proxy

The Call Home service can be configured through an HTTPs proxy server.

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. **service active**

4. **contact-email-address** *email-address*
5. **http-proxy** *proxy-address* **port** *port-number*
6. **profile** **CiscoTAC-1**
7. **no destination transport-method** **email**
8. **exit**
9. **profile** *profile-name*
10. **reporting smart-licensing-data**
11. **destination transport-method** **http**
12. **destination address** **http** *url*
13. **active**
14. **exit**
15. **exit**
16. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 2	call-home Example: Router(config)# call-home	Enters Call Home configuration mode.
Step 3	service active Example: Router(config-call-home)# service active	Enables the Call Home feature.
Step 4	contact-email-address <i>email-address</i> Example: Router(config-call-home)# contact-email-addr sch-smart-licensing@cisco.com	Configures the default email address.
Step 5	http-proxy <i>proxy-address</i> port <i>port-number</i> Example: Router(config-call-home)# http-proxy 198.51.100.10 port 3128	Provides the proxy server information to the Call Home service.
Step 6	profile CiscoTAC-1 Example: Router(config-call-home)# profile CiscoTAC-1	Enables the CiscoTAC-1 profile to be used with the Call Home service. By default, the CiscoTAC-1 profile is disabled.
Step 7	no destination transport-method email Example:	Disables the email option for the Call Home service.

	Command or Action	Purpose
	<code>Router(config-call-home-profile)# no destination transport-method email</code>	
Step 8	exit Example: <code>Router(config-call-home-profile)# exit</code> <code>Router(config-call-home)#</code>	Exits the Call Home destination profile configuration mode and returns to the Call Home configuration mode.
Step 9	profile <i>profile-name</i> Example: <code>Router(config-call-home)# profile test1</code>	Enters the Call Home destination profile configuration mode for the specified destination profile name. If the specified destination profile does not exist, it is created.
Step 10	reporting smart-licensing-data Example: <code>Router(config-call-home-profile)# reporting smart-licensing-data</code>	Enables data sharing with the Call Home service through the configured transport method, in this case, HTTP.
Step 11	destination transport-method http Example: <code>Router(config-call-home-profile)# destination transport-method http</code>	Enables the HTTP message transport method.
Step 12	destination address http <i>url</i> Example: <code>Router(config-call-home-profile)# destination address http https://tools.cisco.com/its/service/odce/services/DDCEService</code>	Connects the router to the Cisco Smart Software Manager.
Step 13	active Example: <code>Router(config-call-home-profile)# active</code>	Enables the destination profile.
Step 14	exit Example: <code>Router(config-call-home-profile)# exit</code>	Exits the Call Home destination profile configuration mode and returns to the Call Home configuration mode.
Step 15	exit Example: <code>Router(config-call-home)# exit</code> <code>Router(config)#</code>	Exits the Call Home configuration mode and returns to the global configuration mode.
Step 16	commit Example: <code>Router(config)# commit</code>	Commits the configuration.

Connecting to CSSM On-Premise

This section describes how to configure the Call Home service for on-premise smart software through connected or disconnected mode.

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. **profile** *profile-name*
4. **reporting smart-licensing-data**
5. **destination transport-method** **http**
6. **destination address** **http** *url*
7. **no destination address** **http** *url*
8. **destination preferred-msg-format** {**long-text** | **short-text** | **xml**}
9. **active**
10. **exit**
11. **exit**
12. **http client source-interface** *ip-version interface-type interface-number*
13. **crypto ca trustpool policy** *crl optional*
14. **commit**
15. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 2	call-home Example: Router(config)# call-home	Enters Call Home configuration mode.
Step 3	profile <i>profile-name</i> Example: Router(config-call-home)# profile test1	Enters the 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	reporting smart-licensing-data Example: Router(config-call-home-profile)# reporting smart-licensing-data	Enables data sharing with the Call Home service through the configured transport method, in this case, HTTP.
Step 5	destination transport-method http Example: Router(config-call-home-profile)# destination transport-method http	Enables the HTTP message transport method.

	Command or Action	Purpose
Step 6	destination address http url Example: <pre>Router(config-call-home-profile)# destination address http http://209.165.201.15/Transportgateway/services/DeviceRequestHandler</pre> Or <pre>Router(config-call-home-profile)# destination address http https://209.165.201.15/Transportgateway/services/DeviceRequestHandler</pre>	Configures the destination URL (CSSM) to which Call Home messages are sent. Note Ensure the IP address or the fully qualified domain name (FQDN) in the destination URL matches the IP address or the FQDN as configured for the Host Name on the CSSM On-Prem.
Step 7	no destination address http url Example: <pre>Router(config-call-home-profile)# no destination address http https://tools.cisco.com/its/service/odbc/services/DDCEService</pre>	Removes the default destination address.
Step 8	destination preferred-msg-format {long-text short-text xml} Example: <pre>Router(config-call-home-profile)# destination preferred-msg-format xml</pre>	(Optional) Configures a preferred message format. The default message format is XML.
Step 9	active Example: <pre>Router(config-call-home-profile)# active</pre>	Enables the destination profile.
Step 10	exit Example: <pre>Router(config-call-home-profile)# exit</pre>	Exits the Call Home destination profile configuration mode and returns to the Call Home configuration mode.
Step 11	exit Example: <pre>Router(config-call-home)# exit Router(config)#</pre>	Exits the Call Home configuration mode and returns to the global configuration mode.
Step 12	http client source-interface ip-version interface-type interface-number Example: <pre>Router(config)# http client source-interface ipv4 Vlan100</pre>	Configures a source interface for the HTTP client. Note This command is mandatory for a VRF interface.
Step 13	crypto ca trustpool policy crl optional Example:	(Optional) Bypasses the Certificate Revocation Lists (CRLs) check and establishes the connection. By default, the CRLs check is mandatory while establishing a TLS

	Command or Action	Purpose
	<code>Router(config)# crypto ca trustpool policy crl optional</code>	connection. We recommend this step when the smart licensing-enabled router is within a network and can rely on the License server to check the certificate status without retrieving and caching each CRL for every peer.
Step 14	commit Example: <code>Router(config)# commit</code>	Commits the configuration.
Step 15	end Example: <code>Router(config)# end</code>	Returns to the global configuration mode.

Installing CSSM On-Premise

For information on installation instructions, see the [Smart Software Manager On-Prem Installation Guide](#).

Registering and Activating Your Router

Product registration securely associates a device with the Smart Account and the Virtual Account of your choice. It also establishes trust between the end product and the CSSM. Tokens are used to register a product with the appropriate Virtual Account on CSSM Cloud (on Cisco.com) or CSSM On-Premise.



Note When the router is in an unregistered state, the licenses are in EVAL (evaluation) mode. Evaluation period will last for 90 days.

A Registration Token:

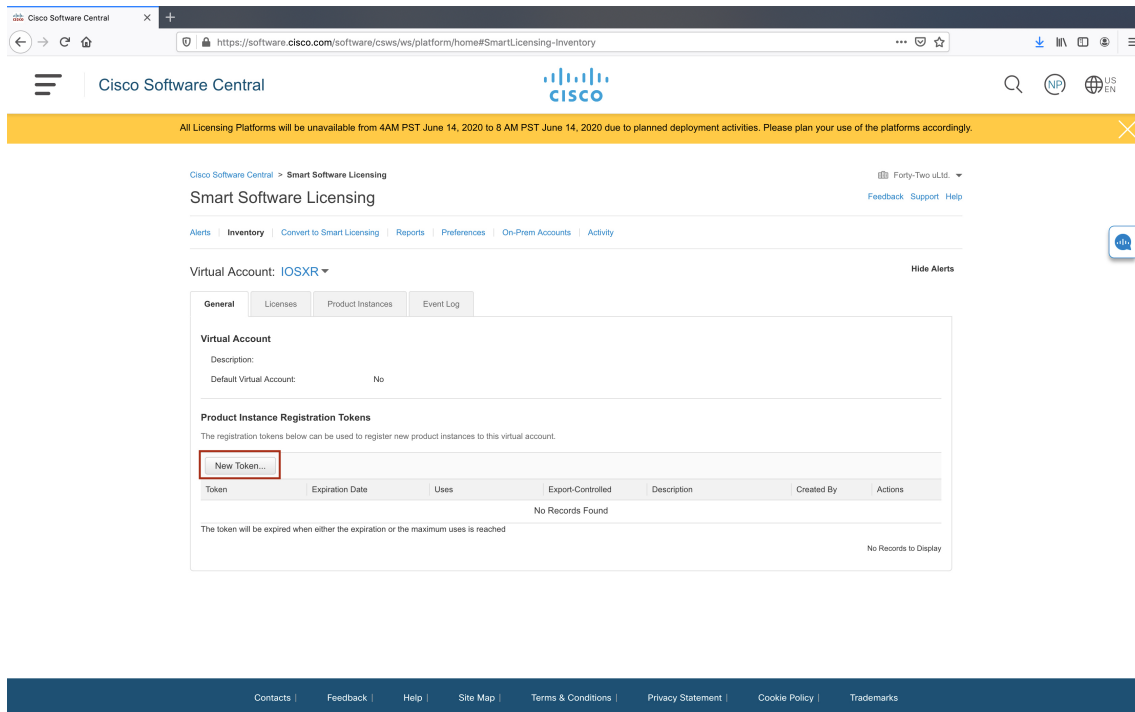
- Can be either used once or reused multiple times. You can set a limit to the number of times a token can be reused when you create the token.
- Can be created and revoked at any time.
- Expires after a period of time (default is 30 days; minimum is one day; maximum is 365 days)

A Registration Token is not:

- Product specific: The same Registration Token can be used on different product types.
- A license, key, or PAK.
- Stored on the Cisco device and they are not persistent.
- Required after the product is registered. Token expiration has no effect on previously registered products; it simply means that that token can no longer be used to register a new product.

Generating a New Token from CSSM

- Step 1** If you choose the direct cloud access deployment option, log in to CSSM from <https://software.cisco.com/#>.
If you chose the mediated access deployment option, log in to CSSM On-Prem from <https://<on-prem-ip-address>:8443>.
- Step 2** Select the **Inventory** tab.
- Step 3** From the Virtual Account drop-down list, choose the virtual account to which you want to register your product.
- Step 4** Select the **General** tab.
- Step 5** Click **New Token**.

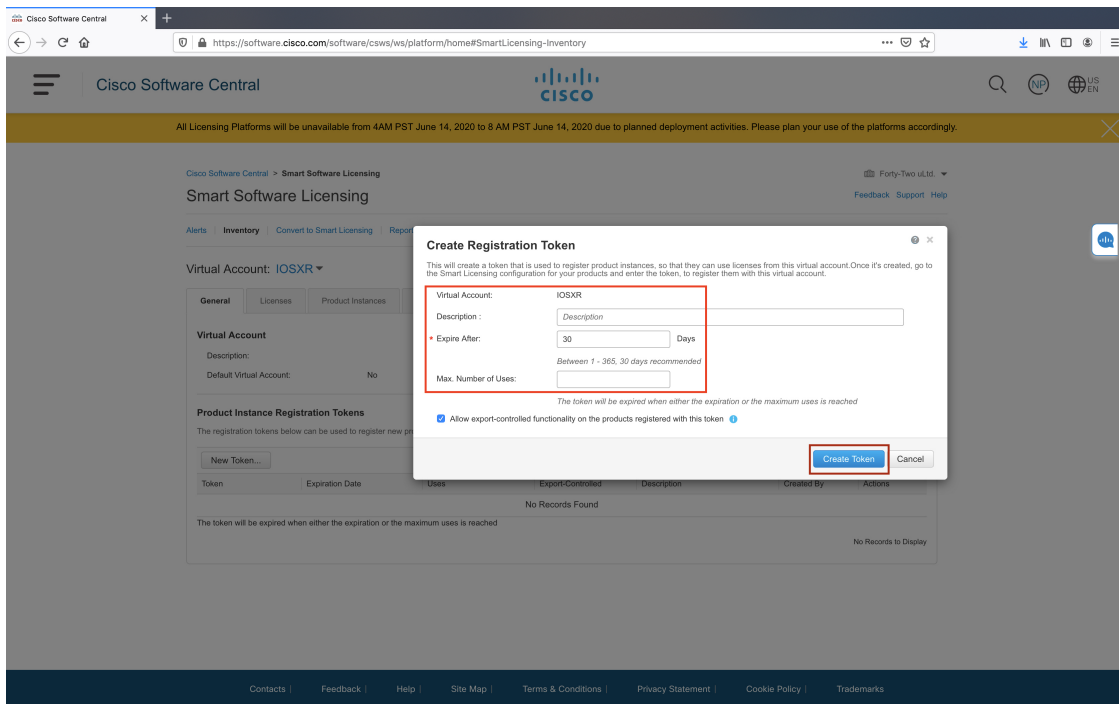


The **Create Registration Token** window is displayed.

- Step 6** In the **Description** field, enter the token description.
- In the **Expire After** field, enter the number of days the token must be active. The default value is 30 days.
- In the **Max. Number of Uses** field, enter the maximum number of uses allowed after which the token expires.
- Select the **Allow export-controlled functionality on the products registered with this token** checkbox to ensure Cisco compliance with US and country-specific export policies and guidelines. For more information, see <https://www.cisco.com/c/en/us/about/legal/global-export-trade.html>.

521050

Generating a New Token from CSSM

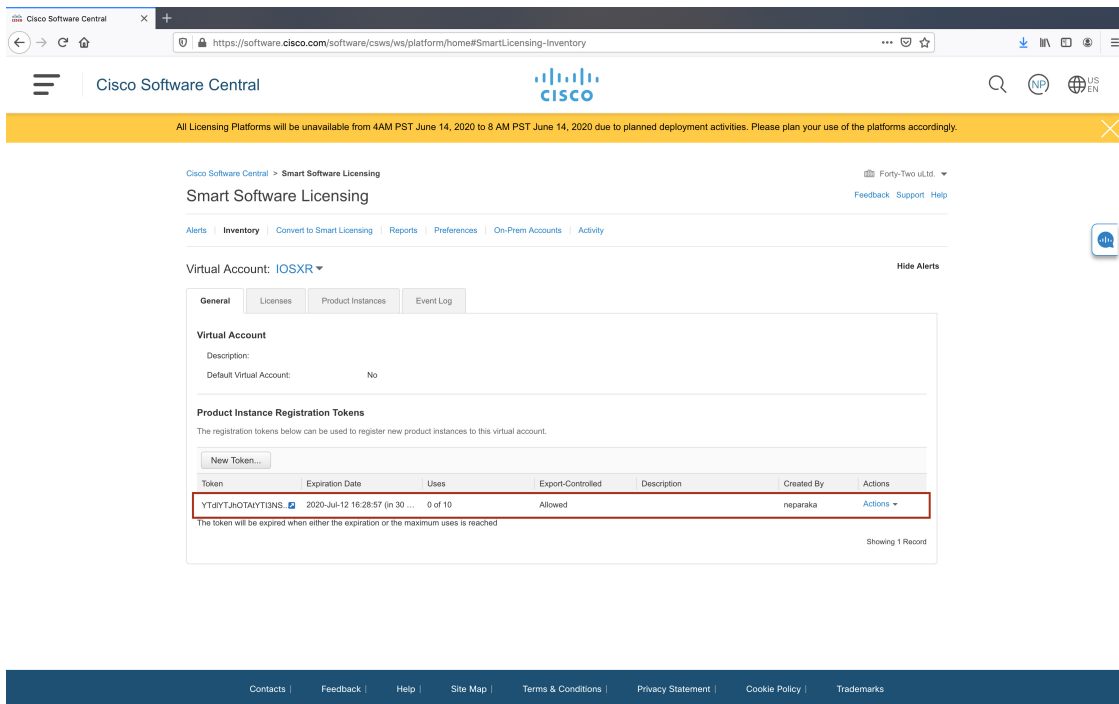


521051

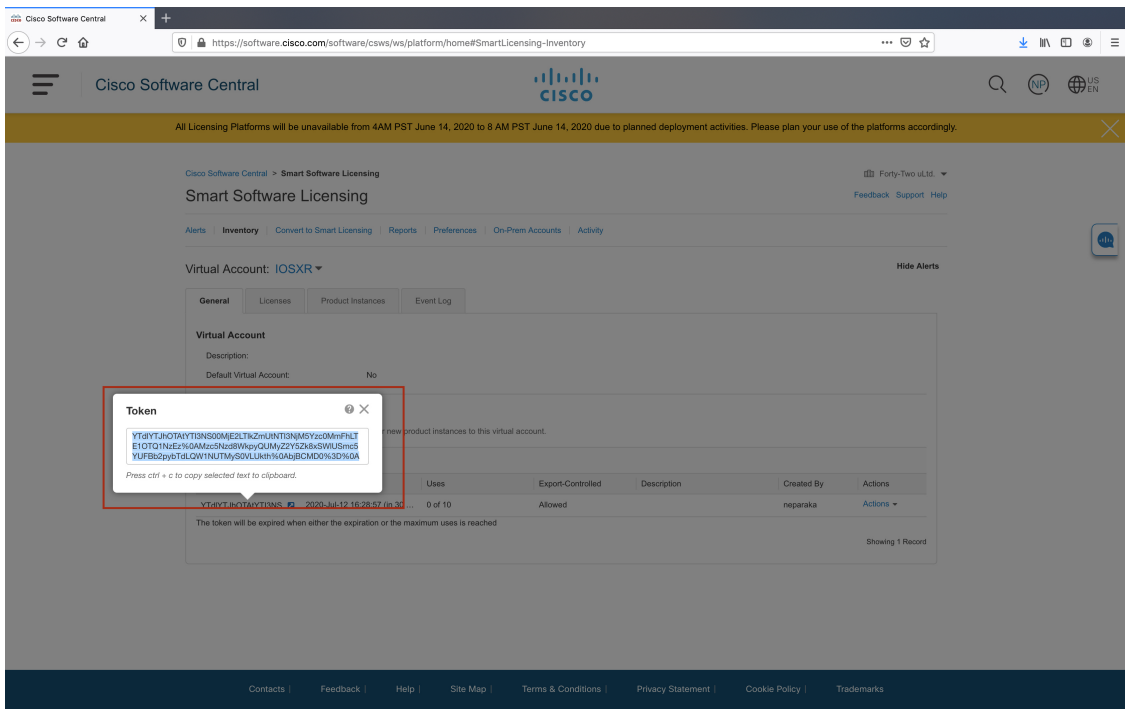
Click **Create Token**.

Step 7

After the token is created, select and copy the token to a text file.



521052



You need this token to register your router.

What to do next

See the *Registering Your Device With the Token* section.

Registering Your Device With the Token

SUMMARY STEPS

1. `license smart register idtoken token-ID`

DETAILED STEPS

	Command or Action	Purpose
Step 1	license smart register idtoken token-ID Example: <pre>license smart register idtoken \$Tl4UytrNXBzbEs1ck8veUtWaG5abnZJOFdDa1FwbVRa%0Ab1RMoz0%3D%0A</pre>	Registers Smart Licensing on the router using the registration token created in the CSSM. On successful registration, the product instance is created in the CSSM virtual account and its license usage is displayed on the CSSM.

Renewing Your Smart Licensing Registration

Your registration is automatically renewed every six months. To find the status of the license, use the **license smart renew auth** command.

As long as the license is in an 'Authorized' or 'Out-of-compliance' (OOC) state, the authorization period is renewed. Grace period starts when an authorization period expires. During the grace period or when the grace period is in the 'Expired' state, the system continues to try to renew the authorization period. If a retry is successful, a new authorization period starts.



Note If the smart license renewal fails, then the product instance goes to an unidentified state and starts consuming the evaluation period.

Before you begin

Ensure that the following conditions are met to renew your smart license:

- Smart licensing is enabled.
- The router is registered.

SUMMARY STEPS

1. **license smart renew {auth | id}**

DETAILED STEPS

	Command or Action	Purpose
Step 1	license smart renew {auth id} Example: Router# license smart renew auth	Renews your token ID or authorization with Cisco smart licensing.

Deregistering Your Router from CSSM

When a router is taken off the inventory, shipped elsewhere for redeployment, or returned to Cisco for replacement, you can deregister that router.

Before you begin

Ensure that a Layer 3 connection to CSSM is available to successfully deregister the device.

SUMMARY STEPS

1. **license smart deregister**

DETAILED STEPS

	Command or Action	Purpose
Step 1	license smart deregister Example: <pre>Router# license smart deregister</pre>	Cancels the registration of the router and sends the router into evaluation mode. All smart licensing entitlements and certificates on the corresponding platform are removed. The product instance of the router stored on CSSM is also removed.

Verifying the Smart Licensing Configuration

Use the following **show** commands to verify the default Smart Licensing configuration. If any issue is detected, take corrective action before making further configurations.

SUMMARY STEPS

1. **show license status**
2. **show license all**
3. **show license status**
4. **show license udi**
5. **show license summary**
6. **show license platform summary**
7. **show license platform detail**
8. **show call-home smart-licensing statistics**

DETAILED STEPS

	Command or Action	Purpose
Step 1	show license status Example: <pre>Router# show license status</pre>	<p>Displays the compliance status of Smart Licensing. Following are the possible status:</p> <ul style="list-style-type: none"> • Waiting—Indicates that the initial state after your device has made a license entitlement request. The device establishes communication with Cisco and successfully registers itself with the Cisco license manager. • Authorized—Indicates that your device is able to communicate with the Cisco license manager, and is authorized to initiate requests for license entitlements. • Out-Of-Compliance—Indicates that one or more of your licenses are out-of-compliance. Buy more licenses, or renew the existing licenses. • Eval Period—Indicates that Smart Licensing is consuming the evaluation period. Register the device with the Cisco Licensing manager, else your license expires.

	Command or Action	Purpose
		<p>Note Repetitive 'Smart Licensing evaluation expired' warning messages are displayed on the console every hour, but there is no functionality impact on the device. To stop these repetitive messages, register the device again with new a registration token.</p> <ul style="list-style-type: none"> • Disabled—Indicates that Smart Licensing is disabled. • Invalid—Indicates that Cisco does not recognize the entitlement tag as the tag is not in the database.
Step 2	show license all Example: Router# show license all	Displays all entitlements in use. The output also displays the associated licensing certificates, compliance status, Unique Device Identifier (UDI), and other details.
Step 3	show license status Example: Router# show license status	Displays the status of all entitlements in use.
Step 4	show license udi Example: Router# show license udi	Displays the Unique Device Identifier (UDI) information.
Step 5	show license summary Example: Router# show license summary	Displays a summary of all entitlements in use.
Step 6	show license platform summary Example: Router# show license platform summary	Displays the registration status and provides detailed information about the essential, advantage without essentials, advantage with essentials, and tracking license consumption in generic or vortex license model.
Step 7	show license platform detail Example: Router# show license platform detail	Displays detailed information about: <ul style="list-style-type: none"> • Licenses that can be consumed on a platform in both, generic and vortex models • The active model, whether generic or vortex model • The current count and the next consumption count of a license
Step 8	show call-home smart-licensing statistics Example: Router# show call-home smart-licensing statistics	Displays statistics of the communication between the Smart Licensing manager and the Cisco back-end using Smart Call Home. <p>Note If the communication fails or drops, check your call home configuration for any errors.</p>

Smart Licensing Configuration Examples

Example: Viewing the Call Home Profile

To display the **http Call Home profile** or the **On-Prem Call Home profile**, use the **show call-home profile all** command.

```
Router# show call-home profile all
Tue Aug 18 23:52:16.590 UTC
```

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

Periodic configuration info message is scheduled every 17 day of the month at 13:15

Periodic inventory info message is scheduled every 17 day of the month at 13:0
```

Alert-group	Severity
-----	-----
inventory	normal

Syslog-Pattern	Severity
-----	-----
.*	critical

```
Router# show call-home profile all
```

```
Wed Aug 19 01:55:14.974 UTC
```

```
Profile Name: CiscoTAC-1
  Profile status: ACTIVE
  Profile mode: Full Reporting
  Reporting Data: Smart Call Home, Smart Licensing
  Preferred Message Format: xml
  Message Size Limit: 3145728 Bytes
  Transport Method: http
  HTTP address(es): http://10.30.110.38/Transportgateway/services/DeviceRequestHandler
  Other address(es): default

Periodic configuration info message is scheduled every 17 day of the month at 13:15

Periodic inventory info message is scheduled every 17 day of the month at 13:0
```

Alert-group	Severity
-----	-----
inventory	normal

Syslog-Pattern	Severity
-----	-----
.*	critical

Example: Viewing License Information Before Registration

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

Example: Viewing License Information Before Registration

```

Router# show license all
Smart Licensing Status
=====

Smart Licensing is ENABLED

Registration:
  Status: UNREGISTERED
  Export-Controlled Functionality: NOT ALLOWED

License Authorization:
  Status: EVAL MODE
  Evaluation Period Remaining: 83 days, 23 hours, 36 minutes, 0 seconds

Export Authorization Key:
  Features Authorized:
    <none>

Utility:
  Status: DISABLED

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

Transport:
  Type: Callhome

License Usage
=====

(ESS-100G-RTU-1):
  Description:
  Count: 1
  Version: 1.0
  Status: EVAL MODE
  Export status: NOT RESTRICTED

(NCS-5501-TRK):
  Description:
  Count: 1
  Version: 1.0
  Status: EVAL MODE
  Export status: NOT RESTRICTED

Product Information
=====
UDI: PID:NCS-5501,SN:FOC2137R1SL

Agent Version
=====
Smart Agent for Licensing: 4.9.6_rel/41

Reservation Info
=====
License reservation: DISABLED

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

Router# show license usage

License Authorization:
  Status: EVAL MODE

```


Evaluation Period Remaining: 83 days, 23 hours, 34 minutes, 34 seconds

```
(ESS-100G-RTU-1):
Description:
Count: 1
Version: 1.0
Status: EVAL MODE
Export status: NOT RESTRICTED
```

```
(NCS-5501-TRK):
Description:
Count: 1
Version: 1.0
Status: EVAL MODE
Export status: NOT RESTRICTED
```

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

```
Router# show license summary
Smart Licensing is ENABLED
```

```
Registration:
Status: UNREGISTERED
Export-Controlled Functionality: NOT ALLOWED
```

```
License Authorization:
Status: EVAL MODE
Evaluation Period Remaining: 83 days, 23 hours, 33 minutes, 52 seconds
```

```
License Usage:
License                               Entitlement tag                Count Status
-----
                                (ESS-100G-RTU-1)              1 EVAL MODE
                                (NCS-5501-TRK)                1 EVAL MODE
```

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

```
Router# show license status
```

```
Smart Licensing is ENABLED
```

```
Utility:
Status: DISABLED
```

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

```
Transport:
Type: Callhome
```

```
Registration:
Status: UNREGISTERED
Export-Controlled Functionality: NOT ALLOWED
```

```
License Authorization:
Status: EVAL MODE
Evaluation Period Remaining: 83 days, 23 hours, 32 minutes, 57 seconds
```

```
Export Authorization Key:
Features Authorized
```

Example: Registering the Router

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

```
Router# license smart register idtoken
Tl4UytrNXBzbEs1ck8veUtWaG5abnZJOFdDa1FwbVRa%0Ab1RMbz0%3D%0A
```

Example: Viewing License Information After Registration

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

```
Router# show license all
Smart Licensing Status
=====

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: Forty-Two uLtd.
  Virtual Account: IOSXR
  Export-Controlled Functionality: ALLOWED
  Initial Registration: SUCCEEDED on Aug 18 2020 23:51:46 UTC
  Last Renewal Attempt: None
  Next Renewal Attempt: Feb 14 2021 23:51:46 UTC
  Registration Expires: Aug 18 2021 23:46:43 UTC

License Authorization:
  Status: OUT OF COMPLIANCE on Aug 18 2020 23:51:57 UTC
  Last Communication Attempt: SUCCEEDED on Aug 18 2020 23:51:57 UTC
  Next Communication Attempt: Aug 19 2020 11:51:57 UTC
  Communication Deadline: Nov 16 2020 23:46:56 UTC

Export Authorization Key:
  Features Authorized:
    <none>

Utility:
  Status: DISABLED

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

Transport:
  Type: Callhome

License Usage
=====

Core and Aggregation Essentials SW Right-to-Use v1.0 per 100G (ESS-100G-RTU-1):
  Description: Core and Aggregation Essentials SW Right-to-Use v1.0 per 100G
  Count: 1
  Version: 1.0
  Status: OUT OF COMPLIANCE
  Export status: NOT RESTRICTED

5501 Base Hardware Tracking PID (NCS-5501-TRK):
  Description: 5501 Base Hardware Tracking PID
  Count: 1
  Version: 1.0
  Status: OUT OF COMPLIANCE
```

```

Export status: NOT RESTRICTED

Product Information
=====
UDI: PID:NCS-5501,SN:FOC2137R1SL

Agent Version
=====
Smart Agent for Licensing: 4.9.6_rel/41

Reservation Info
=====
License reservation: DISABLED

```

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

```

Router# show license usage
License Authorization:
  Status: OUT OF COMPLIANCE on Aug 18 2020 23:51:57 UTC

Core and Aggregation Essentials SW Right-to-Use v1.0 per 100G (ESS-100G-RTU-1):
  Description: Core and Aggregation Essentials SW Right-to-Use v1.0 per 100G
  Count: 1
  Version: 1.0
  Status: OUT OF COMPLIANCE
  Export status: NOT RESTRICTED

5501 Base Hardware Tracking PID (NCS-5501-TRK):
  Description: 5501 Base Hardware Tracking PID
  Count: 1
  Version: 1.0
  Status: OUT OF COMPLIANCE
  Export status: NOT RESTRICTED

```

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

```

Router# show license summary
Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: Forty-Two uLtd.
  Virtual Account: IOSXR
  Export-Controlled Functionality: ALLOWED
  Last Renewal Attempt: None
  Next Renewal Attempt: Feb 14 2021 23:51:46 UTC

License Authorization:
  Status: OUT OF COMPLIANCE
  Last Communication Attempt: SUCCEEDED
  Next Communication Attempt: Aug 19 2020 11:51:56 UTC

License Usage:

```

License	Entitlement tag	Count	Status
Core and Aggregation...	(ESS-100G-RTU-1)	1	OUT OF COMPLIANCE
5501 Base Hardware T...	(NCS-5501-TRK)	1	OUT OF COMPLIANCE

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

```

Router# show license status

Smart Licensing is ENABLED

```

```

Utility:
  Status: DISABLED

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

Transport:
  Type: Callhome

Registration:
  Status: REGISTERED
  Smart Account: Forty-Two uLtd.
  Virtual Account: IOSXR
  Export-Controlled Functionality: ALLOWED
  Initial Registration: SUCCEEDED on Aug 18 2020 23:51:46 UTC
  Last Renewal Attempt: None
  Next Renewal Attempt: Feb 14 2021 23:51:45 UTC
  Registration Expires: Aug 18 2021 23:46:42 UTC

License Authorization:
  Status: OUT OF COMPLIANCE on Aug 18 2020 23:51:57 UTC
  Last Communication Attempt: SUCCEEDED on Aug 18 2020 23:51:57 UTC
  Next Communication Attempt: Aug 19 2020 11:51:56 UTC
  Communication Deadline: Nov 16 2020 23:46:55 UTC

Export Authorization Key:
  Features Authorized:
    <none>

```

Specific License Reservation

Table 14: Feature History Table

Feature Name	Release	Feature Description
Specific License Reservation	Release 7.3.1	Specific License Reservation (SLR) allows customers in highly secure networks to utilize smart licenses without communicating the license information to the Cisco Smart Software manager (CSSM).

SLR is reservation of specific licenses from the smart account as per the usage on the router. Whereas, PLR is for selected customers who don't want tracking of license usage. This license allows unlimited usage without taking into consideration of license usage on the router.

Specific License Reservation (SLR) allows customers in highly secure networks to utilize smart licenses without communicating the license information to the Cisco Smart Software manager (CSSM). Routers that are registered with smart licenses share router information at regular intervals with Cisco Smart Software Manager (CSSM). Routers that are deployed in a highly secure network must not share the router information externally. Cisco offers license reservation as an on-request configuration for these routers.

Specific License Reservation is a solution designed for classified environments that don't allow electronic communication in or out of the environment. In such environments, routers are unable to communicate with

the Cisco Smart Software Manager (CSSM) directly or through CSSM OnPrem. These deployments can use SLR that allows using all entitlements on the router without communicating to Cisco.

To enable and use the SLR feature, generate a reservation request code from the router. Use this code in the CSSM portal to reserve an available SLR license and generate an authorization code. When you enter this authorization code in the router CLI, the license reservation becomes effective.

Limitations

- License reservation provides limited or no functionality to certain smart licensing features such as transfer of licenses between products, license usage, asset management and so on.
- To use the Specific License Reservation feature, you must have an approval and authorization from Cisco. For assistance, go to www.cisco.com/go/scm or contact your account representative.

Configuring Specific License Reservation

Before configuring Specific License Reservation, connect with your account representative and complete the following tasks:

- Create a Smart Account.
- Enable your Smart Account for SLR.
- Get access to SLR PIDs in Cisco Commerce Workspace (CCW).
- Order SLR licenses in CCW and make sure they are available in your Smart Account on CSSM.

Enable Specific License Reservation feature of licensing using the **license smart reservation** command in the global configuration mode.

Workflow

Here is the list of configurations that should be executed in sequence to complete the SLR feature:

1. Enabling Specific License Reservation
2. Generating a Request Code from the Router
3. Reserving a SLR License in Cisco SSM
4. Registering a Router with the Authorization Code

Here is the list of configurations that are used to cancel, verify, remove, and return SLR:

- Cancelling a SLR Request
- Verifying the License Status After Cancelling SLR
- Removing SLR from a Router
- Generating a Return Code from a Router
- Removing the Product Instance from CSSM

Enabling Specific License Reservation

First, enable Flexible Consumption Model (FCM) of licensing. To enable FCM for your hardware, use the **license smart flexible-consumption enable** command in the global configuration mode.

Then, enable Specific License Reservation feature of licensing. In order to enable SLR feature, license reservation must be configured in the global configuration mode:

```
Router(config)# license smart reservation
```

Generating a Request Code from the Router

After enabling the SLR feature, use the **license smart reservation request local** command to generate a request code. Enter this code in CSSM.

Example:

```
RP/0/RSP0/CPU0:router# license smart reservation request local
Fri Mar 6 12:39:02.818 UTC
Enter this request code in the Cisco Smart Software Manager portal:
CD-ZNCS-5501-SE:FOC2118R24P-AVYd1FABK-AC /* This is a sample code */
```

Reserving a SLR License in Cisco SSM

This procedure shows how to generate an authorization code from CSSM portal:

SUMMARY STEPS

1. Go to Cisco SSM from <https://software.cisco.com>
2. Under **License** section, click **Smart Software Licensing**
3. Click the **Inventory** tab and from the **Virtual Account** drop-down list, choose your virtual account.
4. Go to **Licenses** tab and click **License Reservation**.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Go to Cisco SSM from https://software.cisco.com	Log in to the Cisco SSM portal using the username and password provided by Cisco.
Step 2	Under License section, click Smart Software Licensing	
Step 3	Click the Inventory tab and from the Virtual Account drop-down list, choose your virtual account.	You can search by typing your virtual account name, or you can scroll the list and select your account name.
Step 4	Go to Licenses tab and click License Reservation .	The Smart License Reservation window appears. This window has four steps that should be performed in sequence to complete the procedure.

Registering a Router with the Authorization Code

This procedure shows how to register a router with the authorization code that is generated from CSSM:

```
Router# license smart reservation install [<> | file]
```

**Note**

- '<>' is the code itself or an alternative is to import from a file.
- 'file' option is preferred in those cases where the authorization code is long.

Example:

```
RP/0/RSP0/CPU0:router# license smart reservation install file
/disk2:/AuthorizationCode_SN_FOX24XXXXX.txt /* This is a sample code */
```

The "/" before the directory (/disk2:/ or /harddisk:/) is needed because of the linux file path.

The SLR license reservation is enabled, and the SL status changes to Authorized state.

The router stops using the evaluation period.

Verification

To verify the license status after the registration, use the **show license reservation** command:

```
RP/0/RSP0/CPU0:router# show license reservation
Fri Feb 5 16:35:37.340 UTC
License reservation: ENABLED
Overall status:
  Active: PID:NCS-55A2-MOD-S,SN:FOC2245R05H
    Reservation status: RESERVATION IN PROGRESS on Feb 05 2021 16:33:08 UTC
    Request code: CC-ZNCS-55A2-MOD-S:FOC2245R05H-AVYd1FABK-45
```

To cancel, verify, remove, and return SLR

Cancelling a SLR Request

The license reservation request code can be cancelled before or after an authorization code is generated. Note that the authorization file needs to be returned to Cisco SSM if the cancellation request is raised after an authorization code is generated from CSSM.

To cancel an SLR request before the authorization code is installed, perform this procedure:

```
RP/0/RSP0/CPU0:router# license smart reservation cancel local
Fri Feb 5 16:41:55.262 UTC
Reservation cancelled
```

Verifying the License Status After Cancelling SLR

To verify the status of license after cancelling SLR, use the **show license reservation** command:

```
RP/0/RSP0/CPU0:router# show license reservation
Fri Feb 5 16:41:58.284 UTC
License reservation: ENABLED
Overall status:
  Active: PID:NCS-55A2-MOD-S,SN:FOC2245R05H
```

```
Reservation status: NOT INSTALLED
Last return code: CQ8K5Y-gYxar7-baroN9-BH2d4X-EaG7bH-6L7QmQ-mEGwy3-fxSCxo-Dku
```

Removing SLR from a Router

To remove SLR from a router, perform the following tasks:

1. Generate a return code from a router.
2. Enter the return code on CSSM.
3. Remove the product instance.

Generating a Return Code from a Router

To generate a return code, use the **license smart reservation return** command:

To verify the license status after returning a license, use the **show license status** command:

```
Router# license smart reservation return [local]
```

The command prints the reservation return code that must be entered in the CSSM portal to return the license to the virtual pool.

Removing the Product Instance from CSSM

This procedure shows how to remove the product instance from CSSM portal:

SUMMARY STEPS

1. Go to Cisco SSM from <https://software.cisco.com>
2. Under **License** section, click **Smart Software Licensing**
3. Click the **Inventory** tab and from the **Virtual Account** drop-down list, choose your virtual account.
4. Click the **Product** instances tab.
5. Locate the required product instance from the product instances list. Optionally, you can enter a name or product type string in the search tab to locate the product instance.
6. Click the required product instance to expand the same.
7. From the **Actions** drop-down list, choose **Remove**.
8. In the **Reservation Return Code** field, enter the return code.
9. Click **Remove Product Instance**.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Go to Cisco SSM from https://software.cisco.com	Log in to the Cisco SSM portal using the username and password provided by Cisco.
Step 2	Under License section, click Smart Software Licensing	
Step 3	Click the Inventory tab and from the Virtual Account drop-down list, choose your virtual account.	You can search by typing your virtual account name, or you can scroll the list and select your account name.

	Command or Action	Purpose
Step 4	Click the Product instances tab.	The list of product instances that are available is displayed.
Step 5	Locate the required product instance from the product instances list. Optionally, you can enter a name or product type string in the search tab to locate the product instance.	
Step 6	Click the required product instance to expand the same.	The Overview window is displayed.
Step 7	From the Actions drop-down list, choose Remove .	The Remove Product Instance window is displayed.
Step 8	In the Reservation Return Code field, enter the return code.	
Step 9	Click Remove Product Instance .	

The reservation information is removed, and the license is made available in the virtual account.

Configuration Examples for SLR

The following sections provide various SLR examples.

Enabling FCM

To enable FCM, use the **license smart flexible-consumption enable** command:

```
Router# configure
Router(config)# license smart flexible-consumption enable
Router(config)# exit
```

Enabling SLR

Before enabling SLR, you must enable Flexible Consumption Model (FCM) of licensing. To enable FCM for your hardware, use the **license smart flexible-consumption enable** command in the global configuration mode.

To enable SLR, use the **license smart reservation** command in the global configuration mode:

```
Router# configure
Router(config)# license smart reservation
Router(config)# exit
```

Viewing the License Information Before Registering

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

```
RP/0/RSP0/CPU0:router# show license all
Fri Feb 5 16:35:41.983 UTC

Smart Licensing Status
=====
Smart Licensing is ENABLED
License Reservation is ENABLED
```

```

Registration:
Status: RESERVATION IN PROGRESS
Export-Controlled Functionality: NOT ALLOWED
Reservation process started on: Feb 05 2021 16:33:08 UTC

License Authorization:
Status: EVAL MODE
Evaluation Period Remaining: 81 days, 21 hours, 24 minutes, 53 seconds

Export Authorization Key:
Features Authorized:
<none>

Utility:
Status: DISABLED

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

Transport:
Type: Transport Off

Miscellaneous:
Custom Id: <empty>

License Usage
=====
(ADV-100G-RTU-1):
Description:
Count: 1
Version: 1.0
Status: EVAL MODE
Export status: NOT RESTRICTED
Reservation:
Reservation status: NOT INSTALLED

(ADV-100G-SIA-3):
Description:
Count: 1
Version: 1.0
Status: EVAL MODE
Export status: NOT RESTRICTED
Reservation:
Reservation status: NOT INSTALLED

(ESS-100G-RTU-1):
Description:
Count: 1
Version: 1.0
Status: EVAL MODE
Export status: NOT RESTRICTED
Reservation:
Reservation status: NOT INSTALLED

(ESS-100G-SIA-3):
Description:
Count: 1
Version: 1.0
Status: EVAL MODE
Export status: NOT RESTRICTED

```

```
Reservation:
Reservation status: NOT INSTALLED

Product Information
=====
UDI: PID:NCS-55A2-MOD-S,SN:FOC2245R05H

Agent Version
=====
Smart Agent for Licensing: 5.1.8_rel/25

Reservation Info
=====
License reservation: ENABLED

Overall status:
Active: PID:NCS-55A2-MOD-S,SN:FOC2245R05H
      Reservation status: RESERVATION IN PROGRESS on Feb 05 2021 16:33:08 UTC
      Request code: CC-ZNCS-55A2-MOD-S:FOC2245R05H-AVYd1FABK-45
```

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

```
RP/0/RSP0/CPU0:router# show license usage
```

```
Fri Feb 5 16:40:55.693 UTC
```

```
License Authorization:
Status: EVAL MODE
Evaluation Period Remaining: 81 days, 21 hours, 22 minutes, 46 seconds
```

```
(ADV-100G-RTU-1):
Description:
Count: 1
Version: 1.0
Status: EVAL MODE
Export status: NOT RESTRICTED
Reservation:
Reservation status: NOT INSTALLED
```

```
(ADV-100G-SIA-3):
Description:
Count: 1
Version: 1.0
Status: EVAL MODE
Export status: NOT RESTRICTED
Reservation:
Reservation status: NOT INSTALLED
```

```
(ESS-100G-RTU-1):
Description:
Count: 1
Version: 1.0
Status: EVAL MODE
Export status: NOT RESTRICTED
Reservation:
Reservation status: NOT INSTALLED
```

```
(ESS-100G-SIA-3):
Description:
Count: 1
Version: 1.0
Status: EVAL MODE
Export status: NOT RESTRICTED
Reservation:
```

Reservation status: NOT INSTALLED

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

```
RP/0/RSP0/CPU0:router# show license summary
```

Fri Feb 5 16:35:32.192 UTC

Smart Licensing is ENABLED
License Reservation is ENABLED

Registration:
Status: RESERVATION IN PROGRESS
Export-Controlled Functionality: NOT ALLOWED

License Authorization:
Status: EVAL MODE
Evaluation Period Remaining: 81 days, 21 hours, 25 minutes, 3 seconds

License	Entitlement Tag	Count	Status
	(ADV-100G-RTU-1)	1	EVAL MODE
	(ADV-100G-SIA-3)	1	EVAL MODE
	(ESS-100G-RTU-1)	1	EVAL MODE
	(ESS-100G-SIA-3)	1	EVAL MODE

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

```
RP/0/RSP0/CPU0:router# show license status
```

Fri Feb 5 16:35:47.566 UTC

Smart Licensing is ENABLED

Utility:
Status: DISABLED
License Reservation is ENABLED

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

Transport:
Type: Transport Off

Registration:
Status: RESERVATION IN PROGRESS
Export-Controlled Functionality: NOT ALLOWED
Reservation process started on: Feb 05 2021 16:33:08 UTC

License Authorization:
Status: EVAL MODE
Evaluation Period Remaining: 81 days, 21 hours, 24 minutes, 47 seconds

Export Authorization Key:
Features Authorized:
<none>

Miscellaneous:
Custom Id: <empty>

Viewing the License Status After Registering

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

```
RP/0/RSP0/CPU0:router# show license all

Fri Feb 5 16:37:57.228 UTC

Smart Licensing Status
=====
Smart Licensing is ENABLED
License Reservation is ENABLED

Registration:
  Status: REGISTERED - SPECIFIC LICENSE RESERVATION
  Export-Controlled Functionality: ALLOWED
  Initial Registration: SUCCEEDED on Feb 05 2021 16:37:38 UTC

License Authorization:
  Status: AUTHORIZED - RESERVED on Feb 05 2021 16:37:38 UTC

Export Authorization Key:
  Features Authorized:
    <none>

Utility:
  Status: DISABLED

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

Transport:
  Type: Transport Off

Miscellaneous:
  Custom Id: <empty>

License Usage
=====
Core and Aggregation Advanced SW Right-to-Use v1.0 per 100G (ADV-100G-RTU-1):
  Description: Core and Aggregation Advanced SW Right-to-Use v1.0 per 100G
  Count: 1
  Version: 1.0
  Status: AUTHORIZED
  Export status: NOT RESTRICTED
  Reservation:
    Reservation status: SPECIFIC INSTALLED
    Total reserved count: 1

Core & Aggr Advanced SIA per 100G (ADV-100G-SIA-3):
  Description: Core & Aggr Advanced SIA per 100G
  Count: 1
  Version: 1.0
  Status: AUTHORIZED
  Export status: NOT RESTRICTED
  Reservation:
    Reservation status: SPECIFIC INSTALLED
    Total reserved count: 1

Core and Aggregation Essentials SW Right-to-Use v1.0 per 100G (ESS-100G-RTU-1):
  Description: Core and Aggregation Essentials SW Right-to-Use v1.0 per 100G
```

```

Count: 1
Version: 1.0
Status: AUTHORIZED
Export status: NOT RESTRICTED
Reservation:
    Reservation status: SPECIFIC INSTALLED
    Total reserved count: 1

Core & Aggr Essentials SIA per 100G (ESS-100G-SIA-3):
Description: Core & Aggr Essentials SIA per 100G
Count: 1
Version: 1.0
Status: AUTHORIZED
Export status: NOT RESTRICTED
Reservation:
    Reservation status: SPECIFIC INSTALLED
    Total reserved count: 1

Product Information
=====
UDI: PID:NCS-55A2-MOD-S,SN:FOC2245R05H

Agent Version
=====
Smart Agent for Licensing: 5.1.8_rel/25

Reservation Info
=====
License reservation: ENABLED

Overall status:
    Active: PID:NCS-55A2-MOD-S,SN:FOC2245R05H
    Reservation status: SPECIFIC INSTALLED on Feb 05 2021 16:37:38 UTC
    Last Confirmation code: 21662953

Specified license reservations:
    Core & Aggr Advanced SIA per 100G (ADV-100G-SIA-3):
        Description: Core & Aggr Advanced SIA per 100G
        Total reserved count: 1
        Enforcement type: ENFORCED
        Term information:
            Active: PID:NCS-55A2-MOD-S,SN:FOC2245R05H
            Authorization type: SPECIFIC INSTALLED on Feb 05 2021 16:37:38 UTC
            License type: TERM
            Start Date: 2020-SEP-11 UTC
            End Date: 2021-SEP-11 UTC
            Term Count: 1
    Core & Aggr Essentials SIA per 100G (ESS-100G-SIA-3):
        Description: Core & Aggr Essentials SIA per 100G
        Total reserved count: 1
        Enforcement type: ENFORCED
        Term information:
            Active: PID:NCS-55A2-MOD-S,SN:FOC2245R05H
            Authorization type: SPECIFIC INSTALLED on Feb 05 2021 16:37:38 UTC
            License type: TERM
            Start Date: 2020-SEP-11 UTC
            End Date: 2021-SEP-11 UTC
            Term Count: 1
    Core and Aggregation Advanced SW Right-to-Use v1.0 per 100G (ADV-100G-RTU-1):
        Description: Core and Aggregation Advanced SW Right-to-Use v1.0 per 100G
        Total reserved count: 1
        Enforcement type: ENFORCED
        Term information:
            Active: PID:NCS-55A2-MOD-S,SN:FOC2245R05H

```

```

Authorization type: SPECIFIC INSTALLED on Feb 05 2021 16:37:38 UTC
License type: TERM
Start Date: 2021-JAN-11 UTC
End Date: 2022-JAN-11 UTC
Term Count: 1
Core and Aggregation Essentials SW Right-to-Use v1.0 per 100G (ESS-100G-RTU-1):
Description: Core and Aggregation Essentials SW Right-to-Use v1.0 per 100G
Total reserved count: 1
Enforcement type: ENFORCED
Term information:
Active: PID:NCS-55A2-MOD-S,SN:FOC2245R05H
Authorization type: SPECIFIC INSTALLED on Feb 05 2021 16:37:38 UTC
License type: TERM
Start Date: 2021-JAN-11 UTC
End Date: 2022-JAN-11 UTC
Term Count: 1

```

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

```

RP/0/RSP0/CPU0:router# show license usage

Fri Feb 5 16:38:08.167 UTC

License Authorization:
Status: AUTHORIZED - RESERVED on Feb 05 2021 16:37:38 UTC

Core and Aggregation Advanced SW Right-to-Use v1.0 per 100G (ADV-100G-RTU-1):
Description: Core and Aggregation Advanced SW Right-to-Use v1.0 per 100G
Count: 1
Version: 1.0
Status: AUTHORIZED
Export status: NOT RESTRICTED
Reservation:
Reservation status: SPECIFIC INSTALLED
Total reserved count: 1

Core & Aggr Advanced SIA per 100G (ADV-100G-SIA-3):
Description: Core & Aggr Advanced SIA per 100G
Count: 1
Version: 1.0
Status: AUTHORIZED
Export status: NOT RESTRICTED
Reservation:
Reservation status: SPECIFIC INSTALLED
Total reserved count: 1

Core and Aggregation Essentials SW Right-to-Use v1.0 per 100G (ESS-100G-RTU-1):
Description: Core and Aggregation Essentials SW Right-to-Use v1.0 per 100G
Count: 1
Version: 1.0
Status: AUTHORIZED
Export status: NOT RESTRICTED
Reservation:
Reservation status: SPECIFIC INSTALLED
Total reserved count: 1

Core & Aggr Essentials SIA per 100G (ESS-100G-SIA-3):
Description: Core & Aggr Essentials SIA per 100G
Count: 1
Version: 1.0
Status: AUTHORIZED
Export status: NOT RESTRICTED
Reservation:
Reservation status: SPECIFIC INSTALLED
Total reserved count: 1

```

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

```
RP/0/RSP0/CPU0:router# show license summary
```

```
Fri Feb 5 16:37:51.470 UTC
```

```
Smart Licensing is ENABLED
License Reservation is ENABLED
```

```
Registration:
```

```
Status: REGISTERED - SPECIFIC LICENSE RESERVATION
Export-Controlled Functionality: ALLOWED
```

```
License Authorization:
```

```
Status: AUTHORIZED - RESERVED
```

```
License Usage:
```

License	Entitlement Tag	Count	Status
Core and Aggregation...	(ADV-100G-RTU-1)	1	AUTHORIZED
Core & Aggr Advanced...	(ADV-100G-SIA-3)	1	AUTHORIZED
Core and Aggregation...	(ESS-100G-RTU-1)	1	AUTHORIZED
Core & Aggr Essentia...	(ESS-100G-SIA-3)	1	AUTHORIZED

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

```
RP/0/RSP0/CPU0:router# show license status
```

```
Fri Feb 5 16:38:03.370 UTC
```

```
Smart Licensing is ENABLED
```

```
Utility:
```

```
Status: DISABLED
```

```
License Reservation is ENABLED
```

```
Data Privacy:
```

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

```
Transport:
```

```
Type: Transport Off
```

```
Registration:
```

```
Status: REGISTERED - SPECIFIC LICENSE RESERVATION
Export-Controlled Functionality: ALLOWED
Initial Registration: SUCCEEDED on Feb 05 2021 16:37:38 UTC
```

```
License Authorization:
```

```
Status: AUTHORIZED - RESERVED on Feb 05 2021 16:37:38 UTC
```

```
Export Authorization Key:
```

```
Features Authorized:
<none>
```

```
Miscellaneous:
```

```
Custom Id: <empty>
```




CHAPTER 13

Configuring Zero Touch Provisioning

Zero Touch Provisioning (ZTP) works as a Third Party App (TPA) in Route-Switch Processor (RSP) and Route Processor (RP). ZTP was designed to perform two different operations:

- Download and apply an initial configuration.
- Download and execute a shell script.

If the downloaded file content starts with **!! IOS XR** it is considered as a configuration file, and ZTP performs **apply_config** action on the configuration file.

If the downloaded file content starts with **#!/bin/bash**, **#!/bin/sh** or **#!/usr/bin/python** it is considered as a script file, and ZTP executes the script.

ZTP works as following:

1. XR scripts that run on boot, invoke DHCP request.



Note Starting with Cisco IOS XR Release 7.0.1, ZTP follows a default sequential flow as defined in the `ztp.ini` file. ZTP first sends IPv4 DHCP request on all the management ports. In case there is a failure, then ZTP sends IPv6 DHCP request on all the management ports. Similarly, the same order is followed on all the data ports.

2. DHCP server returns either a user script or configuration file.
3. Download the user script or configuration file.
4. Execute the downloaded user script or apply the downloaded configuration.

Prior to Cisco IOS XR Release 6.3.1, ZTP was executed within the default network namespace and could not access the data interfaces directly. Starting with Cisco IOS XR Release 6.3.1, ZTP is executed inside the global Virtual Routing and Forwarding (VRF) network namespace with full access to all the data interfaces.

When ZTP process encounters any error, or when ZTP quits or terminates, it revert to the initial configuration that exists before starting of ZTP process.

**Note**

- When initiated, ZTP checks if the system start-up configuration is applied. If startup configuration is not applied, ZTP waits for 10 minutes before proceeding.
- To boot an image through ZTP, configure the ROMMON reboot mode option to 3.

ZTP Process Flow Sequence

Before Cisco IOS XR Release 7.0.1, during the fresh boot of a router, auto ZTP process is initiated from the management port and in case of failure switches to data port.

Starting with Cisco IOS XR Release 7.0.1, the ZTP process follows a default sequential flow defined in `ztp.ini` file during fresh boot of a router. The following is the default sequence:

1. ZTP sends IPv4 DHCP request first on all the management port. In case there is a failure, then ZTP sends IPv6 DHCP request on all the management port.
2. ZTP sends IPv4 DHCP request first on all the data port. In case there is a failure, then ZTP sends IPv6 DHCP request on all the data port.

**Note**

You can modify the sequence using the `ztp.ini` file.

- [Manual ZTP Invocation](#) , on page 224
- [Authentication on Data Ports](#), on page 225
- [ZTP Bootscript](#), on page 227
- [ZTP Utilities](#), on page 228
- [Customize the ZTP Configurable Options](#), on page 229
- [Examples](#), on page 230

Manual ZTP Invocation

Manual Zero Touch Provisioning (ZTP) can be invoked manually via CLI commands. This manual way helps you to provision the router in stages. Ideal for testing out ZTP configuration without a reboot. If you would like to invoke a ZTP on an interfaces(data ports or management port), you don't have to bring up and configure the interface first. You can execute the **ztp initiate** command, even if the interface is down, ZTP script will bring it up and invoke dhclient. So ZTP could run over all interfaces no matter it is up or down.

Use the **ztp initiate**, **ztp breakout**, **ztp terminate**, **ztp enable**, **ztp disable**, and **ztp clean** commands to force ZTP to run over more interfaces.

- **ztp initiate**— Invokes a new ZTP session. Logs can be found in `/var/log/ztp.log`.
- **ztp terminate**—Terminates any ZTP session in progress.
- **ztp breakout**—Will perform 4x10 breakout detection.
- **ztp enable**—Enables the ZTP at boot.
- **ztp disable**—Disables the ZTP at boot.

- **ztp clean**—Removes only the ZTP state files.

From release 6.2.3, the log file `ztp.log` is saved in `/var/log` folder, and a copy of log file is available at `/disk0:/ztp/ztp.log` location using a soft link. However, executing **ztp clean** clears files saved on disk and not on `/var/log` folder where current ZTP logs are saved. In order to have a log from current ZTP run, you must manually clear the ZTP log file from `/var/log/` folder.

For more information of the commands, see the ZTP command chapter in the .

This task shows the most common use case of manual ZTP invocation: invoke 4x10 breakout discovery and ZTP.

SUMMARY STEPS

1. **ztp breakout**
2. **ztp initiate dataport**

DETAILED STEPS

	Command or Action	Purpose
Step 1	ztp breakout Example: <pre>RP/0/RP0/CPU0:router# ztp breakout</pre>	Tries the 4x10 breakout on 100 GE interfaces that supports breakout and are operationally down after no-shut. If the 10x10 breakout configure brings any 10GE interface operationally up, the breakout configuration is retained; if not, the breakout configuration is reverted.
Step 2	ztp initiate dataport Example: <pre>RP/0/RP0/CPU0:router# ztp initiate dataport</pre> <pre>Wed Apr 22 10:52:24.417 UTC Invoke ZTP? (this may change your configuration) [confirm] [y/n] :y ZTP will now run in the background. ZTP might bring up the interfaces if they are in shutdown state. Please use "show logging" or look at /disk0:/ztp/ztp.log to check progress.</pre>	Invokes DHCP sessions on all data ports that are either up or could be brought up. ZTP runs in the background.

Authentication on Data Ports

On fresh boot, ZTP process is initiated from management ports and may switch to data ports. To validate the connection with DHCP server, authentication is performed on data ports through DHCP option 43 for IPv4 and option 17 for IPv6. These DHCP options are defined in option space and are included within **dhcpgd.conf** and **dhcpgd6.conf** configuration files. You must provide following parameters for authentication while defining option space:

- Authentication code—The authentication code is either 0 or 1; where 0 indicates that authentication is not required, and 1 indicates that MD5 checksum is required.
- Client identifier—The client identifier must be 'exr-config'.

- MD5 checksum—This is chassis serial number. It can be obtained using `echo -n $SERIALNUMBER | md5sum | awk '{print $1}'`.

Here is the sample **dhcpd.conf** configuration. In the example below, the option space called **VendorInfo** is defined with three parameters for authentication:

```
class "vendor-classes" {
    match option vendor-class-identifier;
}

option space VendorInfo;
option VendorInfo.clientId code 1 = string;
option VendorInfo.authCode code 2 = unsigned integer 8;
option VendorInfo.md5sum code 3 = string
option vendor-specific code 43 = encapsulate VendorInfo;
subnet 10.65.2.0 netmask 255.255.255.0 {
    option subnet-mask 255.255.255.0;
    option routers 10.65.2.1;
    range 10.65.2.1 10.65.2.200;
}
host xrv9k-1-mgmt {
    hardware ethernet 00:50:60:45:67:01;
    fixed-address 10.65.2.39;
    vendor-option-space VendorInfo;
    option VendorInfo.clientId "exr-config";
    option VendorInfo.authCode 1;
    option VendorInfo.md5sum "aedef5c457c36390c664f5942ac1ae3829";
    option bootfile-name "http://10.65.2.1:8800/admin-cmd.sh";
}
```

Here is the sample **dhcpd6.conf** configuration file. In the example below, the option space called **VendorInfo** is defined that has code width 2 and length width 2 (as per dhcp standard for IPv6) with three parameters for authentication:

```
log-facility local7;
option dhcp6.name-servers 2001:1451:c632:1::1;
option dhcp6.domain-search "cisco.com";
dhcpv6-lease-file-name "/var/lib/dhcpd/dhcpd6.leases";
option dhcp6.info-refresh-time 21600;
option dhcp6.bootfile-url code 59 = string;
option dhcp6.user-class code 15 = string;
option space CISCO-EXR-CONFIG code width 2 length width 2;
option CISCO-EXR-CONFIG.client-identifier code 1 = string;
option CISCO-EXR-CONFIG.authCode code 2 = integer 8;
option CISCO-EXR-CONFIG.md5sum code 3 = string;
option vsio.CISCO-EXR-CONFIG code 9 = encapsulate CISCO-EXR-CONFIG;
subnet6 2001:1451:c632:1::/64{
    range6 2001:1451:c632:1::2 2001:1451:c632:1::9;
    #host NCS5501-2 {
        #host-identifier option dhcp6.client-id
        00:02:00:00:00:09:46:4f:43:32:30:35:31:52:30:57:34:00;
        option CISCO-EXR-CONFIG.client-identifier "exr-config";
        option CISCO-EXR-CONFIG.authCode 1;
        #invalid md5
        #option CISCO-EXR-CONFIG.md5sum "90fd845ac82c77f834d57a034658d0f1";
        #valid md5
        option CISCO-EXR-CONFIG.md5sum "90fd845ac82c77f834d57a034658d0f0";
        if option dhcp6.user-class = 00:04:69:50:58:45 {
            option dhcp6.bootfile-url "http://[2001:1851:c632:1::1]/NCS5501-2/image.iso";
        }
        else {
            #option dhcp6.bootfile-url

```

```
"http://[2001:1851:c632:1::1]/NCS5501-2/ncs5500-mini-x.iso.sh";
    option dhcp6.bootfile-url "http://[2001:1851:c632:1::1]/NCS5501-2/ztp.cfg";
  }
#}
}
```

ZTP Bootscript

If you want to hard code a script to be executed every boot, configure the following.

```
conf t
  ztp bootscript /disk0:/myscript
commit
```

The above configuration will wait for the first data-plane interface to be configured and then wait an additional minute for the management interface to be configured with an IP address, to ensure that we have connectivity in the third party namespace for applications to use. If the delay is not desired, use:

```
conf t
  ztp bootscript preip /disk0:/myscript
commit
```



Note When the above command is first configured, you will be prompted if you wish to invoke it now. The prompt helps with testing.

This is the example content of **/disk0:/myscript**:

```
#!/bin/bash
exec &> /dev/console # send logs to console
source /pkg/bin/ztp_helper.sh

# If we want to only run one time:
xrcmd "show running" | grep -q myhostname
if [[ $? -eq 0 ]]; then
  echo Already configured
fi

# Set the hostname
cat >/tmp/config <<%%
!! XR config example
hostname myhostname
%%
xrappl /tmp/config

#
# Force an invoke of ZTP again. If there was a username normally it would not run. This
forces it.
# Kill off ztp if it is running already and suppress errors to the console when ztp runs
below and
# cleans up xrcmd that invokes it. ztp will continue to run however.
#
xrcmd "ztp terminate noprompt" 2>/dev/null
xrcmd "ztp initiate noprompt" 2>/dev/null
```

ZTP Utilities

ZTP includes a set of shell utilities that can be sourced within the user script. **ztp_helper.sh** is a shell script that can be sourced by the user script. **ztp_helper.sh** provides simple utilities to access some XR functionalities. Following are the bash functions that can be invoked:

- **xrcmd**—Used to run a single XR exec command:

```
xrcmd "show running"
```

- **xrapply**—Applies the block of configuration, specified in a file:

```
cat >/tmp/config <<%%
!! XR config example
hostname nodel-mgmt-via-xrapply
%%
xrapply /tmp/config
```

- **xrapply_with_reason**—Used to apply a block of XR configuration along with a reason for logging purpose:

```
cat >/tmp/config <<%%
!! XR config example
hostname nodel-mgmt-via-xrapply
%%
xrapply_with_reason "this is a system upgrade" /tmp/config
```

- **xrapply_string**—Used to apply a block of XR configuration in one line:

```
xrapply_string "hostname foo\ninterface GigabitEthernet0/0/0/0\nipv4 address 1.2.3.44
255.255.255.0\n"
```

- **xrapply_string_with_reason**—Used to apply a block of XR configuration in one line along with a reason for logging purposes:

```
xrapply_string_with_reason "system renamed again" "hostname venus\n interface
TenGigE0/0/0/0\n ipv4 address 172.30.0.144/24\n"
```

- **xrreplace**—Used to apply XR configuration replace in XR namespace via a file.

```
cat rtr.cfg <<%%
!! XR config example
hostname nodel-mgmt-via-xrreplace
%%
xrreplace rtr.cfg
```

- **admincmd**—Used to run an admin CLI command in XR namespace. Logs can be found in **/disk0:/ztp/ztp_admincmd.log**

```
admincmd running [show platform]

ztp-user connected from 192.0168.0.1 using console on host
sysadmin-vm:0_RP0# show platform | nomore
Tue Jan 30 23:12:30.757 UTC
Location Card Type HW State SW State Config State
-----
0/RP0 NCS-5501 OPERATIONAL OPERATIONAL NSHUT
```

```
0/FT0 NCS-1RU-FAN-FW OPERATIONAL N/A NSHUT
0/FT1 NCS-1RU-FAN-FW OPERATIONAL N/A NSHUT
0/PM0 NCS-1100W-ACFW OPERATIONAL N/A NSHUT
0/PM1 NCS-1100W-ACFW OPERATIONAL N/A NSHUT
```

- **xrapply_with_extra_auth**—Used to apply XR configuration that requires authentication, in XR namespace via a file. The **xrapply_with_extra_auth** API is used when configurations that require additional authentication to be applied such as alias, flex groups.

```
cat >/tmp/config <<%%
!! XR config example
alias exec alarms show alarms brief system active
alias exec version run cat /etc/show_version.txt
%%
xrapply_with_extra_auth >/tmp/config
```

- **xrreplace_with_extra_auth**—Used to apply XR configuration replace in XR namespace via a file The **xrreplace_with_extra_auth** API is used when configurations that require additional authentication to be applied such as alias, flex groups

```
cat >/tmp/config <<%%
!! XR config example
alias exec alarms show alarms brief system active
alias exec version run cat /etc/show_version.txt
%%
xrreplace_with_extra_auth >/tmp/config
```

Customize the ZTP Configurable Options

Starting with Cisco IOS XR Release 7.0.1, you can customize the following ZTP configurable options in the *ztp.ini* file:

- **ZTP**: You can enable or disable ZTP at boot using CLI or by editing the *ztp.ini* file.
- **Retry**: Set the ZTP DHCP retry mechanism: The available values are infinite and once.
- **Fetcher Priority**: Fetcher defines which port ZTP should use to get the provisioning details. By default, each port has a fetcher priority defined in the *ztp.ini* file. You can modify the default priority of the fetcher. Allowed range is from 0 to 9.



Note Lower the number higher the priority. The value 0 has the highest priority and 9 has the lowest priority.

By default, the USB port has the higher priority.

In the following example, the Mgmt4 port has the highest priority:

```
[Fetcher Priority]
Mgmt4: 0
Mgmt6: 1
DPort4: 2
DPort6: 3
```

- `progress_bar`: Enable progress bar on the console. By default, the progress bar is disabled. To enable the progress bar, add the following entry in the `ztp.ini` file.

```
[Options]
progress_bar: True
```

The following example shows the sample of the `ztp.ini` file:

```
[Startup]
start: True
retry_forever: True

[Fetcher Priority]
Mgmt4: 0
Mgmt6: 1
DPort4: 2
DPort6: 3
```

Enable ZTP Using CLI

If you want to enable ZTP using CLI, use the **ztp enable** command.

Configuration example

```
Router#ztp enable
Fri Jul 12 16:09:02.154 UTC
Enable ZTP? [confirm] [y/n] :y
ZTP Enabled.
```

Disable ZTP Using CLI

If you want to disable ZTP using CLI, use the **ztp disable** command.

Configuration example

```
Router#ztp disable
Fri Jul 12 16:07:18.491 UTC
Disable ZTP? [confirm] [y/n] :y
ZTP Disabled.
Run ZTP enable to run ZTP again.
```

Examples

ZTP logs its operation on the flash file system in the directory `/disk0:/ztp/`. ZTP logs all the transaction with the DHCP server and all the state transition. Prior executions of ZTP are also logged in `/disk0:/ztp/old_logs/`.

The following example displays the execution of a simple configuration script downloaded from a data interface using the command **ztp initiate interface Ten 0/0/0/0 verbose**, this script will unshut all the interfaces of the system and configure a load interval of 30 seconds on all of them.

```
#!/bin/bash
#####
# *** Be careful this is powerful and can potentially destroy your system ***
# *** !!! Use at your own risk !!! ***
#
# Script file should be saved on the backend HTTP server
#####

source ztp_helper.sh
```



```

config_file="/tmp/config.txt"
interfaces=$(xrcmd "show interfaces brief")

function activate_all_if(){
    arInt=$(echo $interfaces | grep -oE '(Te|Fo|Hu) [0-9]*/[0-9]*/[0-9]*/[0-9]*')
    for int in ${arInt[*]}; do
        echo -ne "interface $int\n no shutdown\n load-interval 30\n" >> $config_file
    done
    xrapplly_with_reason "Initial ZTP configuration" $config_file
}

### Script entry point
if [ -f $config_file ]; then
    /bin/rm -f $config_file
else
    /bin/touch $config_file
fi
activate_all_if;
exit 0

```

The following example displays the ZTP logging output:

```

Oct 11 11:05:38 172.30.0.54 ztp-script: Hello from ncs-5001-c !!!
Oct 11 11:05:40 172.30.0.54 ztp-script: current=6.1.1, desired=6.1.1
Oct 11 11:05:40 172.30.0.54 ztp-script: Version match, proceeding to configuration
Oct 11 11:05:41 172.30.0.54 ztp-script: Starting autoprovision process...
Oct 11 11:05:42 172.30.0.54 ztp-script: ### XR K9SEC INSTALL ###
Oct 11 11:05:44 172.30.0.54 ztp-script: ### Downloading complete ###
Oct 11 11:05:55 172.30.0.54 ztp-script: Waiting for k9sec package to be activated
Oct 11 11:06:01 172.30.0.54 ztp-script: ### XR K9SEC INSTALL COMPLETE ###
Oct 11 11:06:03 172.30.0.54 ztp-script: ### Installing midnight commander ###
Oct 11 11:06:04 172.30.0.54 ztp-script: ### Downloading system configuration ###
Oct 11 11:06:05 172.30.0.54 ztp-script: ### Downloading system configuration complete ###
Oct 11 11:06:06 172.30.0.54 ztp-script: ### Applying initial system configuration ###
Oct 11 11:06:11 172.30.0.54 ztp-script: !!! Checking for errors !!!
Oct 11 11:06:14 172.30.0.54 ztp-script: ### Applying system configuration complete ###
Oct 11 11:06:15 172.30.0.54 ztp-script: Autoprovision complete...

```




CHAPTER 14

Upgrading Field-Programmable Device

An FPD is a field programmable logic device which contains non-volatile, re-programmable memory to define its internal wiring and functionality. The contents of this non-volatile memory are called the FPD image or FPD firmware. Over the lifespan of an FPD, FPD firmware images may need upgrades for bug fixes or functionality improvements. These upgrades are performed in the field with minimum system impact.

- [Prerequisites for FPD Image Upgrades, on page 233](#)
- [Overview of FPD Image Upgrade Support, on page 233](#)
- [FPD upgrade service, on page 233](#)

Prerequisites for FPD Image Upgrades

You must install the FPD pie before you install the SMUs or Service Packs. If you install the SMU or Service Packs before the FPD pie, the FPDs on the line card may not upgrade. In such cases, you must remove the SMUs and Service Packs and reload the router.

Overview of FPD Image Upgrade Support

An FPD image is used to upgrade the software on an FPD.

FPD versions must be compatible with the Cisco IOS XR software that is running on the router; if an incompatibility exists between an FPD version and the Cisco IOS XR software, the device with the FPGA may not operate properly until the incompatibility is resolved.



Note

- It is mandatory to upgrade all the required FPDs before doing a reload when you are upgrading FPDs on line cards. This is because, partial FPD component upgrades might result in booting errors (in some cases).
- You must not reload any line card or the router before all FPD image upgrades are completed successfully.

FPD upgrade service

The main tasks of the FPD upgrade service are:

- Check FPD image version to decide if a specific firmware image needs an upgrade or not.
- Manual FPD Image Upgrade using the **upgrade hw-module fpd** command.
- Invoke the appropriate device driver with a name of the new image to load.

An FPD image package is used to upgrade FPD images. The **install activate** command is used to place the FPD binary files into the expected location on the boot devices.

Supported Upgrade Methods

Method	Remarks
Manual Upgrade	Upgrade using CLI, force upgrade supported.

Determining Upgrade Requirement

Use the **show hw-module fpd** command to determine if an FPD upgrade is required. Check for NEED UPGD in the Status column.

Example

```
Router: #show hw - module fpd
```

```
Wed Dec 14 07:08:08.424 UTC
```

```
Auto-upgrade:Disabled
```

Location	Card type	HWver	FPD device	ATR Status	FPD Versions	
					Running	Programd
0/0	NC55-18H18F	1.0	MIFPGA	NEED UPGD	7.01	7.01
0/0	NC55-18H18F	1.0	Bootloader	CURRENT	1.14	1.14
0/0	NC55-18H18F	1.0	IOFPGA	CURRENT	0.07	0.07
0/0	NC55-18H18F	1.0	SATA-M600-MCT	CURRENT	0.23	0.23

Use the **show fpd package** command to find out which FPGAs are supported with your current software release and minimum hardware requirements for each module.

Manual FPD upgrade

Manual FPD upgrade is performed using the **upgrade hw-module fpd** command. All cards or all FPGA in a card can be upgraded. If reload is required to activate FPD, the upgrade should be complete. Line-cards, fabric cards and RP cards cannot be reloaded during the process of the FPD upgrade.

FPD upgrade is transaction-based:

- Each fpd upgrade CLI execution is one transaction.
- Only one transaction is allowed at any given time.
- One transaction may include one or many FPD upgrades

The **force** option can be used to forcibly upgrade the FPD (regardless of whether it is required or not). It triggers all FPDs to be upgraded or downgraded. The **force** option can also be used to downgrade or upgrade the FPGAs even after the version check.

**Note**

- Sometimes, FPDs can have primary and backup images.
- Force FPD upgrade with **upgrade hw-module location all fpd all force** command affects forwarding over BVI interface. You must reload involved locations to recover.
- The use of the **force** option when performing an FPD upgrade is not recommended except under explicit direction from Cisco engineering or TAC for a one-time purpose only.
- FPD upgrade should be performed in Admin mode only.
- A new FPD upgrade should be issued only when previous FPD upgrades have been completed on the same FPD with the following syslog message:

```
RP/0/RP0/CPU0:May 10 10:11:44.414 UTC: fpd-serv[205]: %INFRA-FPD_Manager-1-UPGRADE_ALERT
: FPD Upgrade Completed (use "show hw-module fpd" to check upgrade status)
```

How to Upgrade FPD Images

You must determine if an FPD image upgrade is needed using the **show hw-module fpd** command and perform the upgrade, if needed, under the following circumstances:

- Migrate the software to a later Cisco IOS XR software release.
- Swap line cards from a system running a different Cisco IOS XR software release.
- Insert a new line card.

In the event of an FPD incompatibility with your card, you might receive the following error message:

```
LC/0/0/CPU0:Jul 5 03:00:18.929 UTC: optics_driver[220]: %L2-OPTICS-3-BAD_FPGA_IMAGE :
Detected bad MI FPGA image programmed in MI FPGA SPI flash in 0/0/CPU0 location: Failed to
validate meta data CRC
LC/0/0/CPU0:Jul 5 03:00:19.019 UTC: optics_driver[220]: %L2-OPTICS-3-BACKUP_FPGA_LOADED :
Detected Backup FPGA image running on 0/0/CPU0 - primary image corrupted (@0x8c = 0x44)
RP/0/RP0/CPU0:Jul 5 03:00:48.987 UTC: fpd-serv[301]: %PKT_INFRA-FM-3-FAULT_MAJOR : ALARM_MAJOR
:FDP-NEED-UPGRADE :DECLARE :0/0:
```

Upgrades to the Cisco IOS XR software might result in an FPD incompatibility. Ensure that you perform the FPD upgrade procedure and resolve all incompatibilities, for the cards to function properly.

**Note**

The use of the **force** option when performing a FPD upgrade is not recommended except under explicit direction from Cisco engineering or TAC for a one-time purpose only.

Before you begin

- The FPD upgrade procedure is performed while the card is online. At the end of the procedure the card must be reloaded before the FPD upgrade is complete. To reload the card, you can use the **hw-module**

location <location> reload command in Admin mode, during the next maintenance window. The upgrade procedure is not complete until the card is reloaded.

- During the FPD upgrade, you *must not* do the following:
 - Reload, perform an online insertion and removal (OIR) of a line card (LC), or power down the chassis. Doing so may cause the node to enter an unusable state.
 - Press **Ctrl-C** if the console appears to hang without any output. Doing so may abort the upgrade.
- If you are not sure whether a card requires an FPD upgrade, you can install the card and use the **show hw-module fpd** command to determine if the FPD image on the card is compatible with the currently running Cisco IOS XR software release.

Procedure

	Command or Action	Purpose
Step 1	show hw-module fpd location {all node-id} Example: <pre>RP/0/RP0/CPU0:router# show hw-module fpd location all</pre> or <pre>RP/0/RP0/CPU0:router# show hw-module fpd location 0/4/cpu0</pre>	Displays the current FPD image versions for the specified card or all cards installed in the router. Use this command to determine if you must upgrade the FPD image on your card.
Step 2	admin Example: <pre>RP/0/RP0/CPU0:router# admin</pre>	Enters mode.
Step 3	(Optional) show fpd package Example:	Displays which cards are supported with your current Cisco IOS XR software release, which FPD image you need for each card, and what the minimum hardware requirements are for the various modules. (A minimum hardware requirement version of 0.0 indicates that all hardware can support this FPD image version.) If there are multiple FPD images for your card, use this command to determine which FPD image to use if you want to upgrade only a specific FPD type.
Step 4	upgrade hw-module fpd {all fpga-type} [force] location [all node-id] Example: <pre># upgrade hw-module fpd all location 0/3/1 . . .</pre>	Upgrades all the current FPD images that must be upgraded on the specified card with new images. Before continuing to the next step, wait for confirmation that the FPD upgrade has successfully completed. Status messages, similar to these, are displayed to the screen until the FPD upgrade is completed: <pre>FPD upgrade started.</pre>

	Command or Action	Purpose
	<p>Successfully upgraded 1 FPD for SPA-2XOC48POS/RPR</p> <p>on location 0/3/1</p>	<pre> FPD upgrade in progress.. FPD upgrade in progress.. FPD upgrade sent to location xxxx FPD upgrade sent to location yyyy FPD upgrade in progress.. FPD upgrade finished for location xxx FPD upgrade in progress.. FPD upgrade finished for location yyyy FPD upgrade completed. </pre> <p>The “FPD upgrade in progress.” message is printed every minute. These logs are information logs, and as such, are displayed if the logging console informational command is configured.</p> <p>If Ctrl-C is pressed while the FPD upgrade is in progress, the following warning message is displayed:</p> <pre> FPD upgrade in progress on some hardware, aborting now is not recommended as it might cause HW programming failure and result in RMA of the hardware. Do you want to continue? [Confirm(y/n)] </pre> <p>If you confirm that you want to abort the FPD upgrade procedure, this message is displayed:</p> <pre> FPD upgrade process has been aborted, please check the status of the hardware and reissue the upgrade command if required. </pre>

	Command or Action	Purpose
		<p>Note</p> <ul style="list-style-type: none"> • If your card supports multiple FPD images, you can use the show fpd package admin command to determine what specific image to upgrade in the upgrade hw-module fpd command. • A message is displayed when router modules cannot get upgraded during upgrade with location all option indicating that the FPGA is intentionally skipped during upgrade. To upgrade such FPGAs, you can use the CLI command with a particular location explicitly specified. For example, upgrade hw-module fpd all location 0/3/1. • It is recommended to upgrade all FPGAs on a given node using the upgrade hw-module fpd all location {all node-id} command. Do not upgrade the FPGA on a node using the upgrade hw-module fpd <individual-fpd> location {all node-id} as it may cause errors in booting the card.
Step 5	exit Example: sysadmin-vm:0_RP0# exit	
Step 6	hw-module location { node-id all } reload	Use the hw-module location reload command to reload a line card. sysadmin-vm:0_RP0# hw-module location 0/3 reload
Step 7	exit	
Step 8	show hw-module fpd	Verifies that the FPD image on the card has been successfully upgraded by displaying the status of all FPDs in the system.

Configuration Examples for FPD Image Upgrade

The following examples indicates the use of commands associated with the FPD image upgrade procedure.

show fpd package Command Output: Example

Use the **show fpd package** command in System Admin EXEC mode to find out which line cards are supported with your current Cisco IOS XR software release, which FPD image package you need for each line card, and what the minimum hardware requirements are for each module. If multiple FPD images are available for your card, they are listed as Subtype fpga2, fpga3, and so on.



Note The FPD name used in the FPD Description column of the output of the `show fpd package` command includes the last ten characters of DCO-PID. Depending on the slot and port numbers, the FPD name is appended with DCO_0, DCO_1, or DCO_2. For example, the FPD names for CFP2-WDM-D-1HL in port 0 and port 1 are -WDM-D-1HL_DCO_0 and WDM-D-1HL_DCO_1 respectively.



Note The FPD name used in the FPD Description column of the output of the `show fpd package` command displays QDD_instance_port-number. For example, depending on the instance and the port number, the FPD names for the QDD-400G-ZR-S and QDD-400G-ZRP-S modules will be QDD_0_3, QDD_1_0, and so on.

The following example shows sample output from the `show fpd package` command:

```
show fpd package
Tue Jan 22 13:56:00.212 UTC
```

Field Programmable Device Package					
Card Type	FPD Description	Req Reload	SW Ver	Min Req SW Ver	Min Req Board Ver
NC55-1200W-ACFW	LIT-PrimCU-ACFW (A)	NO	2.09	2.09	0.0
NC55-900W-ACFW-I	LIT-PrimCU-ACFW-I (A)	NO	1.04	1.04	0.0
NC55-900W-DCFW-I	LIT-PrimCU-DCFW-I (A)	NO	2.260	2.260	0.0
NC55-930W-DCFW-C	LIT-PrimCU-DCFW-C (A)	NO	2.259	2.259	0.0
NC55-MPA-12T-S	MPAFPGA	YES	0.27	0.27	0.0
NC55-MPA-1TH2H-S	-WDM-D-1HL_DCO_2	NO	38.518	38.518	0.1
	MPAFPGA	YES	0.53	0.53	0.0
	WDM-DE-1HL_DCO_2	NO	38.518	38.518	0.1
	WDM-DS-1HL_DCO_2	NO	38.268	38.268	0.1
NC55-MPA-2TH-HX-S	-WDM-D-1HL_DCO_0	NO	38.518	38.518	0.1
	-WDM-D-1HL_DCO_1	NO	38.518	38.518	0.1
	MPAFPGA	YES	0.53	0.53	0.0
	WDM-DE-1HL_DCO_0	NO	38.518	38.518	0.1
	WDM-DE-1HL_DCO_1	NO	38.518	38.518	0.1
	WDM-DS-1HL_DCO_0	NO	38.268	38.268	0.1
	WDM-DS-1HL_DCO_1	NO	38.268	38.268	0.1
NC55-MPA-2TH-S	-WDM-D-1HL_DCO_0	NO	38.518	38.518	0.1
	-WDM-D-1HL_DCO_1	NO	38.518	38.518	0.1
	MPAFPGA	YES	0.53	0.53	0.0
	WDM-DE-1HL_DCO_0	NO	38.518	38.518	0.1
	WDM-DE-1HL_DCO_1	NO	38.518	38.518	0.1
	WDM-DS-1HL_DCO_0	NO	38.268	38.268	0.1
	WDM-DS-1HL_DCO_1	NO	38.268	38.268	0.1
NC55-MPA-4H-HD-S	MPAFPGA	YES	0.53	0.53	0.0
NC55-MPA-4H-HX-S	MPAFPGA	YES	0.53	0.53	0.0

show fpd package Command Output: Example

NC55-MPA-4H-S	MPAFPGA	YES	0.53	0.53	0.0
NC55A2-MOD-SE-H-S	Bootloader (A)	YES	1.11	1.11	0.0
	CPU-IOFPGA (A)	YES	1.18	1.18	0.1
	MB-IOFPGA (A)	YES	0.18	0.18	0.1
	MB-MIFPGA	YES	0.19	0.19	0.0
	SATA (A)	NO	5.00	5.00	0.0
NCS-55A2-MOD-HD-S	Bootloader (A)	YES	1.11	1.11	0.0
	CPU-IOFPGA (A)	YES	1.18	1.18	0.1
	MB-IOFPGA (A)	YES	0.18	0.18	0.1
	MB-MIFPGA	YES	0.19	0.19	0.0
	SATA (A)	NO	5.00	5.00	0.0
NCS-55A2-MOD-HX-S	Bootloader (A)	YES	1.11	1.11	0.0
	CPU-IOFPGA (A)	YES	1.18	1.18	0.1
	MB-IOFPGA (A)	YES	0.18	0.18	0.1
	MB-MIFPGA	YES	0.19	0.19	0.0
	SATA (A)	NO	5.00	5.00	0.0
NCS-55A2-MOD-S	Bootloader (A)	YES	1.11	1.11	0.0
	CPU-IOFPGA (A)	YES	1.18	1.18	0.1
	MB-IOFPGA (A)	YES	0.18	0.18	0.1
	MB-MIFPGA	YES	0.19	0.19	0.0
	SATA (A)	NO	5.00	5.00	0.0
NCS-55A2-MOD-SE-S	Bootloader (A)	YES	1.11	1.11	0.0
	CPU-IOFPGA (A)	YES	1.18	1.18	0.1
	MB-IOFPGA (A)	YES	0.18	0.18	0.1
	MB-MIFPGA	YES	0.19	0.19	0.0
	SATA (A)	NO	5.00	5.00	0.0
	STATSFPGA	YES	0.01	0.01	0.0

This table describes the significant fields shown in the display:

Table 15: show fpd package Field Descriptions

Field	Description
Card Type	Module part number.
FPD Description	Description of all FPD images available for the line card.
Type	Hardware type. Possible types can be: <ul style="list-style-type: none"> • spa—Shared port adapter • lc—Line card
Subtype	FPD subtype. These values are used in the upgrade hw-module fpd command to indicate a specific FPD image type to upgrade.
SW Version	FPD software version recommended for the associated module running the current Cisco IOS XR software.
Min Req SW Vers	Minimum required FPD image software version to operate the card. Version 0.0 indicates that a minimum required image was not programmed into the card.

Field	Description
Min Req HW Vers	Minimum required hardware version for the associated FPD image. A minimum hardware requirement of version 0.0 indicates that all hardware can support this FPD image version.

upgrade hw-module fpd Command Output: Example

Use the **upgrade hw-module fpd** command to upgrade the FPD image on a line card. The upgrade can be executed for all FPDs or for specific FPDs that need an upgrade. To upgrade all FPDs, use **upgrade hw-module fpd all location all** command. To upgrade a specific FPD image type, use the FPD subtype value in the **upgrade hw-module fpd** command.

show platform Command Output: Example

Use the **show platform** command to verify that the line card is up and running.

Auto FPD Upgrade

Table 16: Feature History Table

Feature Name	Release Information	Feature Description
Auto FPD Upgrade	Release 7.3.2	This functionality enables automatic upgrade and reload for field-programmable devices (FPDs) whenever the Cisco IOS XR image has a newer FPD version. This functionality upgrades all route processors and line card FPDs simultaneously while displaying upgrade triggers on the console.

Effective Cisco IOS XR Release 7.3.2, you can enable automatic upgrade of FPD by using the “**fpd auto-upgrade enable**” command.

To automatically upgrade all FPDs, use:

```
RP/0/RP0/CPU0:IOS(config)#fpd auto-upgrade enable
```

To reload the interface modules following the fpd auto-upgrade, use:

```
RP/0/RP0/CPU0:IOS(config)#fpd auto-reload enable
```

Limitations and Usage Guidelines

Limitations

- FPD auto-upgrade should be enabled only in the XR VM and *not* in the System Admin VM.
- With auto-upgrade enabled, if any card is in RELOAD REQUIRED state, auto-upgrade is re-triggered during any SSO or FPD-serv process restart.

- When an interface module (IM) or route processor (RP) is in RELOAD REQUIRED state and auto-upgrade is enabled, FPD upgrades are triggered again.
- With auto-upgrade enabled, if line card is inserted, an auto-upgrade is triggered. During this phase optics alarms are generated. If auto-reload is not enabled, you must reload the line cards manually to clear these alarms.
- SATA allows you to upgrade or downgrade when an FPD version change is available. Therefore, when auto-upgrade is enabled, the system automatically downgrades if lower versions are available. This behavior is specific only to SATA FPDs.
- FPD auto-reload is applicable for line cards only. Line cards are automatically reloaded after the fpd auto-upgrade process is completed.
- Cisco NCS 5500 Series Routers do not support ISSU.
- TimingICs do not support **auto fpd upgrade** on NCS5500 Series Routers as the TimingIC requires a card reload immediately after upgrade. For the same reason, the TimingICs are not upgraded if the user specifies **location all** in the **auto fpd upgrade** command. To upgrade a TimingIC FPD, specify the FPD name along with the card location. For example, **upgrade hw-module fpd TimngIC-A location 0/RP0/cpu0**.

Usage Guidelines—Online Insertion of Line Cards

When a line card with a lower FPD version is inserted, one of the following scenarios apply:

- If fpd auto-upgrade and auto-reload are enabled, and a new line card is inserted, the system upgrades the line card FPDs automatically with the latest FPDs and reloads the line cards.
- If fpd auto-upgrade and auto-reload are both disabled, no action is required.
- If fpd auto-upgrade is enabled and auto-reload is disabled, the following alarms are displayed on the console:

```
RP/0/RP1/CPU0:Jun 1 10:05:46.095 UTC: optics_driver[231]: %PKT_INFRA-FM-3-FAULT_MAJOR
: ALARM_MAJOR :OPTICS SUPPORTED_ERROR :DECLARE : Optics0/5/0/6: Optics0/5/0/6
RP/0/RP1/CPU0:Jun 1 10:05:46.096 UTC: optics_driver[231]: %PKT_INFRA-FM-2-FAULT_CRITICAL
: ALARM_CRITICAL :OPTICS NOT SUPPORTED :DECLARE : Optics0/5/0/6: Optics0/5/0/6
```

You must reload the line cards manually to clear these alarms

Usage Guidelines—Online Insertion of RPs

When fpd auto-upgrade is enabled and a new RP is inserted, the system upgrades the RP FPDs automatically with the latest FPDs.



Note RPs are not reloaded automatically. You must manually reload the RP or chassis for the latest FPD version to reflect.



Note Reload of active RPs and line cards impacts the network traffic.

Table 17: Action Required on FPDs After Auto Upgrade

FPD	Action Required
IOFPGA	Manual reload required
ADM	Upgraded version available immediately
PRIMARY-BIOS	Manual reload required
SATA	Upgraded version available immediately
PSOC	Upgraded version available immediately
IMFPGA	Manual reload required, if auto-reload is not configured

Automatic FPD Upgrade for PSU

During the installation and Power Supply Unit (PSU) insertion process, the Field-Programmable Devices (FPD) associated with the PSUs are automatically upgraded.



Note The PSUs are upgraded sequentially, hence the PSU FPD upgrades take longer. You can choose to exclude PSUs from the auto upgrade flow. This restricts the PSUs from being upgraded either upon insertion, or during system upgrade.

To exclude the PSU FPDs from auto upgrading, use the following CLI:

fpd auto-upgrade exclude pm

```
RP/0/RSP0/CPU0:router# show running-config fpd auto-upgrade
Wed Mar 30 20:52:55.079 UTC
fpd auto-upgrade enable
fpd auto-upgrade exclude pm
```



Note When you upgrade from an earlier unsupported version to a version that supports Automatic FPD upgrade for PSU, the PSU upgrade might happen on bootup.

Upgrade Failure

On failure of an FPD upgrade, you get a warning with the following syslog message:

```
LC/0/5/CPU0:Jun 27 05:02:25.742 UTC: optics_driver[216]: %INFRA-FPD_Driver-1-UPGRADE_ALERT
: FPD MIFPGA@0/5 image programming completed with UPGD FAIL state Info: [Image verification
failed at offset 0x5c8, flash value = 0x0, image value = 0x40, image size = 4194304]
LC/0/5/CPU0:Jun 27 05:02:26.570 UTC: optics_driver[216]: %INFRA-FPD_Driver-1-UPGRADE_ALERT
: FPD MIFPGA@0/5 image programming completed with UPGD FAIL state Info: [Image verification
failed at offset 0x1e, flash value = 0x56, image value = 0xff, image size = 4194304]
```

When you use the **show hw-module fpd** command, the status column displays **UPGD FAIL** to indicate failure of the FPD upgrade.

**Note**

-
- Do not reload the line card with a failed FPD upgrade image.
 - Upgrade failed FPDs will be fixed with a manual upgrade.
 - Contact Cisco TAC or your account representative if the FPD upgrade failure is not repaired.
-



CHAPTER 15

Y.1564 - Ethernet Service Activation Test

- [Y.1564 - Ethernet Service Activation Test, on page 245](#)
- [CoS DEI Support Matrix, on page 247](#)
- [Configuration Examples, on page 248](#)
- [Verification, on page 248](#)

Y.1564 - Ethernet Service Activation Test

With the deployment of Ethernet in service provider networks, ethernet services have evolved significantly. Not only is ethernet found at the User Network Interface (UNI) but can also be deployed anywhere in the network, creating a Network-to-Network Interface (NNI). With the capability to prioritize traffic, high availability, and its built-in resiliency, service providers are now using this technology to deliver advanced services. In the absence of any standardized test methodologies that can measure delay, jitter, loss, and throughput at a port, the ITU-T recommendation Y.1564 addresses the gap

Y.1564 - Ethernet Service Activation Test (or performance test methodology) is a testing procedure which tests service turn-up, installation, and troubleshooting of Ethernet-based services. This test methodology was created to have a standard way of measuring Ethernet-based services in the industry.

Cisco implementation of ITU-T Y.1564 has three key objectives:

- To serve as a network SLA validation tool, ensuring that a service meets its guaranteed performance settings in a controlled test time.
- To ensure that all services carried by the network meet their SLA objectives at their maximum committed rate, thus proving that under maximum load, network devices and paths can support all traffic as designed.
- To perform medium-term and long-term service testing, confirming that network elements can properly carry all services while under stress during a soaking period.

The following Key Performance Indicators (KPI) metrics are collected to ensure that the configured SLAs are met for the service or stream.

- Frame Transfer Delay (FTD) or latency—Measures the round-trip time (RTT) taken by a test frame to travel through a network device, or across the network and back to the test port.
- Frame Loss Ratio (FLR)—Measures the number of packets lost from the total number of packets sent. Frame loss can be due to a number of issues such as network congestion or errors during transmissions.

Supported Modes

The mode of operation that is supported for Y.1564 is the Two-way statistics collection mode. In the two-way mode, the sender generates the test traffic used to perform the test, which is then looped back by the remote node. The statistics are measured and collected locally on the sender

The following encapsulations are supported by Y.1564 SADT feature:

- dot1q
- dot1q + second dot1q
- dot1ad
- dot1ad + second dot1q
- priority tagged
- untagged



Note *Default encapsulation* is supported if there are no other sub-interfaces configured with *untagged encapsulation*.

Usage Guideline and Limitations

- Rewrite with POP option is supported with Color Blind mode with Outer-Cos value of 0.
- Rewrite Push and Translate on Encapsulation Untagged is not supported.
- Y.1564 doesn't support L1 loopback.
- Y.1564 doesn't support measuring and analyzing jitter.
- When utilizing the SAT engine Rx_bytes statistics feature, there can be potential inaccuracies in the following conditions:
 - During tests incorporating EMIX sequences that encounter packet drops.
 - When handling LMM packets originating from TGEN, Y.1731 protocols, or any unidentified sources.
- SAT supports a scale of four parallel sessions per system. However, all four sessions can not operate as color aware sessions simultaneously due to limitations in Class of Service (CoS) combinations.
- SAT over bundle interface functions by selecting one of its members for transmission. Therefore, at least one member must be in the 'UP' state to initiate an SAT session.
- For optimal performance, it's recommended to use the Ethernet Data Plane Loopback functionality (EDPL) on the peer side for SAT. EDPL loops back and swaps the MAC addresses of Layer 2 packets generated. If the peer node doesn't support EDPL functionality, you can configure SAT to generate Layer 2 packets with the destination MAC address equal to the source MAC address. In such cases, the peer can perform an L1 loopback.

CoS DEI Support Matrix

Table 18: Class of Service (CoS) and Discard Eligibility Identifier (DEI) Matrix

		CoS							
		0	1	2	3	4	5	6	7
DEI	0	Ö	Ö	Ö	Ö	Ö	X	Ö	X
	1	Ö	Ö	X	X	X	X	X	X

Ö - Supported

X- Not Supported

Supported Bandwidth Parameters

Bandwidth Parameters	Internal Direction	External Direction
Committed Information Rate	Y	Y
Exceeded Information Rate	Y	Y

Service Activation Test Target Matrix

Target	Internal Direction	External Direction
L2 Interface over physical main/sub interfaces	Y	Y
L2 Interface over bundle main/sub interfaces	Y	Y
L2 PW VPWS over physical main/sub interfaces	Y	Y
L2 PW VPWS over bundle main/sub interfaces	Y	Y
L2 EVPN/XConnect over physical main/sub interfaces	Y	Y
L2 EVPN/XConnect over bundle main/sub interfaces	Y	Y
L2 VPLS PW	N	N
L2 EVPN Bridge-Domain	N	N

Target	Internal Direction	External Direction
L3 Interfaces	N	N

Configuration Examples

The following example shows how to start service-activation test on an interface with external direction:

```
RP/0/RP0/CPU0:router ethernet service-activation-test start interface TenGigE 10/0/0/1
profile test destination 00ab.6009.9c3c direction external
```

The following example shows how to start service-activation test on an interface with internal direction:

```
RP/0/RP0/CPU0:router ethernet service-activation-test start interface TenGigE 10/0/0/1
profile test destination 00ab.6009.9c3c direction internal
```

The following example shows how to stop service-activation-test on an interface:

```
RP/0/RP0/CPU0:router ethernet service-activation-test stop interface TenGigE 10/0/0/1
```

The following example shows how to stop all service-activation-tests:

```
RP/0/RP0/CPU0:router ethernet service-activation-test stop all
```

Configuring Ethernet Service Activation Test Color Profile

The following example shows how to configure a color-blind profile for ethernet service activation test:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)# ethernet service-activation-test
RP/0/RP0/CPU0:router(config-ethsat)#profile sattest1
RP/0/RP0/CPU0:router(config-ethsat-prf)#outer-cos 4
RP/0/RP0/CPU0:router(config-ethsat-prf)#duration 8 minutes
RP/0/RP0/CPU0:router(config-ethsat-prf)#information-rate 11800 mbps
RP/0/RP0/CPU0:router(config-ethsat-prf)#packet-size 1000
```

The following example shows how to configure a color-aware profile for ethernet service activation test:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)# ethernet service-activation-test
RP/0/RP0/CPU0:router(config-ethsat)#profile sattest3
RP/0/RP0/CPU0:router(config-ethsat-prf)#outer-cos 4
RP/0/RP0/CPU0:router(config-ethsat-prf)#duration 1 minutes
RP/0/RP0/CPU0:router(config-ethsat-prf)#color-aware cir 7 gbps eir-color cos 1
RP/0/RP0/CPU0:router(config-ethsat-prf)#information-rate 8 gbps
RP/0/RP0/CPU0:router(config-ethsat-prf)#packet-size 1000
```

Verification

To verify the interfaces on which Y.1564 (ethernet service activation test) is enabled, use the **show ethernet service-activation-test brief** command. The following is a sample output of an enabled device:

```
RP/0/RP0/CPU0:router#show ethernet service-activation-test brief
Interface TenGigE0/0/0/14.1
  Service activation tests permitted (external only)
  Test completed:
    Duration 1 minute(s)
```

```
Information rate 1 Gbps
Color-aware, CIR: 900 Mbps, EIR: DEI set, CoS 1
External, Two-way, Destination 00:11:00:22:00:33
Packet size EMIX, Sequence 'abceg', Pattern hex 0x00
Packet format: SAT MEF-49
Outer CoS 2
```

Results:

```
Step 1, Information Rate 1 Gbps
CIR packets:
Tx packets: 11276845, bytes: 6743553310
Rx packets: 11276845, bytes: 6743553310
FL: 0, FLR: 0%
FD: Min 5.144us, Mean 6.681us, Max 9.576us
IFDV: Not supported
Out of order packets: 0 (0%)
Error packets: 0 (0%)

EIR packets:
Tx packets: 1252895, bytes: 749231210
Rx packets: 1252895, bytes: 749231210
FL: 0, FLR: 0%
FD: Min 5.160us, Mean 6.679us, Max 8.800us
IFDV: Not supported
Out of order packets: 0 (0%)
Error packets: 0 (0%)
```




CHAPTER 16

Configuration and File System Management

This module describes methods for configuration management and file transfer enhancements.

- [Auto-Save Configuration, on page 251](#)
- [Increasing Commit Limit, on page 252](#)

Auto-Save Configuration

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.

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
```

When filename is accessed through VRF, you can specify filename in **filename** **<protocol>://<user>@<host>:<port>;<vrf name>/<url-path>/<file-name>** format.

Increasing Commit Limit

Table 19: Feature History Table

Feature Name	Release Information	Feature Description
Increasing Commit Limit	Release 7.3.6	<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 cfs check 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 clear configuration ascii inconsistency command, which performs an ASCII backup after 55 minutes.</p> <p>The feature modifies the following commands:</p> <ul style="list-style-type: none"> • cfs check • clear configuration ascii inconsistency

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.

In rebase, the router automatically saves your changes to the backup 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 up to 40 seconds to complete 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 7.3.6, 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 commits count and configuration data size using the **show cfmgr commitdb** command. If the commits 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 operation only after 55 minutes since the last commit was performed after the ASCII backup.

Guidelines and Restrictions for Increasing the Commit Limit

- The **cfs check** command increases the rebase limits only for one instance. After executing the **cfs check** command, the router will perform a rebase operation after 40 commits or when the configuration data reaches 4 MB. Once the router performs a rebase operation, the limits will reset to the default values of 20 commits and 2 MB configuration data. To enable 40 commits and 4 MB configuration data, you must perform the **cfs check** command again.
- After executing the **cfs check** command, if a router switches over to standby RP, the rebase limits are retained as 40 commits and configuration data of 4 MB. However, if the router reloads, the rebase limits are reset to 20 commits and 2 MB of configuration data. For example, after executing the **cfs check** command, if the router switches over to standby RP after 30 commits, it will still have ten more commits before a rebase. However, if the router reloads, the rebase limits are reset to default 20 commits and 2 MB of configuration data.
- 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 begin the ASCII backup operation only 55 minutes after the last commit performed after the ASCII backup. 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 first commit operation. 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.

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 Stamp
~~~~  ~~~~~      ~~~~                               ~~~~~~
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
```




CHAPTER 17

Synchronous Ethernet ESMC and SSM

Synchronous Ethernet is an extension of Ethernet designed to provide the reliability found in traditional SONET/SDH and T1/E1 networks to Ethernet packet networks by incorporating clock synchronization features. It supports the Synchronization Status Message (SSM) and Ethernet Synchronization Message Channel (ESMC) for synchronous Ethernet clock synchronization.

Synchronous Ethernet incorporates the Synchronization Status Message (SSM) used in Synchronous Optical Networking (SONET) and Synchronous Digital Hierarchy (SDH) networks. While SONET and SDH transmit the SSM in a fixed location within the frame, Ethernet Synchronization Message Channel (ESMC) transmits the SSM using a protocol: the IEEE 802.3 Organization-Specific Slow Protocol (OSSP) standard.

The ESMC carries a Quality Level (QL) value identifying the clock quality of a given synchronous Ethernet timing source. Clock quality values help a synchronous Ethernet node derive timing from the most reliable source and prevent timing loops.

When configured to use synchronous Ethernet, the router synchronizes to the best available clock source. If no better clock sources are available, the router remains synchronized to the current clock source.

The router supports QL-enabled mode.

- [Frequency Synchronization Timing Concepts, on page 255](#)
- [SyncE Hardware Support Matrix, on page 257](#)
- [Configuring Frequency Synchronization, on page 259](#)
- [Verifying the Frequency Synchronization Configuration, on page 260](#)
- [Verifying the ESMC Configuration, on page 263](#)
- [Verifying Controllers Timing LEDs, on page 264](#)

Frequency Synchronization Timing Concepts

The Cisco IOS XR frequency synchronization infrastructure is used to select between different frequency sources to set the router backplane frequency and time-of-day. There are two important concepts that must be understood with respect to the frequency synchronization implementation.

Sources

A source is a piece of hardware that inputs frequency signals into the system or transmits them out of the system. There are four types of sources:

- Line interfaces. This includes SyncE interfaces.

- Clock interfaces. These are external connectors for connecting other timing signals, such as, GPS, BITS.
- PTP clock. If IEEE 1588 version 2 is configured on the router, a PTP clock may be available to frequency synchronization as a source of the time-of-day and frequency.
- Internal oscillator. This is a free-running internal oscillator chip.

Each timing source has a Quality Level (QL) associated with it which gives the accuracy of the clock. This QL information is transmitted across the network via SSMs over the Ethernet Synchronization Messaging Channel (ESMC) or SSMs contained in the SONET/SDH frames so that devices know the best available source to synchronize to. In order to define a preferred network synchronization flow, and to help prevent timing loops, you can assign priority values to particular timing sources on each router. The combination of QL information and user-assigned priority levels allows each router to choose a timing source to use to clock its SyncE and SONET/SDH interfaces, as described in the ITU standard G.781.

Selection Points

A selection point is any point where a choice is made between several frequency signals, and possibly one or more of them are selected. Selection points form a graph representing the flow of timing signals between the different cards in a router running Cisco IOS XR software. For example, one or multiple selection points select between the different Synchronous Ethernet inputs available on a single line card, and the result of these selection points is forwarded to a selection point on the RSP to select between the selected source from each card.

The input signals to the selection points can be:

- Received directly from a source.
- The output from another selection point on the same card.
- The output from a selection point on a different card.

The output of a selection point can be used in a number of ways:

- Used to drive the signals sent out of a set of sources.
- As input into another selection point on the card.
- As input into a selection point on another card.

Use the `show frequency synchronization selection` command to see a detailed view of the different selection points within the system.



Note

- We recommend you to configure, and enable Frequency Synchronization selection input on two interfaces per line card.
- For link aggregation, you must configure and enable Frequency Synchronization selection input on a single bundle member.

Restrictions

- SyncE is not supported on Gigabit Ethernet 0/0/0/24 to 0/0/0/31 ports.
- The Precision Time Protocol (PTP) session flaps during Route Processor Failover (RPFO).

SyncE Hardware Support Matrix



Note The table also contains support details of upcoming releases. You can read this table in context of the current release and see relevant *Release Notes* for more information on supported features and hardware.

This table provides a detailed information on the timing features that are supported on the following hardware variants.

Hardware Variant	Features	Cisco IOS XR Release	Comments
NCS-57C3-MODS-SYS NCS-57C3-MODS-SYS	E-SyncE	Release 7.9.1	
NC57-24DD	SyncE	Release 7.5.1	
NC57-18DD-SE	SyncE	Release 7.5.1	
NCS-57C1-48Q6-SYS	SyncE	Release 7.5.1	
	E-SyncE	Release 7.9.1	
RP:NC57-MOD-RP-2E with NCS573-MODS-SYS and NCS-573-MOD-SYS	SyncE	Release 7.4.1	1G clock recovery is supported from IOS XR Release 7.6.1.
NCS-57B1-5DSE-SYS NCS-57B1-6D24-SYS	SyncE	Release 7.3.1	
	E-SyncE	Release 7.9.1	
RP: NC55-RP2-E Line card: NC57-36H6D-S	SyncE	Release 7.3.2	<ul style="list-style-type: none"> • Release 7.3.2 - Supports Compatible Mode only • Release 7.7.1 - Supports both Native and Compatible mode. • SyncE is not supported on 100GE interfaces, when they are used in 1G mode.

Hardware Variant	Features	Cisco IOS XR Release	Comments
RP:NC55-RP-E with Line cards: NC55-MOD-A-S and NC55-32T16Q4H-AT	SyncE	Release 7.1.1	SyncE is not supported on 25GE or 100GE interfaces, when they are used in 1G mode.
RP:NC55-RP2-E with Line cards: NC55-MOD-A-S and NC55-32T16Q4H-AT	SyncE	Release 7.1.1	SyncE is not supported on 25GE or 100GE interfaces, when they are used in 1G mode.
RP:NC55-RP2-E with Line card:NC55-32T16Q4H-AT	SyncE	Release 7.1.1	SyncE is not supported on 25GE or 100GE interfaces, when they are used in 1G mode.
NCS-55A1-36H-SE-S	SyncE	Release 7.0.1	SyncE is not supported on 25GE or 100GE interfaces, when they are used in 1G mode.
NCS-55A1-36H-S	SyncE	Release 7.0.1	SyncE is not supported on 25GE or 100GE interfaces, when they are used in 1G mode.
NCS-55A1-24Q6H-S NCS-55A1-24Q6H-SS	SyncE	Release 6.6.25	SyncE is not supported on 25GE or 100GE interfaces, when they are used in 1G mode.
NCS-55A1-48Q6H	SyncE	Release 6.6.25	SyncE is not supported on 25GE or 100GE interfaces, when they are used in 1G mode.
NCS-55A1-24H	SyncE	Release 6.5.2	SyncE is not supported on 25GE or 100GE interfaces, when they are used in 1G mode.
NCS55A2-MOD	SyncE	Release 6.5.1	SyncE is not supported on 25GE or 100GE interfaces, when they are used in 1G mode.
	E-syncE	Release 7.9.1	E-SyncE is not supported on 25GE or 100GE interfaces, when they are used in 1G mode.

Hardware Variant	Features	Cisco IOS XR Release	Comments
RP:NC55-RP-E Linecard:NC55-MOD-A-S	SyncE	Release 6.5.1	SyncE is not supported on 100GE interfaces, when they are used in 1G mode.
RP:NC55-RP-E Linecard:NC55-36X100G-A-SE	SyncE	Release 6.3.2	SyncE is not supported on 25GE or 100GE interfaces, when they are used in 1G mode.
NCS5501-SE	SyncE	Release 6.3.2	SyncE is not supported on 25GE or 100GE interfaces, when they are used in 1G mode. SyncE is supported on 10G from ports 8 to 15, but it is not supported on these ports in 1G mode.

SyncE features are supported on the following MPAs:

- NC55-MPA-2TH-S
- NC55-MPA-1TH2H-S
- NC55-MPA-1TH2H-HD-S
- NC55-MPA-4H-S
- NC55-MPA-4H-HD-S
- NC55-MPA-12T-S

Configuring Frequency Synchronization

Enabling Frequency Synchronization on the Router

This task describes the router-level configuration required to enable frequency synchronization.

```
RP/0/RP0/CPU0:Router# configure
RP/0/RP0/CPU0:Router(config)# frequency synchronization
RP/0/RP0/CPU0:Router(config-freqsync)# clock-interface timing-mode system
RP/0/RP0/CPU0:Router(config-freqsync)# quality itu-t option 1 generation 1
RP/0/RP0/CPU0:Router(config-freqsync)# log selection changes
RP/0/RP0/CPU0:Router(config-freqsync)# commit
```

Configuring Frequency Synchronization on an Interface

By default, there is no frequency synchronization on line interfaces. Use this task to configure an interface to participate in frequency synchronization.

Before You Begin

You must enable frequency synchronization globally on the router.

```
RP/0/RP0/CPU0:R1#config terminal
RP/0/RP0/CPU0:R1(config)#interface TenGigabitEthernet 0/0/0/0
RP/0/RP0/CPU0:R1(config-if)#frequency synchronization
RP/0/RP0/CPU0:R1(config-if-freqsync)#selection input
RP/0/RP0/CPU0:R1(config-if-freqsync)#wait-to-restore 10
RP/0/RP0/CPU0:R1(config-if-freqsync)#priority 5
RP/0/RP0/CPU0:R1(config-if-freqsync)#quality transmit exact itu-t option 1 PRC
RP/0/RP0/CPU0:R1(config-if-freqsync)#quality receive exact itu-t option 1 PRC
RP/0/RP0/CPU0:R1(config-if-freqsync)#commit
or
RP/0/RP0/CPU0:router(config-freqsync)# commit
```

Configuring Frequency Synchronization on a Clock Interface

To enable a clock interface to be used as frequency input or output, you must configure the port parameters and frequency synchronization, as described in this task.

```
RP/0/RP0/CPU0:R1#configure
RP/0/RP0/CPU0:R1(config)# clock-interface sync 2 location 0/RP0/CPU0
RP/0/RP0/CPU0:R1(config-clock-if)# port-parameters
RP/0/RP0/CPU0:R1(config-clk-parms)# gps-input tod-format cisco pps-input ttl
RP/0/RP0/CPU0:R1(config-clk-parms)# exit
RP/0/RP0/CPU0:R1(config-clock-if)# frequency synchronization
RP/0/RP0/CPU0:R1(config-clk-freqsync)# selection input
RP/0/RP0/CPU0:R1(config-clk-freqsync)# wait-to-restore 1
RP/0/RP0/CPU0:R1(config-clk-freqsync)# quality receive exact itu-t option 1 PRC
```

Verifying the Frequency Synchronization Configuration

After performing the frequency synchronization configuration tasks, use this task to check for configuration errors and verify the configuration.

1. show frequency synchronization selection

```
RP/0/RP0/CPU0:R5# show frequency synchronization selection
Fri Apr 24 12:49:32.833 UTC
Node 0/RP1/CPU0:
=====
Selection point: T0-SEL-B (3 inputs, 1 selected)
  Last programmed 3d04h ago, and selection made 3d04h 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
4	HundredGigE0/7/0/0	0/RP1/CPU0 ETH_RXMUX 4	PRC	10	Locked
	PTP [0/RP1/CPU0]	n/a	PRC	254	Available
	Internal0 [0/RP1/CPU0]	n/a	SEC	255	Available

```

Selection point: 1588-SEL (2 inputs, 1 selected)
  Last programmed 3d04h ago, and selection made 3d04h 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
== =====
4  HundredGigE0/7/0/0    0/RP1/CPU0 ETH_RXMUX 4          PRC  10  Locked
   Internal0 [0/RP1/CPU0] n/a                      SEC 255 Available

Selection point: CHASSIS-TOD-SEL (2 inputs, 1 selected)
Last programmed 3d04h ago, and selection made 3d04h 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/RP1/CPU0]      n/a                      100  Yes   Available
   HundredGigE0/7/0/0    0/RP1/CPU0 T0-SEL-B 4      100  No    Available

Selection point: ETH_RXMUX (1 inputs, 1 selected)
Last programmed 3d04h ago, and selection made 3d04h ago
Next selection points
SPA scoped      : None
Node scoped     : T0-SEL-B 1588-SEL
Chassis scoped: None
Router scoped   : None
Uses frequency selection
S  Input                Last Selection Point          QL  Pri  Status
== =====
4  HundredGigE0/7/0/0    n/a                      PRC  10  Available

```

2. show frequency synchronization configuration-errors

```

RP/0/RP0/CPU0:router# show frequency synchronization configuration-errors
Node 0/2/CPU0:
=====
interface GigabitEthernet0/2/0/0 frequency synchronization
* Frequency synchronization is enabled on this interface, but isn't enabled globally.
interface GigabitEthernet0/2/0/0 frequency synchronization quality transmit exact itu-t
option 2 generation 1 PRS
* The QL that is configured is from a different QL option set than is configured
globally.

```

Displays any errors that are caused by inconsistencies between shared-plane (global) and local-plane (interface) configurations. There are two possible errors that can be displayed:

- Frequency Synchronization is configured on an interface (line interface or clock-interface), but is not configured globally.
- The QL option configured on some interface does not match the global QL option. Under an interface (line interface or clock interface), the QL option is specified using the quality transmit and quality receive commands. The value specified must match the value configured in the global quality itu-t option command, or match the default (option 1) if the global quality itu-t option command is not configured.

Once all the errors have been resolved, meaning there is no output from the command, continue to the next step.

3. show frequency synchronization interfaces brief

```
RP/0/RP0/CPU0:R5# show frequency synchronization interfaces brief
Flags: > - Up
      d - SSM Disabled
      s - Output squelched
Fl   Interface
D - Down      S - Assigned for selection
  x - Peer timed out  i - Init state
Last Selection Point
Pri Time
Status
=====
>S  TenGigE0/0/0/0      PRC   PRC    1 DNU   TenGigE0/0/0/0
>x  TenGigE0/0/0/1      Fail  n/a   100 PRC   TenGigE0/0/0/0
>x  TwentyFiveGigE0/0/0/30 Fail  n/a   100 PRC   TenGigE0/0/0/0

RP/0/RP0/CPU0:R5#
```

Verifies the configuration. Note the following points:

- All line interface that have frequency synchronization configured are displayed.
- All clock interfaces and internal oscillators are displayed.
- Sources that have been nominated as inputs (in other words, have selection input configured) have 'S' in the Flags column; sources that have not been nominated as inputs do not have 'S' displayed.



Note Internal oscillators are always eligible as inputs.

- '>' or 'D' is displayed in the flags field as appropriate.

If any of these items are not true, continue to the next step.

4. show processes fsyncmgr location node-id

This command verifies that the fsyncmgr process is running on the appropriate nodes.

```
RP/0/RP0/CPU0:R5# show processes fsyncmgr location 0/0/cPU0
Thu Feb 1 06:26:32.979 UTC
Job Id: 181
PID: HYPERLINK "tel:3411"3411
Process name: fsyncmgr
Executable path:
/opt/cisco/XR/packages/ncs540-iosxr-fwding-1.0.0.0-r63226I/all/bin/fsyncmgr Instance #:
1
Version ID: 00.00.0000
Respawn: ON
Respawn count: 1
Last started: Tue Jan 23 04:26:57 HYPERLINK "tel:2018"2018
Process state: Run
Package state: Normal
core: MAINMEM
Max. core: 0
Level: 100
Placement: None
startup_path:
/opt/cisco/XR/packages/ncs540-iosxr-fwding-1.0.0.0-r63226I/all/startup/fsyncmgr.startup
Ready: 2.063s
Process cpu time: 168.480 user, 129.980 kernel, 298.460 total
```



```

JID TID Stack pri state NAME rt_pri
181 HYPERLINK "tel:3411"3411 OK 20 Sleeping fsyncmgr 0
181 HYPERLINK "tel:3572"3572 OK 20 Sleeping lwm_debug_threa 0
181 HYPERLINK "tel:3573"3573 OK 20 Sleeping fsyncmgr 0
181 HYPERLINK "tel:3574"3574 OK 20 Sleeping lwm_service_thr 0
181 HYPERLINK "tel:3575"3575 OK 20 Sleeping qsm_service_thr 0
181 HYPERLINK "tel:3622"3622 OK 20 Sleeping fsyncmgr 0
181 HYPERLINK "tel:3781"3781 OK 20 Sleeping fsyncmgr 0
181 HYPERLINK "tel:3789"3789 OK 20 Sleeping fsyncmgr 0

```

Verifying the ESMC Configuration

show frequency synchronization interfaces

```

Router# show frequency synchronization interfaces
      Interface TenGigE0/0/0/0 (up)
      Assigned as input for selection
      Wait-to-restore time 0 minutes
      SSM Enabled
Peer Up for 2d01h, last SSM received 0.320s ago
Peer has come up 1 times and timed out 0 times
ESMC SSMs Total Information Event DNU/DUS
Sent: HYPERLINK "tel:178479"178479 HYPERLINK "tel:178477"178477 2 HYPERLINK "tel:178463"178463
0
      Received: HYPERLINK "tel:178499"178499 HYPERLINK "tel:178499"178499 0
      Input:
      Up
      Last received QL: Opt-I/PRC
      Effective QL: Opt-I/PRC, Priority: 1, Time-of-day Priority 100
      Supports frequency
      Output:
      Selected source: TenGigE0/0/0/0
      Selected source QL: Opt-I/PRC
      Effective QL: DNU
      Next selection points: ETH_RXMUX
      Interface TenGigE0/0/0/1 (up)
      Wait-to-restore time 5 minutes
      SSM Enabled
      Peer Timed Out for 2d01h, last SSM received never
      Peer has come up 0 times and timed out 1 times
      ESMC SSMs Total Information Event DNU/DUS
      Sent: HYPERLINK "tel:178479"178479 HYPERLINK "tel:178477"178477 2 0
      Received: 0 0 0
      Input:
      Down - not assigned for selection
      Supports frequency
      Output:
      Selected source: TenGigE0/0/0/0
      Selected source QL: Opt-I/PRC
      Effective QL: Opt-I/PRC
      Next selection points: ETH_RXMUX
      Interface TwentyFiveGigE0/0/0/30 (up)
      Wait-to-restore time 5 minutes
      SSM Enabled
      Peer Timed Out for 01:50:24, last SSM received 01:50:30 ago
      Peer has come up 1 times and timed out 1 times
ESMC SSMs Total Information Event DNU/DUS
Sent: HYPERLINK "tel:75086"75086 HYPERLINK "tel:75085"75085 1 0
Received: HYPERLINK "tel:68457"68457 HYPERLINK "tel:68455"68455 2 HYPERLINK "tel:68443"68443
Input:

```

```
Down - not assigned for selection
Supports frequency
Output:
Selected source: TenGigE0/0/0/0
Selected source QL: Opt-I/PRC
Effective QL: Opt-I/PRC
Next selection points: ETH_RXMUX
```

Verifying Controllers Timing LEDs

```
Router# show controllers timing led status location 0/RP0/CPU0
LED Status:
  BITS0: Off
  BITS1: Off
  Sync: Green
  GNSS: Off
  GPS: NA
```



INDEX

A

access-group command [128](#)
 admin configure command [236](#)
 admin-config submode, *See* admin configure command
 authenticate command [129–130](#)
 authentication-key command [129–130](#)

B

broadcast client command [126–127](#)
 broadcast command [126–127](#)
 broadcastdelay command [126](#)

C

CDP [65, 67, 69](#)
 enabling [67](#)
 functional overview [65](#)
 monitoring [69](#)
 cdp (global) command [67](#)
 cdp (interface) command [67](#)
 cdp advertise v1 command [68](#)
 cdp holdtime command [68](#)
 cdp timer command [68–69](#)

F

FPD images [233, 236](#)
 displaying [236](#)
 default information [236](#)
 minimum and current versions [236](#)
 overview for SPAs [233](#)
 upgrade [236](#)
 FPGA [233](#)
 devices, methods of upgrading [233](#)

H

hw-module reload command [238](#)

L

line console command [10](#)

line template command [11–12](#)

M

master command [134](#)
 MIB, description [18](#)

N

NTP [123, 125, 127, 129, 131, 134, 136](#)
 configuring an authoritative NTP server [134](#)
 configuring broadcast-based NTP associations [125](#)
 configuring NTP access groups [127](#)
 configuring NTP authentication [129](#)
 configuring poll-based associations [123](#)
 disabling NTP services on an interface [131](#)
 updating the hardware clock [136](#)

P

physical terminals [7–8](#)
 console line template [7](#)
 description [7](#)
 default line template [7](#)
 description [7](#)
 line template guidelines [8](#)

S

sample output [238, 241](#)
 server command [124](#)
 show cdp command [68–69](#)
 show cdp entry command [70](#)
 show cdp interface command [70](#)
 show cdp neighbors command [70](#)
 show cdp traffic command [70](#)
 show fpd package command [236, 238](#)
 show hw-module fpd command [236](#)
 show line command [13](#)
 show platform command [241](#)
 show snmp command [26–27](#)
 show snmp engineid command [26–27](#)
 show snmp group command [26, 28](#)
 show snmp users command [26, 28](#)

show snmp view command [26, 28](#)
 show terminal command [13–14](#)
 show users command [13–14](#)
 SNMP (Simple Network Management Protocol) [18, 22, 26](#)
 agent, description [18](#)
 manager, description [18](#)
 MIB, description [18](#)
 versions [22, 26](#)
 SNMPv3 benefits [22](#)
 SNMPv3 costs [22](#)
 SNMPv3, configuring [26](#)
 snmp-server chassis-id command [36–37](#)
 snmp-server contact command [36–37](#)
 snmp-server engineid local command [26](#)
 snmp-server group command [26–27](#)
 snmp-server ipv4 dscp command [40](#)
 snmp-server ipv4 precedence command [40](#)
 snmp-server location command [36–37](#)
 snmp-server packetsize command [38](#)
 snmp-server queue-length command [38–39](#)
 snmp-server trap source command [38–39](#)
 snmp-server trap-timeout command [39](#)
 snmp-server user command [26–27](#)
 snmp-server view command [26–27](#)
 snmp-server vrf vrf-name command [26](#)
 source command [133](#)

SPA (shared port adapter) [233, 235](#)
 <I_Italic>See also <Default Para Font>FPD
 images<\$nopage> [233](#)
 See also [FPD images](#)

T

trap notifications [24](#)
 trusted-key command [129–130](#)

U

update-calendar command [136](#)
 upgrade hw-module fpd command [236, 241](#)

V

virtual terminals [7–9, 11](#)
 default line template [7](#)
 description [7](#)
 user-defined line templates [8](#)
 description [8](#)
 vty pools [9, 11](#)
 creating [11](#)
 description [9](#)
 modifying [11](#)