



Interface and Hardware Component Configuration Guide for Cisco NCS 5500 Series Routers, IOS XR Release 7.5.x

First Published: 2021-11-01

Last Modified: 2023-03-30

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 xv

Changes to This Document xv

Obtaining Documentation and Submitting a Service Request xv

CHAPTER 1

New and Changed Feature Information 1

Interface and Hardware Component Features Added or Modified in IOS XR Release 7.5.x 1

CHAPTER 2

Preconfiguring Physical Interfaces 3

Physical Interface Preconfiguration Overview 4

Prerequisites for Preconfiguring Physical Interfaces 4

Benefits of Interface Preconfiguration 4

How to Preconfigure Physical Interfaces 5

Information About Preconfiguring Physical Interfaces 6

Use of the Interface Preconfigure Command 10

CHAPTER 3

Advanced Configuration and Modification of the Management Ethernet Interface 13

Prerequisites for Configuring Management Ethernet Interfaces 13

How to Perform Advanced Management Ethernet Interface Configuration 14

Configuring a Management Ethernet Interface 14

IPv6 Stateless Address Auto Configuration on Management Interface 17

Modifying the MAC Address for a Management Ethernet Interface 19

Verifying Management Ethernet Interface Configuration 20

Information About Configuring Management Ethernet Interfaces 21

CHAPTER 4

Configuring Ethernet Interfaces 23

Configuring Gigabit Ethernet Interfaces 25

Information About Configuring Ethernet	29
Default Configuration Values for 100-Gigabit Ethernet	29
Network Interface Speed	30
Configuring Network Interface Speed	30
Using the speed command	30
Using the negotiation auto command	32
Using speed and negotiation auto command	34
Ethernet MTU	36
Link Layer Discovery Protocol (LLDP)	37
Specifying User-Defined LLDP TLV Values	38
Enabling LLDP Globally	39
Enabling LLDP Per Interface	40
Dense Wavelength Division Multiplexing Tunable Optics	41
Configuring the DWDM Tunable Optics	45
Priority Flow Control (PFC)	53
Restrictions for PFC	54
Configuring Priority Flow Control	54
Optical Transport Networks	57
Restrictions and Important Guidelines	57
OTN Architecture	58
Configuring OTN Interface	59
How to Configure Interfaces in Breakout Mode	61
Information About Breakout	61
Configure Breakout in a Port	61
Remove the Breakout Configuration	62
Verify a Breakout Configuration	62

CHAPTER 5

Configuring Ethernet OAM	63
Ethernet OAM	63
Ethernet Link OAM	64
Neighbor Discovery	64
EFD	64
MIB Retrieval	65
Miswiring Detection (Cisco-Proprietary)	65

SNMP Traps	65
How to Configure Ethernet OAM	66
Configuring Ethernet Link OAM	66
Configuration Examples for Ethernet OAM	75
Configuration Examples for EOAM Interfaces	75
Ethernet CFM	77
Maintenance Domains	79
Services	81
Maintenance Points	81
MIP Creation	81
MEP and CFM Processing Overview	82
CFM Protocol Messages	84
Continuity Check (IEEE 802.1ag and ITU-T Y.1731)	84
Maintenance Association Identifier (MAID)	85
Loopback (IEEE 802.1ag and ITU-T Y.1731)	88
Linktrace (IEEE 802.1ag and ITU-T Y.1731)	89
Configurable Logging	91
Flexible VLAN Tagging for CFM	91
Configuring Ethernet CFM	92
Configuring a CFM Maintenance Domain	93
Configuring Services for a CFM Maintenance Domain	94
Enabling and Configuring Continuity Check for a CFM Service	96
Configuring Automatic MIP Creation for a CFM Service	97
Configuring Cross-Check on a MEP for a CFM Service	99
Configuring Other Options for a CFM Service	101
Configuring CFM MEPs	103
Configuring Y.1731 AIS	105
Configuring AIS in a CFM Domain Service	105
Configuring AIS on a CFM Interface	107
Configuring Flexible VLAN Tagging for CFM	108
Verifying the CFM Configuration	109
Troubleshooting Tips	110
Configuration Examples for Ethernet CFM	111
Ethernet CFM Domain Configuration: Example	111

- Ethernet CFM Service Configuration: Example 111
- Flexible Tagging for an Ethernet CFM Service Configuration: Example 111
- Continuity Check for an Ethernet CFM Service Configuration: Example 111
- MIP Creation for an Ethernet CFM Service Configuration: Example 111
- Cross-check for an Ethernet CFM Service Configuration: Example 112
- Other Ethernet CFM Service Parameter Configuration: Example 112
- MEP Configuration: Example 112
- Ethernet CFM Show Command: Examples 112
- Ethernet CFM Show Command for flexible packet format: Examples 115
- AIS for CFM Configuration: Examples 119
- AIS for CFM Show Commands: Examples 120
- show ethernet cfm interfaces ais Command: Example 120
- show ethernet cfm local meps Command: Examples 120
- show ethernet cfm local meps detail Command: Example 122
- CFM Adaptive Bandwidth Notifications 122
 - Bandwidth Notification Messages 123
 - Restrictions for CFM Bandwidth Notifications 124
 - Bandwidth Reporting 124
 - Damping Algorithm 125
 - Conformance Testing Algorithm 126
 - Embedded Event Manager 127
 - Event Publishing 128
 - Configure CFM Bandwidth Notifications 128
- CFM Over Bundles 130
- Y.1731 Performance Monitoring 131
 - Two-Way Delay Measurement for Scalability 132
 - Configuring Two-Way Delay Measurement 132
 - Synthetic Loss Measurement 135
 - Configuring Synthetic Loss Measurement 136
- Unidirectional Link Detection Protocol 141
 - Types of Fault Detection 141
 - UDLD Modes of Operation 142
 - Configure UDLD 142
- Link Loss Forwarding 145

Restrictions for LLF	145
Configure Link Loss Forwarding	146

CHAPTER 6**Configuring Integrated Routing and Bridging 149**

IRB Introduction	149
Bridge-Group Virtual Interface	150
Supported Features on a BVI	150
BVI Interface and Line Protocol States	151
Prerequisites for Configuring IRB	151
Restrictions for Configuring IRB	152
How to Configure IRB	153
Configuring the Bridge Group Virtual Interface	153
Configuration Guidelines	153
Configuring the Layer 2 AC Interfaces	155
Configuring a Bridge Group and Assigning Interfaces to a Bridge Domain	156
Associating the BVI as the Routed Interface on a Bridge Domain	157
Displaying Information About a BVI	159
Additional Information on IRB	159
Packet Flows Using IRB	159
Packet Flows When Host A Sends to Host B on the Bridge Domain	160
Packet Flows When Host A Sends to Host C From the Bridge Domain to a Routed Interface	160
Packet Flows When Host C Sends to Host B From a Routed Interface to the Bridge Domain	161
Configuration Examples for IRB	161
Basic IRB Configuration: Example	161
IPv4 Addressing on a BVI Supporting Multiple IP Networks: Example	162
IRB With BVI and VRRP Configuration: Example	162

CHAPTER 7**Configuring Link Bundling 165**

Limitations and Compatible Characteristics of Ethernet Link Bundles	166
Configuring Ethernet Link Bundles	167
Configuring LACP Fallback	172
VLANs on an Ethernet Link Bundle	173
Configuring VLAN over Bundles	174

174

LACP Short Period Time Intervals	178
Configuring the Default LACP Short Period Time Interval	178
Configuring Custom LACP Short Period Time Intervals	180
Bundle Consistency Checker	186
Information About Configuring Link Bundling	190
IEEE 802.3ad Standard	190
Link Bundle Configuration Overview	191
Link Switchover	191
LACP Fallback	192
<hr/>	
CHAPTER 8	Configuring Traffic Mirroring 193
Introduction to Traffic Mirroring	194
Traffic Mirroring Terminology	194
Traffic Mirroring Types	195
Characteristics of Source Port	195
Characteristics of Destination Port	195
Characteristics of Monitor Session	196
Supported Scale	196
Restrictions	196
SPAN Types, Supported Features, and Configurations	200
Local SPAN	200
Remote SPAN	201
Configure Remote Traffic Mirroring	201
SPAN on Subinterfaces	204
Monitoring Traffic Mirroring on a Layer 2 Interface	204
ACL-based SPAN	204
Configuring Security ACLs for Traffic Mirroring	205
Configuring UDF-Based Security ACL for Traffic Mirroring	205
DSCP Bitmask to Filter Ingress SPAN Traffic	207
Multiple SPAN ACL Sessions in Single Interface	209
Monitor Multiple SPAN ACL and Security ACL Sessions	212
Attaching the Configurable Source Interface	213
ERSPAN	214
Introduction to ERSPAN Egress Rate Limit	215

ERSPAN Traffic to a Destination Tunnel in a Non-Default VRF	218
DSCP Marking on Egress GRE Tunnel in ERSPAN	219
SPAN over Pseudowire	221
Configure SPAN over Pseudowire	221
Verify SPAN over Pseudowire	221
SPAN-to-File	223
SPAN-to-File Enhancements	223
File Mirroring	226
Configure File Mirroring	227
Forward-Drop Packets Mirroring	228
Mirror Forward-Drop Packets	228
Troubleshoot Traffic Mirroring	229

CHAPTER 9	Configuring Virtual Loopback and Null Interfaces	233
	Information About Configuring Virtual Interfaces	233
	Virtual Loopback Interface Overview	233
	Prerequisites for Configuring Virtual Interfaces	234
	Configuring Virtual Loopback Interfaces	234
	Null Interface Overview	236
	Configuring Null Interfaces	236
	Configuring Virtual IPv4 Interfaces	238

CHAPTER 10	Configuring GRE Tunnels	241
	Configuring GRE Tunnels	241
	Single Pass GRE Encapsulation Allowing Line Rate Encapsulation	243
	Configure GRE Single-Pass Entropy	244
	Running Configuration	247
	Verification	250

CHAPTER 11	Configuring IP-in-IP Tunnels	255
	IP-in-IP Decapsulation	258
	Decapsulation Using Tunnel Source Direct	261
	Guidelines and Limitations	262
	Configure Decapsulation Using Tunnel Source Direct	262

CHAPTER 12	Understand Generic UDP Encapsulation	265
	Restrictions	266
	Configure GUE	267
	Flexible Assignment of UDP Port Numbers for Decapsulation	269
	Guidelines for Setting up Decapsulation Using Flexible Port Numbers	269
	Outer-Header Hashing Support for IPoGREoGUE and MPLSoGREoUDP Flows	270
	Running Configuration	271
	Verification	272

CHAPTER 13	Configuring 400G Digital Coherent Optics	273
	Configuring Frequency	276
	Configuring Chromatic Dispersion	278
	Configuring Optical Transmit Power	280
	Configuring Muxponder Mode	283
	Configuring Modulation	286
	Configuring DAC Rate	288
	Configuring FEC	289
	Configuring Loopback	291
	Configuring Performance Monitoring	292
	Configuring PM Parameters	292
	Configuring Alarms Threshold	295
	Alarms Troubleshooting	297
	CD Alarm	297
	Clear the CD Alarm	297
	DGD Alarm	297
	Clear the DGD Alarm	298
	FLEXO_LOF	298
	Clear the FLEXO_LOF Alarm	298
	FLEXO_LOM	298
	Clear the FLEXO_LOM Alarm	298
	HI-LASERBIAS Alarm	298
	Clear the HI-LASERBIAS Alarm	299
	HI-RXPOWER Alarm	299

Clear the HI-RXPOWER Alarm	299
HI-RXPOWER Warn	299
Clear the HI-RXPOWER Warn Alarm	299
HI-TEMP Alarm	300
Clear the HI-TEMP Alarm	300
HI-TEMP Warn	300
Clear the HI-TEMP Warn Alarm	300
HI-TXPOWER Alarm	300
Clear the HI-TXPOWER Alarm	301
HI-TXPOWER Warn	301
Clear the HI-TXPOWER Warn Alarm	301
IMPROPER-REM	301
Clear the IMPROPER-REM Alarm	301
LOF	302
Clear the LOF Alarm	302
LOL	302
Clear the LOL Alarm	302
LOM	302
Clear the LOM Alarm	302
LO-RXPOWER Alarm	303
Clear the LO-RXPOWER Alarm	303
LO-RXPOWER Warn	303
Clear the LO-RXPOWER Warn Alarm	303
LOS	303
Clear the LOS Alarm	304
LOS-P	304
Clear the LOS-P Alarm	304
LO-TXPOWER Alarm	304
Clear the LO-TXPOWER Alarm	304
LO-TXPOWER Warn	304
Clear the LO-TXPOWER Warn Alarm	305
OOR_CD	305
Clear the OOR_CD Alarm	305
OSNR Alarm	305

Clear the OSNR Alarm	305
UNC-WORD Alarm	306
Clear the UNC-WORD Alarm	306
WVL-OOL	306
Clear the WVL-OOL Alarm	306

CHAPTER 14

Configuring Controllers	307
Optics Controllers	307
Maintenance Mode	308
Performance Monitoring	309
How to Configure Controllers	309
Configuring Optics Controller	309
Restrictions and Usage Guidelines for Port Modes	311
Configure Port Mode Speed	312
Configuring Wavelength	317
Configuring Coherent DSP Controller	319
Configuring Performance Monitoring	320
Verify Controller Details	321
Replace Optical Module	322

CHAPTER 15

Global Navigation Satellite System	325
Configuring the Global Navigation Satellite System	326
Information About GNSS	326
Overview of GNSS	326
Operation of GNSS Module	327
Prerequisites for GNSS	328
Restrictions for GNSS	328
Configure GNSS	329

CHAPTER 16

Configuring WAN-PHY Controllers	331
WAN-PHY Controller	331
Restrictions	332
Configuring SONET Mode on an Interface	332
Configuring SDH Mode on an Interface	334

TSoP Smart SFP for SDH and SONET Encapsulation	337
Prerequisites for TSoP	338
Restrictions for TSoP	338
Guidelines for TSoP Smart SFP	338
De-jitter Buffer	339
Configuration for TSoP	340



Preface

The *Interface and Hardware Component Configuration Guide for Cisco NCS 5500 Series Routers* provides information and procedures related to router interface and hardware configuration.

The preface contains the following sections:

- [Changes to This Document, on page xv](#)
- [Obtaining Documentation and Submitting a Service Request, on page xv](#)

Changes to This Document

This table lists the technical changes made to this document since it was first released.

Table 1: Changes to This Document

Date	Summary
March 2023	Republished for Release 7.5.4
September 2022	Republished for Release 7.5.3
November 2021	Initial release of this document.

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at: <http://www.cisco.com/c/en/us/td/docs/general/whatsnew/whatsnew.html>

Subscribe to *What's New in Cisco Product Documentation*, which lists all new and revised Cisco technical documentation, as an RSS feed and deliver content directly to your desktop using a reader application. The RSS feeds are a free service.



CHAPTER 1

New and Changed Feature Information

This table summarizes the new and changed feature information for the *Interface and Hardware Component Configuration Guide for Cisco NCS 5500 Series Routers*, and tells you where they are documented.

- [Interface and Hardware Component Features Added or Modified in IOS XR Release 7.5.x](#), on page 1

Interface and Hardware Component Features Added or Modified in IOS XR Release 7.5.x

Table 2: New and Changed Features

Feature	Description	Introduced in Release	Where Documented
Specifying User-Defined LLDP TLV Values	This feature is introduced.	Release 7.5.3	Specifying User-Defined LLDP TLV Values , on page 38
ERSPAN Traffic to a Destination Tunnel in a Non-Default VRF	This feature is introduced.	Release 7.5.3	ERSPAN Traffic to a Destination Tunnel in a Non-Default VRF , on page 218
Outer-Header Hashing Support	This feature is introduced.	Release 7.5.3	Outer-Header Hashing Support for IPoGREoGUE and MPLSoGREoUDP Flows , on page 270
String-based MAID support for Offloaded Endpoints	This feature is introduced.	Release 7.5.1	Continuity Check (IEEE 802.1ag and ITU-T Y.1731) , on page 84
Extending GNSS Functionality to Cisco Network Convergence System 5700 Series variants	This feature is introduced.	Release 7.5.1	Configuring the Global Navigation Satellite System , on page 326

Feature	Description	Introduced in Release	Where Documented
OTN Support for NC55-MPA-12T-S MPA on Cisco NCS 5500 Series Routers.	This feature is introduced.	Release 7.5.1	.
Support for Link Loss Forwarding (LLF)	This feature is introduced.	Release 7.5.1	Link Loss Forwarding
Mirror first option support	This feature is introduced.	Release 7.5.2	SPAN-to-File Enhancements



CHAPTER 2

Preconfiguring Physical Interfaces

This module describes the preconfiguration of physical interfaces.

Preconfiguration is supported for these types of interfaces and controllers:

- 100-Gigabit Ethernet
- Management Ethernet

Preconfiguration allows you to configure line cards before they are inserted into the router. When the cards are inserted, they are instantly configured. The preconfiguration information is created in a different system database tree, rather than with the regularly configured interfaces. That database tree is known as the *preconfiguration directory* on the route processor.

There may be some preconfiguration data that cannot be verified unless the line card is present, because the verifiers themselves run only on the line card. Such preconfiguration data is verified when the line card is inserted and the verifiers are initiated. A configuration is rejected if errors are found when the configuration is copied from the preconfiguration area to the active area.



Note One Gigabit Ethernet interface is not supported. Only physical interfaces can be preconfigured.



Note Eight quadrature amplitude modulation (8QAM) requires V2 (or higher) CFP2 version and 5.23 (or higher) firmware.



Note From Cisco IOS XR Release 6.3.2, a six-seconds delay is introduced in error propagation from the driver to DPA for the MACSec line card and Oldcastle platforms. As a result, the BER algorithm on these platforms knows the error with a delay of 6 seconds.

- [Physical Interface Preconfiguration Overview](#), on page 4
- [Prerequisites for Preconfiguring Physical Interfaces](#), on page 4
- [Benefits of Interface Preconfiguration](#), on page 4
- [How to Preconfigure Physical Interfaces](#), on page 5
- [Information About Preconfiguring Physical Interfaces](#), on page 6

Physical Interface Preconfiguration Overview

Preconfiguration is the process of configuring interfaces before they are present in the system. Preconfigured interfaces are not verified or applied until the actual interface with the matching location (rack/slot/module) is inserted into the router. When the anticipated line card is inserted and the interfaces are created, the precreated configuration information is verified and, if successful, immediately applied to the running configuration of the router.



Note When you plug the anticipated line card in, make sure to verify any preconfiguration with the appropriate **show** commands.

Use the **show run** command to see interfaces that are in the preconfigured state.



Note We recommend filling out preconfiguration information in your site planning guide, so that you can compare that anticipated configuration with the actual preconfigured interfaces when that line card is installed and the interfaces are up.



Tip Use the **commit best-effort** command to save the preconfiguration to the running configuration file. The **commit best-effort** command merges the target configuration with the running configuration and commits only valid configuration (best effort). Some configuration might fail due to semantic errors, but the valid configuration still comes up.

Prerequisites for Preconfiguring Physical Interfaces

Before preconfiguring physical interfaces, ensure that this condition is met:

- Preconfiguration drivers and files are installed. Although it may be possible to preconfigure physical interfaces without a preconfiguration driver installed, the preconfiguration files are required to set the interface definition file on the router that supplies the strings for valid interface names.

Benefits of Interface Preconfiguration

Preconfigurations reduce downtime when you add new cards to the system. With preconfiguration, the new cards can be instantly configured and actively running during cards bootup.

Another advantage of performing a preconfiguration is that during a cards replacement, when the cards is removed, you can still see the previous configuration and make modifications.

How to Preconfigure Physical Interfaces

This task describes only the most basic preconfiguration of an interface.

SUMMARY STEPS

1. **configure**
2. **interface preconfigure** *type interface-path-id*
3. Use one of the following commands:
 - **ipv4 address** *ip-address subnet-mask*
 - **ipv4 address** *ip-address /prefix*
4. Configure additional interface parameters, as described in this manual in the configuration chapter that applies to the type of interface that you are configuring.
5. **end** or **commit** best-effort
6. **show running-config**

DETAILED STEPS

Step 1 **configure**

Example:

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

Enters global configuration mode.

Step 2 **interface preconfigure** *type interface-path-id*

Example:

```
RP/0/RP0/CPU0:router(config)# interface preconfigure HundredGigE 0/3/0/2
```

Enters interface preconfiguration mode for an interface, where *type* specifies the supported interface type that you want to configure and *interface-path-id* specifies the location where the interface will be located in *rack/slot/module/port* notation.

Step 3 Use one of the following commands:

- **ipv4 address** *ip-address subnet-mask*
- **ipv4 address** *ip-address /prefix*

Example:

```
RP/0/RP0/CPU0:router(config-if-pre)# ipv4 address 192.168.1.2/31
```

Assigns an IP address and mask to the interface.

Step 4 Configure additional interface parameters, as described in this manual in the configuration chapter that applies to the type of interface that you are configuring.

Step 5 **end** or **commit** best-effort

Example:

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

or

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

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes: `Uncommitted changes found, commit them before exiting (yes/no/cancel)?`
- 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 best-effort** command to save the configuration changes to the running configuration file and remain within the configuration session. The **commit best-effort** command merges the target configuration with the running configuration and commits only valid changes (best effort). Some configuration changes might fail due to semantic errors.

Step 6 **show running-config****Example:**

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

(Optional) Displays the configuration information currently running on the router.

Example

This example shows how to preconfigure a basic Ethernet interface:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface preconfigure HundredGigE 0/3/0/24
RP/0/RP0/CPU0:router(config-if)# ipv4 address 192.168.1.2/31
RP/0/RP0/CPU0:router(config-if-pre)# commit
```

Information About Preconfiguring Physical Interfaces

From Cisco IOS XR Release 7.0.2, the NC57-18DD-SE follows the following port mapping:

- Port number 0-17 (nine pairs) and 24-29 (three pairs): They together drive 400G mode. This means that if the top port is in 400G mode, the bottom port is unusable. These ports are retimer ports.

- Port number 18-23 (six ports): They are direct connected ports and are individually capable of 400G mode.



Note There's a limitation for ports 0, 1 and 14, 15. You have to insert modules of similar speed (40G or 100G) into these pairs of ports. For example, if you insert 40G module in port 0, then 40G module must be inserted in port 1.



Note For 400G-only mode, the ports to be used are 0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 19, 20, 21, 22, 23, 24, 26, and 28.

For detailed information on port mapping and usage, see the figure *NC57-18DD-SE Line Card* in chapter *NCS 5500 Series Modular Router Overview of Hardware Installation Guide for Cisco NCS 5500 Series Modular Routers* guide.

To control the interfaces which are created, use the `hw-module port-range mode` command with the following modes:

- 40-100: This is the default port mode. Two ports are created in 100G mode by default. Online Insertion and Removal (OIR) to 40G creates the 40G port.
- 400: The first port created is 400G. No port is created for the bottom port.
- 2x100: For 2x100 mode. This supports QDD-2X100-LR4 optics.

Port range can be in the form of n to $n+1$. Example: 0,1 or 6,7. The port range is valid for ports 0-17 and 24-29. To configure a port with 400G rate:

```
RP/0/RP0/CPU0:router(config)#hw-module port-range 0 1 location 0/3/CPU0 mode 400
RP/0/RP0/CPU0:router(config)#commit
Wed Feb  6 03:23:12.923 UTC
LC/0/3/CPU0:Feb  6 03:23:13.548 UTC: ifmgr[281]: %PKT_INFRA-LINK-3-UPDOWN : Interface
HundredGigE0/3/0/1, changed state to Down
LC/0/3/CPU0:Feb  6 03:23:13.548 UTC: ifmgr[281]: %PKT_INFRA-LINK-3-UPDOWN : Interface
HundredGigE0/3/0/0, changed state to Down
RP/0/RP0/CPU0:router(config)#end
RP/0/RP0/CPU0:router#show ipv4 int br location 0/3/CPU0
Wed Feb  6 03:26:07.935 UTC
```

Interface	IP-Address	Status	Protocol	Vrf-Name
FourHundredGigE0/3/0/0	unassigned	Shutdown	Down	default
HundredGigE0/3/0/2	unassigned	Shutdown	Down	default
HundredGigE0/3/0/3	unassigned	Shutdown	Down	default
HundredGigE0/3/0/4	unassigned	Shutdown	Down	default
HundredGigE0/3/0/5	unassigned	Shutdown	Down	default
HundredGigE0/3/0/6	unassigned	Shutdown	Down	default

To change a port mode:

```
RP/0/RP0/CPU0:router#conf
Thu Jan  9 05:13:02.853 UTC
RP/0/RP0/CPU0:router(config)#hw-module port-range 2 3 location 0/3/CPU0 mode 2x100
RP/0/RP0/CPU0:router(config)#commit
Thu Jan  9 05:13:11.411 UTC
LC/0/3/CPU0:Jan  9 05:13:11.469 UTC: optics_driver[196]: %PKT_INFRA-FM-3-FAULT_MAJOR :
ALARM_MAJOR :PORTMODE SPEED MISMATCH :CLEAR :0/3/CPU0: Optics0/3/0/3
```

```
LC/0/3/CPU0:Jan  9 05:13:13.141 UTC: ifmgr[228]: %PKT_INFRA-LINK-3-UPDOWN : Interface
HundredGigE0/3/0/3, changed state to Down
LC/0/3/CPU0:Jan  9 05:13:13.141 UTC: ifmgr[228]: %PKT_INFRA-LINK-3-UPDOWN : Interface
HundredGigE0/3/0/2, changed state to Down
RP/0/RP0/CPU0:router(config)#end
RP/0/RP0/CPU0:router#show ipv4 int br location 0/3/CPU0
Thu Jan  9 05:13:24.245 UTC
```

Interface	IP-Address	Status	Protocol	Vrf-Name
FortyGigE0/3/0/28	unassigned	Shutdown	Down	default
HundredGigE0/3/0/29	unassigned	Shutdown	Down	default
HundredGigE0/3/0/2/0	unassigned	Down	Down	default
HundredGigE0/3/0/2/1	unassigned	Down	Down	default
HundredGigE0/3/0/3/0	unassigned	Down	Down	default
HundredGigE0/3/0/3/1	unassigned	Down	Down	default

Use the following commands for the newly configured image:

```
hw-module port-range 0 1 location 0/6/CPU0 mode 400
hw-module port-range 2 3 location 0/6/CPU0 mode 400
hw-module port-range 4 5 location 0/6/CPU0 mode 400
hw-module port-range 6 7 location 0/6/CPU0 mode 400
hw-module port-range 8 9 location 0/6/CPU0 mode 400
hw-module port-range 10 11 location 0/6/CPU0 mode 400
hw-module port-range 12 13 location 0/6/CPU0 mode 400
hw-module port-range 14 15 location 0/6/CPU0 mode 400
hw-module port-range 16 17 location 0/6/CPU0 mode 400
hw-module port-range 24 25 location 0/6/CPU0 mode 400
hw-module port-range 26 27 location 0/6/CPU0 mode 400
hw-module port-range 28 29 location 0/6/CPU0 mode 400
hw-module port-range 0 1 location 0/6/CPU0 mode 2x100
hw-module port-range 2 3 location 0/6/CPU0 mode 2x100
hw-module port-range 4 5 location 0/6/CPU0 mode 2x100
hw-module port-range 6 7 location 0/6/CPU0 mode 2x100
hw-module port-range 8 9 location 0/6/CPU0 mode 2x100
hw-module port-range 10 11 location 0/6/CPU0 mode 2x100
hw-module port-range 12 13 location 0/6/CPU0 mode 2x100
hw-module port-range 14 15 location 0/6/CPU0 mode 2x100
hw-module port-range 16 17 location 0/6/CPU0 mode 2x100
hw-module port-range 24 25 location 0/6/CPU0 mode 2x100
hw-module port-range 26 27 location 0/6/CPU0 mode 2x100
hw-module port-range 28 29 location 0/6/CPU0 mode 2x100
```

NCS-57B1-6D24-SYS and NCS-57B1-5DSE-SYS have the following port-mapping characteristics:

- All ports use QSFP-DD; however, the first 24 ports are 100G, and the last few ports (six for non-SE and five for SE) are 400G.
- The two port types available are "400G Direct" port and "Quad Port Group".
- Each port type supports one or more speeds and breakout modes, such as 400G/4x100G/2x100G/8x50G/100G/4x25G/40G/4x10G/10G.



Note For specific transceiver support check the [optics compatibility matrix](#).

400G Direct Ports

Ports 24-29 for non-SE and ports 24-28 for SE are 400G direct ports, which support 400G individually.

Table 3: Possible Optics and Breakout for 400G Direct Ports

Optics	Breakout
QSFP-DD 400G	4x100G, 8x50G
QSFP56 200G	2x100G, 4x50G
QSFP28-DD 2x100G	2x(4x25G)
QSFP28 100G	4x25G
QSFP+ 40G	4x10G

Configuration Examples

The following are some configuration examples. All possible combinations are not listed here.

```
Router(config)#controller optics 0/0/0/n breakout 4x100
```

Result: Hu0/0/0/n/0-3

```
Router(config)#controller optics 0/0/0/n breakout 8x50
```

Result: Fi0/0/0/n/0-7

```
Router(config)#controller optics 0/0/0/n breakout 2x100
```

Result: Hu0/0/0/n/0-1

```
Router(config)#controller optics 0/0/0/n breakout 4x25
```

Result: TF0/0/0/n/0-3

```
Router(config)#controller optics 0/0/0/n breakout 4x10
```

Result: Te0/0/0/n/0-3

Quad Port Groups

Quad port groups have the following characteristics:

- There are six quad port groups of 4 QSFP-DD: (0,3), (4-7), (8-11), (12-15), (16-19), and (20-23).
- Each group shares 400G.
- Each port supports any combination of 40G/100G optics by default for a total of 400G per group.
- Each group has two port pairs, for example (0,1) and (2,3) for group (0,3).
- Breakout is only supported on the even (top) port of a port pair. The odd (bottom) port is automatically disabled. The odd port should be empty.
- QSA is supported only on 100G QSFP-DD ports, not on 400G QSFP-DD ports.
- Only 10G SFP+ optics are supported. There is no support for 1G.
- Linear optics not supported.
- For combinations with other optics type, see Quad Port Group table below, and consider 10G as one of the 40G optics.
- 4x25G breakout cannot co-exist with 40G or 4x10G breakout in the same port group.

Table 4: Possible Optics and Breakout for Quad Port Groups

Port N	Port N+1	Port N+2	Port N+3
100G or 40G	100G or 40G	100G or 40G	100G or 40G
4x10G	Disabled	4x10G	Disabled
4x10G	Disabled	100G or 40G	100G or 40G
100G or 40G	100G or 40G	4x10G	Disabled
4x25G	Disabled	4x25G	Disabled
4x25G	Disabled	100G	100G
100G	100G	4x25G	Disabled

Configuration Examples

The following are some configuration examples. All possible combinations are not listed here.

1. Quad Port Group in 4x10G breakout mode

```
Router(config)#hw-module port-range n n+1 location 0/RP0/CPU0 mode 4x10
```

Results:

- Te0/0/0/n/0-3: Port n+1 will be automatically disabled.
- Fo0/0/0/n+2~n+3 or Hu0/0/0/n+2~n+3 Ports n+2 and n+3 by default will be either 40G or 100G.
- For breakout in port n+2, a new breakout configuration is needed for port range n+2 n+3, as only top port n+2 supports breakout (bottom port n+3 is disabled) and cannot have a mix of 4x10G and 4x25G in the same port group.

2. Quad Port Group in 4x25G breakout mode

```
Router(config)#hw-module port-range n n+1 location 0/RP0/CPU0 mode 4x25
```

Results (TF0/0/0/n/0-3 or Hu0/0/0/n+2~n+3):

- Ports n+2 and n+3 can only be 100G.
- For breakout in port n+2, a new breakout configuration is needed for port range n+2 n+3, as only top port n+2 supports breakout (bottom port n+3 is disabled) and cannot have a mix of 4x10G and 4x25G in the same port group.

To preconfigure interfaces, you must understand these concepts:

Use of the Interface Preconfigure Command

Interfaces that are not yet present in the system can be preconfigured with the **interface preconfigure** command in global configuration mode.

The **interface preconfigure** command places the router in interface configuration mode. Users should be able to add any possible interface commands. The verifiers registered for the preconfigured interfaces verify

the configuration. The preconfiguration is complete when the user enters the **end** command, or any matching exit or global configuration mode command.



Note It is possible that some configurations cannot be verified until the line card is inserted.

Do not enter the **no shutdown** command for new preconfigured interfaces, because the no form of this command removes the existing configuration, and there is no existing configuration.

Users are expected to provide names during preconfiguration that will match the name of the interface that will be created. If the interface names do not match, the preconfiguration cannot be applied when the interface is created. The interface names must begin with the interface type that is supported by the router and for which drivers have been installed. However, the slot, port, subinterface number, and channel interface number information cannot be validated.



Note Specifying an interface name that already exists and is configured (or an abbreviated name like Hu0/3/0/0) is not permitted.



CHAPTER 3

Advanced Configuration and Modification of the Management Ethernet Interface

This module describes the configuration of Management Ethernet interfaces.

Before you can use Telnet to access the router through the LAN IP address, you must set up a Management Ethernet interface and enable Telnet servers.



Note Although the Management Ethernet interfaces on the system are present by default, the user must configure these interfaces to use them for accessing the router, using protocols and applications such as Simple Network Management Protocol (SNMP), HTTP, extensible markup language (XML), TFTP, Telnet, and command-line interface (CLI).

- [Prerequisites for Configuring Management Ethernet Interfaces, on page 13](#)
- [How to Perform Advanced Management Ethernet Interface Configuration, on page 14](#)
- [Information About Configuring Management Ethernet Interfaces, on page 21](#)

Prerequisites for Configuring Management Ethernet Interfaces

Before performing the Management Ethernet interface configuration procedures that are described in this chapter, be sure that the following tasks and conditions are met:

- You have performed the initial configuration of the Management Ethernet interface.
- You know how to apply the generalized interface name specification *rack/slot/module/port*.



Note For transparent switchover, both active and standby Management Ethernet interfaces are expected to be physically connected to the same LAN or switch.

How to Perform Advanced Management Ethernet Interface Configuration

This section contains the following procedures:

Configuring a Management Ethernet Interface

Perform this task to configure a Management Ethernet interface. This procedure provides the minimal configuration required for the Management Ethernet interface.

SUMMARY STEPS

1. **configure**
2. **interface MgmtEth** *interface-path-id*
3. **ipv4 address** *ip-address mask*
4. **mtu** *bytes*
5. **no shutdown**
6. **end** or **commit**
7. **show interfaces MgmtEth** *interface-path-id*

DETAILED STEPS

Step 1 **configure**

Example:

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

Enters global configuration mode.

Step 2 **interface MgmtEth** *interface-path-id*

Example:

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

Enters interface configuration mode and specifies the Ethernet interface name and notation *rack/slot/module/port*.

The example indicates port 0 on the RP card that is installed in slot 0.

Step 3 **ipv4 address** *ip-address mask*

Example:

```
RP/0/RP0/CPU0:router(config-if)# ipv4 address 1.76.18.150/16 (or)
ipv4 address 1.76.18.150 255.255.0.0
```

Assigns an IP address and subnet mask to the interface.

- Replace *ip-address* with the primary IPv4 address for the interface.

- Replace *mask* with the mask for the associated IP subnet. The network mask can be specified in either of two ways:
- The network mask can be a four-part dotted decimal address. For example, 255.255.0.0 indicates that each bit equal to 1 means that the corresponding address bit belongs to the network address.
- The network mask can be indicated as a slash (/) and number. For example, /16 indicates that the first 16 bits of the mask are ones, and the corresponding bits of the address are network address.

Step 4 `mtu bytes`**Example:**

```
RP/0/RP0/CPU0:router(config-if)# mtu 1488
```

(Optional) Sets the maximum transmission unit (MTU) byte value for the interface. The default is 1514.

- The default is 1514 bytes.
- The range for the Management Ethernet interface Interface **mtu** values is 64 to 1514 bytes.

Step 5 `no shutdown`**Example:**

```
RP/0/RP0/CPU0:router(config-if)# no shutdown
```

Removes the shutdown configuration, which removes the forced administrative down on the interface, enabling it to move to an up or down state.

Step 6 `end` or `commit`**Example:**

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

or

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

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

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

- 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.

Step 7 show interfaces MgmtEth *interface-path-id***Example:**

```
RP/0/RP0/CPU0:router# show interfaces MgmtEth 0/RP0/CPU0/0
```

(Optional) Displays statistics for interfaces on the router.

Example

This example displays advanced configuration and verification of the Management Ethernet interface on the RP:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface MgmtEth 0/RP0/CPU0/0
RP/0/RP0/CPU0:router(config-if)# ipv4 address 1.76.18.150/16

RP/0/RP0/CPU0:router(config-if)# no shutdown
RP/0/RP0/CPU0:router(config-if)# commit
RP/0/RP0/CPU0:router:Mar 26 01:09:28.685 :ifmgr[190]:%LINK-3-UPDOWN :Interface
MgmtEth0/RP0/CPU0/0, changed state to Up
RP/0/RP0/CPU0:router(config-if)# end

RP/0/RP0/CPU0:router# show interfaces MgmtEth 0/RP0/CPU0/0

MgmtEth0/RP0/CPU0/0 is up, line protocol is up
  Interface state transitions: 3
  Hardware is Management Ethernet, address is 1005.cad8.4354 (bia 1005.cad8.4354)
  Internet address is 1.76.18.150/16
  MTU 1488 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)
    reliability 255/255, txload 0/255, rxload 0/255
  Encapsulation ARPA,
  Full-duplex, 1000Mb/s, 1000BASE-T, link type is autonegotiation
  loopback not set,
  Last link flapped 00:00:59
  ARP type ARPA, ARP timeout 04:00:00
  Last input 00:00:00, output 00:00:02
  Last clearing of "show interface" counters never
  5 minute input rate 4000 bits/sec, 3 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    21826 packets input, 4987886 bytes, 0 total input drops
      0 drops for unrecognized upper-level protocol
    Received 12450 broadcast packets, 8800 multicast packets
      0 runts, 0 giants, 0 throttles, 0 parity
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    1192 packets output, 217483 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
  3 carrier transitions

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

interface MgmtEth0/RP0/CPU0/0
  mtu 1488
  ipv4 address 1.76.18.150/16
  ipv6 address 2002::14c:125a/64
```



```
ipv6 enable
!
```

The following example displays VRF configuration and verification of the Management Ethernet interface on the RP with source address:

```
RP/0/RP0/CPU0:router# show run interface MgmtEth 0/RP0/CPU0/0
interface MgmtEth0/RP0/CPU0/0
 vrf httpupload
 ipv4 address 10.8.67.20 255.255.0.0
 ipv6 address 2001:10:8:67::20/48
!
```

```
RP/0/RP0/CPU0:router# show run http
Wed Jan 30 14:58:53.458 UTC
http client vrf httpupload
http client source-interface ipv4 MgmtEth0/RP0/CPU0/0
```

```
RP/0/RP0/CPU0:router# show run vrf
Wed Jan 30 14:59:00.014 UTC
vrf httpupload
!
```

IPv6 Stateless Address Auto Configuration on Management Interface

Perform this task to enable IPv6 stateless auto configuration on Management interface.

SUMMARY STEPS

1. **configure**
2. **interface MgmtEth** *interface-path-id*
3. **ipv6 address autoconfig**
4. **end** or **commit**
5. **show ipv6 interfaces** *interface-path-id*

DETAILED STEPS

Step 1 **configure**

Example:

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

Step 2 **interface MgmtEth** *interface-path-id*

Example:

```
RP/0/RP0/CPU0:router(config)# interface MgmtEth 0/RP0/CPU0/0
Enters interface configuration mode and specifies the Ethernet interface name and notation rack/slot/module/port.
The example indicates port 0 on the RP card that is installed in slot 0.
```

Step 3 **ipv6 address autoconfig****Example:**

```
RP/0/RP0/CPU0:router(config-if)# ipv6 address autoconfig
```

Enable IPv6 stateless address auto configuration on the management port.

Step 4 **end or commit****Example:**

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

or

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

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

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

- 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.

Step 5 **show ipv6 interfaces *interface-path-id*****Example:**

```
RP/0/RP0/CPU0:router# show ipv6 interfaces gigabitEthernet 0/2/0/0
```

(Optional) Displays statistics for interfaces on the router.

Example

This example displays :

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface MgmtEth 0/RP0/CPU0/0
RP/0/RP0/CPU0:router(config-if)# ipv6 address autoconfig
RP/0/RP0/CPU0:router(config-if)# commit
RP/0/RP0/CPU0:router# show ipv6 interfaces gigabitEthernet 0/2/0/0
```

```

Fri Nov  4 16:48:14.372 IST
GigabitEthernet0/2/0/0 is Up, ipv6 protocol is Up, Vrfid is default (0x60000000)
  IPv6 is enabled, link-local address is fe80::d1:1eff:fe2b:baf
  Global unicast address(es):
    5::d1:1eff:fe2b:baf [AUTO CONFIGURED], subnet is 5::/64 <<<<<< auto configured address

  Joined group address(es): ff02::1:ff2b:baf ff02::2 ff02::1
  MTU is 1514 (1500 is available to IPv6)
  ICMP redirects are disabled
  ICMP unreachable are enabled
  ND DAD is enabled, number of DAD attempts 1
  ND reachable time is 0 milliseconds
  ND cache entry limit is 1000000000
  ND advertised retransmit interval is 0 milliseconds
  Hosts use stateless autoconfig for addresses.
  Outgoing access list is not set
  Inbound common access list is not set, access list is not set
  Table Id is 0xe0800000
  Complete protocol adjacency: 0
  Complete glean adjacency: 0
  Incomplete protocol adjacency: 0
  Incomplete glean adjacency: 0
  Dropped protocol request: 0
  Dropped glean request: 0

```

Modifying the MAC Address for a Management Ethernet Interface

Perform this task to configure the MAC layer address of the Management Ethernet interfaces for the RPs.

SUMMARY STEPS

1. **configure**
2. **interface MgmtEth** *interface-path-id*
3. **mac-address** *address*
4. **end** or **commit**

DETAILED STEPS

Step 1 **configure**

Example:

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

Enters global configuration mode.

Step 2 **interface MgmtEth** *interface-path-id*

Example:

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

Enters interface configuration mode and specifies the Management Ethernet interface name and instance.

Step 3 **mac-address** *address*

Example:

```
RP/0/RP0/CPU0:router(config-if)# mac-address 0001.2468.ABCD
```

Configures the MAC layer address of the Management Ethernet interface.

Note • To return the device to its default MAC address, use the **no mac-address** address command.

Step 4 end or commit

Example:

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

OR

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

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

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

- 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 Management Ethernet Interface Configuration

Perform this task to verify configuration modifications on the Management Ethernet interfaces.

SUMMARY STEPS

1. **show interfaces MgmtEth** *interface-path-id*
2. **show running-config interface MgmtEth** *interface-path-id*

DETAILED STEPS

Step 1 show interfaces MgmtEth *interface-path-id*

Example:

```
RP/0/RP0/CPU0:router# show interfaces MgmtEth 0/RP0/CPU0/0
```

Displays the Management Ethernet interface configuration.

Step 2 `show running-config interface MgmtEth interface-path-id`

Example:

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

Displays the running configuration.

Information About Configuring Management Ethernet Interfaces

To configure Management Ethernet interfaces, you must understand the following concept:



CHAPTER 4

Configuring Ethernet Interfaces

This module describes the configuration of Ethernet interfaces.

Table 5: Feature History Table

Feature Name	Release	Description
OTN Support for NC55-MPA-12T-S MPA on Cisco NCS 5500 Series Routers.	Release 7.5.1	<p>This release introduces support for Optical Network Transport (OTN) on NC55-MPA-12T-S Modular Port Adapter (MPA) on the following Cisco NCS 5500 Series Line cards:</p> <ul style="list-style-type: none"> • NCS-55A2-MOD-S • NCS-55A2-MOD-SE-S • NCS-55A2-MOD-HX-S • NCS-55A2-MOD-SE-H-S • NCS-55A2-MOD-HD-S <p>OTN is a superior technology that bridges the gap between next-generation IP and legacy time-division multiplexing (TDM) networks by acting as a converged transport layer for newer packet-based and existing TDM services. OTN provides robust transport services that leverage many benefits of SONET/SDH, such as resiliency and performance monitoring, while adding enhanced multi-rate capabilities in packet traffic.</p> <p>The Cisco NCS 5500 Series Routers support Ethernet, SONET/SDH, and OTN client interfaces with data rates from 1 to 10 Gigabits per second.</p> <p>To enable OTN, use the pm otn report enable command in the otu2e or odu2e mode.</p>

The following distributed ethernet architecture delivers network scalability and performance, while enabling service providers to offer high-density, high-bandwidth networking solutions.

- 10-Gigabit
- 40-Gigabit
- 100-Gigabit



Tip You can programmatically configure and manage the Ethernet interfaces using `openconfig-ethernet-if.yang` and `openconfig-interfaces.yang` OpenConfig data models. To get started with using data models, see the *Programmability Configuration Guide for Cisco NCS 5500 Series Routers*.

These solutions are designed to interconnect the router with other systems in point-of-presence (POP)s, including core and edge routers and Layer 2 and Layer 3 switches.

Restrictions for Configuring Ethernet Interfaces

- Router does not support configuration of the static mac address.
- As per design, traffic logs for incoming CRC error packets don't display packets per second (PPS) and other packet-specific information, as highlighted below.

```
Router# show interface tenGigE 0/0/0/10 | include packets

5 minute input rate 541242000 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
  0 packets input, 7718374402816 bytes, 0 total input drops
  Received 0 broadcast packets, 0 multicast packets
  2952 packets output, 389664 bytes, 0 total output drops
  Output 0 broadcast packets, 2952 multicast packets
```

- To save power consumption, the router shuts down the ethernet interfaces with no configuration other than **no shutdown** after a graceful line card or system reload. To avoid ethernet interfaces going down in such scenarios, you must provide a description and the **no shutdown** configuration under the interface. You can use the [description \(interface\)](#) to add a description to an interface.
- The router doesn't support connecting a 1Gig copper cable to a 25GbE or higher speed QSFP ports.
- For 1Gig fibre cable, the router doesn't support auto-negotiation for 25GbE or higher speed QSFP ports.
- [Configuring Gigabit Ethernet Interfaces](#), on page 25
- [Information About Configuring Ethernet](#), on page 29
- [Link Layer Discovery Protocol \(LLDP\)](#), on page 37
- [Dense Wavelength Division Multiplexing Tunable Optics](#), on page 41
- [Priority Flow Control \(PFC\)](#) , on page 53
- [Optical Transport Networks](#), on page 57
- [How to Configure Interfaces in Breakout Mode](#), on page 61

Configuring Gigabit Ethernet Interfaces

Restrictions and Important Guidelines

- NC55-MPA-12T-S supports 1G optics in eight ports. The ports are 0 to 3 and 8 to 11.
- NC55-MPA-12T-S supports 10G optics in ports 4 to 7.

Use this procedure to create a basic Ethernet interface configuration.

SUMMARY STEPS

1. **show version**
2. **show interfaces** [**GigE** | **TenGigE** | | | **HundredGigE**] *interface-path-id*
3. **configure**
4. **interface** [**GigE** | **TenGigE** | | | **HundredGigE**] *interface-path-id*
5. **ipv4 address** *ip-address mask*
6. **mtu** *bytes*
7. **no shutdown**
8. **end** or **commit**
9. **show interfaces** [**GigE** **TenGigE** **HundredGigE**] *interface-path-id*

DETAILED STEPS

Step 1 **show version**

Example:

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

(Optional) Displays the current software version, and can also be used to confirm that the router recognizes the line card.

Step 2 **show interfaces** [**GigE** | **TenGigE** | | | **HundredGigE**] *interface-path-id*

Example:

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

(Optional) Displays the configured interface and checks the status of each interface port.

Step 3 **configure**

Example:

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

Enters global configuration mode.

Step 4 **interface** [**GigE** | **TenGigE** | | | **HundredGigE**] *interface-path-id*

Example:

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

Enters interface configuration mode and specifies the Ethernet interface name and notation *rack/slot/module/port*. Possible interface types for this procedure are:

- GigE
- 10GigE
- 100GigE

Note • The example indicates a 100-Gigabit Ethernet interface in the line card in slot 1.

Step 5 **ipv4 address** *ip-address mask***Example:**

```
RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224
```

Assigns an IP address and subnet mask to the interface.

- Replace *ip-address* with the primary IPv4 address for the interface.
- Replace *mask* with the mask for the associated IP subnet. The network mask can be specified in either of two ways:
 - The network mask can be a four-part dotted decimal address. For example, 255.0.0.0 indicates that each bit equal to 1 means that the corresponding address bit belongs to the network address.
 - The network mask can be indicated as a slash (/) and number. For example, /8 indicates that the first 8 bits of the mask are ones, and the corresponding bits of the address are network address.

Step 6 **mtu** *bytes***Example:**

```
RP/0/RP0/CPU0:router(config-if)# mtu 2000
```

(Optional) Sets the MTU value for the interface.

- The configurable range for MTU values is 1514 bytes to 9646 bytes.
- The default is 1514 bytes for normal frames and 1518 bytes for 802.1Q tagged frames.

Step 7 **no shutdown****Example:**

```
RP/0/RP0/CPU0:router(config-if)# no shutdown
```

Removes the shutdown configuration, which forces an interface administratively down.

Step 8 **end** or **commit****Example:**

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

or

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

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

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

- 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.

Step 9 **show interfaces [GigE TenGigE HundredGigE] interface-path-id**

Example:

```
RP/0/RP0/CPU0:router# show interfaces HundredGigE 0/1/0/1
```

(Optional) Displays statistics for interfaces on the router.

Example

This example shows how to configure an interface for a 100-Gigabit Ethernet line card:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface HundredGigE 0/1/0/1
RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224

RP/0/RP0/CPU0:router(config-if)# mtu 2000

RP/0/RP0/CPU0:router(config-if)# no shutdown
RP/0/RP0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes
```

```
RP/0/RP0/CPU0:router# show interfaces HundredGigE 0/5/0/24
HundredGigE0/5/0/24 is up, line protocol is up
  Interface state transitions: 1
  Hardware is HundredGigE, address is 6219.8864.e330 (bia 6219.8864.e330)
  Internet address is 3.24.1.1/24
  MTU 9216 bytes, BW 100000000 Kbit (Max: 100000000 Kbit)
    reliability 255/255, txload 3/255, rxload 3/255
  Encapsulation ARPA,
  Full-duplex, 100000Mb/s, link type is force-up
  output flow control is off, input flow control is off
  Carrier delay (up) is 10 msec
  loopback not set,
  Last link flapped 10:05:07
  ARP type ARPA, ARP timeout 04:00:00
  Last input 00:08:56, output 00:00:00
  Last clearing of "show interface" counters never
  5 minute input rate 1258567000 bits/sec, 1484160 packets/sec
  5 minute output rate 1258584000 bits/sec, 1484160 packets/sec
    228290765840 packets input, 27293508436038 bytes, 0 total input drops
    0 drops for unrecognized upper-level protocol
  Received 15 broadcast packets, 45 multicast packets
    0 runts, 0 giants, 0 throttles, 0 parity
  0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
  212467849449 packets output, 25733664696650 bytes, 0 total output drops
  Output 23 broadcast packets, 15732 multicast packets
  39 output errors, 0 underruns, 0 applique, 0 resets
```

```
0 output buffer failures, 0 output buffers swapped out
0 carrier transitions
```

```
RP/0/RP0/CPU0:router# show running-config interface HundredGigE 0/5/0/24
```

```
interface HundredGigE 0/5/0/24
mtu 9216
service-policy input linerate
service-policy output elinerate
ipv4 address 3.24.1.1 255.255.255.0
ipv6 address 3:24:1::1/64
flow ipv4 monitor perfv4 sampler fsm ingress
!
```

Information About Configuring Ethernet

This section provides the following information sections:

Default Configuration Values for 100-Gigabit Ethernet

This table describes the default interface configuration parameters that are present when an interface is enabled on a 100-Gigabit Ethernet line card.



Note You must use the **shutdown** command to bring an interface administratively down. The interface default is **no shutdown**. When a line card is first inserted into the router, if there is no established preconfiguration for it, the configuration manager adds a shutdown item to its configuration. This shutdown can be removed only by entering the **no shutdown** command.

Table 6: 100-Gigabit Ethernet line card Default Configuration Values

Parameter	Configuration File Entry	Default Value
MTU	mtu	<ul style="list-style-type: none"> • 1514 bytes for normal frames • 1518 bytes for 802.1Q tagged frames. • 1522 bytes for Q-in-Q frames.
MAC address	mac address	Hardware burned-in address (BIA)

Network Interface Speed

1Gig interfaces connected through copper or fiber cable can have interface speed of either 100 Mbps or 1000 Mbps. This is applicable on 1Gig interface with a 1000Base-T module (GLC-TE). By default 1G interface has following capabilities:

- Speed—1000 Mbps for fiber cable and autonegotiate for copper cable
- Duplex—Full
- Pause—Receive Part (RX) and Transmit Part (TX)

The copper and fiber cables have same default values as mentioned above but autonegotiation is default for copper cable.

The speed can either configured or set to autonegotiate with remote end interface. When in autonegotiation mode, an interface is capable of negotiating the speed of 100 Mbps or 1000 Mbps depending on the speed at the remote end interface; and other parameters such as full duplex and pause are also autonegotiated.

Autonegotiation is an optional function of the Fast Ethernet standard that enables devices to automatically exchange information over a link about speed and duplex abilities. Autonegotiation is very useful for ports where devices with different capabilities are connected and disconnected on a regular basis.



Note Autonegotiation is disabled by default, but it's mandatory on QSFP-100G-CUxM link. You must enable autonegotiation manually when you use 100GBASE-CR4 DAC cable.

Configuring Network Interface Speed

You can configure the network interface speed by using one of the following methods:

- Using the **speed** command
- Using the **negotiation auto** command
- Using both **speed** and **negotiation auto** command



Note Cisco recommends configuring network interface speed in autonegotiation mode.

Using the speed command

When you configure the speed of the network interface (1G) using the **speed** command, the interface speed is forced to the configured speed by limiting the speed value of the auto negotiated parameter to the configured speed.

This sample configuration forces the Gig interface speed to 100Mbps.



Note The interface speed at remote end is also set to 100Mbps.

```
#configuration
(config)#interface GigabitEthernet 0/0/0/31
(config-if)#speed 100
(config-if)#commit
(config-if)#end
```

Use the **show controller GigE** and **show interface GigE** commands to verify if the speed is configured to 100Mbps and autonegotiation is disabled:

```
#show controllers GigabitEthernet 0/0/0/31
Operational data for interface GigabitEthernet0/0/0/31:
State:
  Administrative state: enabled
  Operational state: Up
  LED state: Green On
Phy:
  Media type: Four-pair Category 5 UTP PHY, full duplex
Optics:
  Vendor: CISCO
  Part number: SBCU-5740ARZ-CS1
  Serial number: AVC194525HW
  Wavelength: 0 nm
Digital Optical Monitoring:
  Transceiver Temp: 0.000 C
  Transceiver Voltage: 0.000 V

Alarms key: (H) Alarm high, (h) Warning high
            (L) Alarm low, (l) Warning low

```

Wavelength	Tx Power	Rx Power	Laser Bias
Lane (nm)	(dBm) (mW)	(dBm) (mW)	(mA)
0 n/a	0.0 1.0000	0.0 1.0000	0.000

```

DOM alarms:
  No alarms

Alarm          Alarm      Warning   Warning   Alarm
Thresholds    High       High      Low       Low
-----
Transceiver Temp (C):    0.000    0.000    0.000    0.000
Transceiver Voltage (V): 0.000    0.000    0.000    0.000
Laser Bias (mA):        0.000    0.000    0.000    0.000
Transmit Power (mW):     1.000    1.000    1.000    1.000
Transmit Power (dBm):    0.000    0.000    0.000    0.000
Receive Power (mW):      1.000    1.000    1.000    1.000
Receive Power (dBm):     0.000    0.000    0.000    0.000
Statistics:
  FEC:
    Corrected Codeword Count: 0
    Uncorrected Codeword Count: 0

MAC address information:
  Operational address: 0035.1a00.e67c
  Burnt-in address: 0035.1a00.e62c
Autonegotiation disabled.

Operational values:
Speed: 100Mbps /*Gig interface speed is set to 100Mbps */
Duplex: Full Duplex
Flowcontrol: None
Loopback: None (or external)
MTU: 1514
MRU: 1514
Forward error correction: Disabled
```

```
#show interfaces GigabitEthernet 0/0/0/31
GigabitEthernet0/0/0/31 is up, line protocol is up
  Interface state transitions: 7
  Hardware is GigabitEthernet, address is 0035.1a00.e62c (bia 0035.1a00.e62c)
  Internet address is Unknown
  MTU 1514 bytes, BW 100000 Kbit (Max: 100000 Kbit)
    reliability 255/255, txload 0/255, rxload 0/255
  Encapsulation ARPA,
  Full-duplex, 100Mb/s, TFD, link type is force-up
  output flow control is off, input flow control is off
  Carrier delay (up) is 10 msec
  loopback not set,
  Last link flapped 00:00:30
  Last input 00:00:00, output 00:00:00
  Last clearing of "show interface" counters never
  30 second input rate 1000 bits/sec, 1 packets/sec
  30 second output rate 0 bits/sec, 1 packets/sec
    90943 packets input, 11680016 bytes, 0 total input drops
    0 drops for unrecognized upper-level protocol
    Received 0 broadcast packets, 90943 multicast packets
      0 runts, 0 giants, 0 throttles, 0 parity
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    61279 packets output, 4347618 bytes, 0 total output drops
    Output 0 broadcast packets, 8656 multicast packets
    0 output errors, 0 underruns, 0 applique, 0 resets
    0 output buffer failures, 0 output buffers swapped out
  8 carrier transitions
```

In the above show output you will observe that the state of the GigabitEthernet0/0/0/31 is up, and line protocol is up. This is because the speed at both ends is 100Mbps.

Using the negotiation auto command

When you configure the network interface speed using **negotiation auto** command, the speed is autonegotiated with the remote end interface. This command enhances the speed capability to 100M or 1G to be negotiated with the peer.

This sample configuration sets the interface speed to autonegotiate:



Note The interface speed at remote end is set to 100Mbps.



Note Before Cisco IOS XR Software Release 7.3.2, the default setting for autonegotiation varied with different platforms under the NCS 5500 family. On NCS 540 and NCS 55A2, 100G autonegotiation was enabled by default.

From Cisco IOS XR Software Release 7.3.2 onwards, autonegotiation is not enabled by default. Use the **negotiation auto** command to enable autonegotiation.

```
#configuration
(config)#interface GigabitEthernet 0/0/0/31
(config-if)#negotiation auto
```



```
(config-if)#commit
(config-if)#end
```

Use the **show controller GigE** and **show interface GigE** commands to verify if the speed is autonegotiated:

```
#show interfaces GigabitEthernet 0/0/0/31
GigabitEthernet0/0/0/31 is up, line protocol is up
  Interface state transitions: 10
  Hardware is GigabitEthernet, address is 0035.1a00.e62c (bia 0035.1a00.e62c)
  Internet address is Unknown
  MTU 1514 bytes, BW 100000 Kbit (Max: 100000 Kbit)
    reliability 255/255, txload 0/255, rxload 0/255
  Encapsulation ARPA,
  Full-duplex, 100Mb/s, TFD, link type is autonegotiation
  output flow control is off, input flow control is off
  Carrier delay (up) is 10 msec
  loopback not set,
  Last link flapped 00:00:01
  Last input 00:00:00, output 00:00:00
  Last clearing of "show interface" counters never
  30 second input rate 1000 bits/sec, 1 packets/sec
  30 second output rate 0 bits/sec, 0 packets/sec
  91005 packets input, 11687850 bytes, 0 total input drops
    0 drops for unrecognized upper-level protocol
  Received 0 broadcast packets, 91005 multicast packets
    0 runts, 0 giants, 0 throttles, 0 parity
  0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
  61307 packets output, 4350024 bytes, 0 total output drops
  Output 0 broadcast packets, 8668 multicast packets
  0 output errors, 0 underruns, 0 applique, 0 resets
  0 output buffer failures, 0 output buffers swapped out
  15 carrier transitions
```

In the above show output you see that GigabitEthernet0/0/0/31 is up, and line protocol is up.

```
#show controllers GigabitEthernet 0/0/0/31
Operational data for interface GigabitEthernet0/0/0/31:

State:
  Administrative state: enabled
  Operational state: Up
  LED state: Green On

Phy:
  Media type: Four-pair Category 5 UTP PHY, full duplex
  Optics:
    Vendor: CISCO
    Part number: SBCU-5740ARZ-CS1
    Serial number: AVC194525HW
    Wavelength: 0 nm
  Digital Optical Monitoring:
    Transceiver Temp: 0.000 C
    Transceiver Voltage: 0.000 V

  Alarms key: (H) Alarm high, (h) Warning high
              (L) Alarm low, (l) Warning low

    Wavelength  Tx Power      Rx Power      Laser Bias
  Lane  (nm)      (dBm)         (mW)          (dBm)         (mW)         (mA)
  ---  -
  0     n/a      0.0   1.0000     0.0   1.0000     0.000

  DOM alarms:
    No alarms
```

```

Alarm
Thresholds
-----
Transceiver Temp (C):      0.000    0.000    0.000    0.000
Transceiver Voltage (V):  0.000    0.000    0.000    0.000
Laser Bias (mA):          0.000    0.000    0.000    0.000
Transmit Power (mW):       1.000    1.000    1.000    1.000
Transmit Power (dBm):      0.000    0.000    0.000    0.000
Receive Power (mW):        1.000    1.000    1.000    1.000
Receive Power (dBm):       0.000    0.000    0.000    0.000
Statistics:
FEC:
    Corrected Codeword Count: 0
    Uncorrected Codeword Count: 0

MAC address information:
Operational address: 0035.1a00.e67c
Burnt-in address: 0035.1a00.e62c

Autonegotiation enabled:
    No restricted parameters

Operational values:
    Speed: 100Mbps
    Duplex: Full Duplex
    Flowcontrol: None
    Loopback: None (or external)
    MTU: 1514
    MRU: 1514
    Forward error correction: Disabled

```

Using speed and negotiation auto command

When you configure the speed of the network interface (1G) using the **speed** and **negotiation auto** command, the interface autonegotiates all the paramets (full-duplex and pause) except speed. The speed is forced to the configured value.

This sample shows how to configures Gig interface speed to 100Mbps and autonegotiate other parameters:



Note The interface speed at remote end is set to 100Mbps.

```

#configuration
(config)#interface GigabitEthernet 0/0/0/31
(config-if)#negotiation auto
(config-if)#speed 100
(config-if)#end

```

Use the **show controller GigE** and **show interface GigE** command to verify if the link is up, speed is forced to 100Mbps and autonegotiation is enabled:

```

#show interfaces GigabitEthernet 0/0/0/31
GigabitEthernet0/0/0/31 is up, line protocol is up
Interface state transitions: 9
Hardware is GigabitEthernet, address is 0035.1a00.e62c (bia 0035.1a00.e62c)
Internet address is Unknown
MTU 1514 bytes, BW 100000 Kbit (Max: 100000 Kbit)
    reliability 255/255, txload 0/255, rxload 0/255
Encapsulation ARPA,

```

```

Full-duplex, 100Mb/s, TFD, link type is autonegotiation
output flow control is off, input flow control is off
Carrier delay (up) is 10 msec
loopback not set,
Last link flapped 00:00:03
Last input 00:00:00, output 00:00:00
Last clearing of "show interface" counters never
30 second input rate 0 bits/sec, 1 packets/sec
30 second output rate 0 bits/sec, 0 packets/sec
 90968 packets input, 11683189 bytes, 0 total input drops
 0 drops for unrecognized upper-level protocol
Received 0 broadcast packets, 90968 multicast packets
      0 runts, 0 giants, 0 throttles, 0 parity
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
61287 packets output, 4348541 bytes, 0 total output drops
Output 0 broadcast packets, 8664 multicast packets
0 output errors, 0 underruns, 0 applique, 0 resets
0 output buffer failures, 0 output buffers swapped out
12 carrier transitions
    
```

In the above show output you will observe that the GigabitEthernet0/0/0/31 is up, and line protocol is up This is because the speed at both ends is 100Mbps.

#show controllers GigabitEthernet 0/0/0/31

Operational data for interface GigabitEthernet0/0/0/31:

State:

```

Administrative state: enabled
Operational state: Up
LED state: Green On
    
```

Phy:

Media type: Four-pair Category 5 UTP PHY, full duplex

Optics:

```

Vendor: CISCO
Part number: SBCU-5740ARZ-CS1
Serial number: AVC194525HW
Wavelength: 0 nm
    
```

Digital Optical Monitoring:

```

Transceiver Temp: 0.000 C
Transceiver Voltage: 0.000 V
    
```

Alarms key: (H) Alarm high, (h) Warning high
(L) Alarm low, (l) Warning low

Lane	Wavelength (nm)	Tx Power		Rx Power		Laser Bias (mA)
		(dBm)	(mW)	(dBm)	(mW)	
0	n/a	0.0	1.0000	0.0	1.0000	0.000

DOM alarms:

No alarms

Alarm Thresholds	Alarm High	Warning High	Warning Low	Alarm Low
Transceiver Temp (C):	0.000	0.000	0.000	0.000
Transceiver Voltage (V):	0.000	0.000	0.000	0.000
Laser Bias (mA):	0.000	0.000	0.000	0.000
Transmit Power (mW):	1.000	1.000	1.000	1.000
Transmit Power (dBm):	0.000	0.000	0.000	0.000
Receive Power (mW):	1.000	1.000	1.000	1.000
Receive Power (dBm):	0.000	0.000	0.000	0.000

```

Statistics:
  FEC:
    Corrected Codeword Count: 0
    Uncorrected Codeword Count: 0

MAC address information:
  Operational address: 0035.1a00.e67c
  Burnt-in address: 0035.1a00.e62c

Autonegotiation enabled:
  Speed restricted to: 100Mbps /* autonegotiation is enabled and speed is forced to
  100Mbps*/

Operational values:
  Speed: 100Mbps
  Duplex: Full Duplex
  Flowcontrol: None
  Loopback: None (or external)
  MTU: 1514
  MRU: 1514
  Forward error correction: Disabled

```

Ethernet MTU

The Ethernet maximum transmission unit (MTU) is the size of the largest frame, minus the 4-byte frame check sequence (FCS), that can be transmitted on the Ethernet network. Every physical network along the destination of a packet can have a different MTU.

Cisco IOS XR software supports two types of frame forwarding processes:

- Fragmentation for IPV4 packets—In this process, IPv4 packets are fragmented as necessary to fit within the MTU of the next-hop physical network.



Note IPv6 does not support fragmentation.

- MTU discovery process determines largest packet size—This process is available for all IPV6 devices, and for originating IPv4 devices. In this process, the originating IP device determines the size of the largest IPv6 or IPV4 packet that can be sent without being fragmented. The largest packet is equal to the smallest MTU of any network between the IP source and the IP destination devices. If a packet is larger than the smallest MTU of all the networks in its path, that packet will be fragmented as necessary. This process ensures that the originating device does not send an IP packet that is too large.



Note To enable hashing for L3 header only when the majority of traffic is fragmented, use the [hw-module profile load-balance algorithm L3-Only](#) command.

Jumbo frame support is automatically enable for frames that exceed the standard frame size. The default value is 1514 for standard frames and 1518 for 802.1Q tagged frames. These numbers exclude the 4-byte frame check sequence (FCS).

Following are the supported MTU properties on devices containing NC55 first generation line cards, NCS 5501, NCS5501-SE cards:

- Each physical port can have a different MTU.
- Main interface of each bundle can have one MTU value.
- L3 sub-interface (bundle or physical) shares MTU profiles and can have a maximum of 3 unique configured MTUs per NPU.



Note L2 sub-interface MTU is not supported.

Following are the supported MTU profiles on devices containing NC55 second generation line cards. Each profile has a limit of 7 configured MTUs per NPU.

- Port MTU profile: This is shared by bundle main interfaces and physical main interfaces.
- RIF MTU profile: This is used by the L3 sub-interface (bundle or physical) on the device.



Note L2 sub-interface MTU is not supported.

For more information about the architecture, refer to the [NCS 5500 and NCS 5700 Fixed Platform Architecture white paper](#).

Link Layer Discovery Protocol (LLDP)

Cisco Discovery Protocol (CDP) is a device discovery protocol that runs over Layer 2. Layer 2 is also known as the data link layer that runs on all Cisco-manufactured devices, such as routers, bridges, access servers, and switches. CDP allows the network management applications to automatically discover and learn about other Cisco devices that connect to the network.

To support non-Cisco devices and to allow for interoperability between other devices, it also supports the IEEE 802.1AB LLDP. LLDP is also a neighbor discovery protocol that is used for network devices to advertise information about themselves to other devices on the network. This protocol runs over the data link layer, which allows two systems running different network layer protocols to learn about each other.

With LLDP, you can also access the information about a particular physical network connection. If you use a non-Cisco monitoring tool (via SNMP,) LLDP helps you identify the Object Identifiers (OIDs) that the system supports. The following are the supported OIDs:

- 1.0.8802.1.1.2.1.4.1.1.4
- 1.0.8802.1.1.2.1.4.1.1.5
- 1.0.8802.1.1.2.1.4.1.1.6
- 1.0.8802.1.1.2.1.4.1.1.7
- 1.0.8802.1.1.2.1.4.1.1.8
- 1.0.8802.1.1.2.1.4.1.1.9
- 1.0.8802.1.1.2.1.4.1.1.10

- 1.0.8802.1.1.2.1.4.1.1.11
- 1.0.8802.1.1.2.1.4.1.1.12

Specifying User-Defined LLDP TLV Values

It is possible to override the system default values for some of the mandatory LLDP Type-Length-Values (TLVs) that are advertised by routers to their directly connected neighboring devices. While advertising their identity and capabilities, routers can assign user-defined meaningful names instead of autogenerated values. Using the following CLIs you can specify these user-defined values:

- Router(config)#lldp tlv-select: Choose the wire-power management (Cisco 4-wire Power via MDI TLV or IEEE 802.3 DTE Power)
- Router(config)#lldp system-name *system-name*
- Router(config)#lldp system-description *system-description*
- Router(config)#lldp chassis-id-type *chassis-type*
- Router(config)#lldp chassis-id *local-chassis-id*



Note The **chassis-id** value is configurable only when the **chassis-id-type** is set as **Local**. If there is a mismatch, you encounter a configuration failed error message.

The configured values, such as the system name, system description, chassis-id, chassis-type become part of the TLV in the LLDP packets that are sent to its neighbors. Values are transmitted only to LLDP enabled interfaces to which the router is connected.

You can assign any of the following values for the `chassis-id-type`. The chassis-id-types are objects that are part of the [management information base \(MIB\)](#). Depending on the selected chassis-id-type, values are assigned to these objects, and they are advertised by the router to its neighboring devices.

chassis-id-type	Description
chassis-component	Chassis identifier based on the value of entPhysicalAlias object that is defined in IETF RFC 2737.
interface-alias	Chassis identifier based on the value of ifAlias object as defined in IETF RFC 2863.
interface-name	Chassis identifier based on the name of the interface.
local	Chassis identifier based on a locally defined value.
mac-address	Chassis identifier based on the value of a unicast source address.
network-address	Chassis identifier based on a network address that is associated with a particular chassis.

chassis-id-type	Description
port-component	Chassis identifier based on the value of entPhysicalAlias object defined in IETF RFC 2737 for a port or backplane component.



Tip You can programmatically modify default values of LLDP TLVs by using the `openconfig-lldp` OpenConfig data model. To get started with using data models, see the *Programmability Configuration Guide for Cisco NCS 5500 Series Routers*.

Configuration Example

This example shows the configuration for the LLDP TLVs that will be advertised by routers to their directly connected neighboring devices.

```
Router(config)#lldp system-name cisco-xr
Router(config)#lldp system-description cisco-xr-edge-device
Router(config)#lldp chassis-id-type local
Router(config)#lldp chassis-id ce-device9
```

Running Configuration

```
Router#show lldp
Tue Sep 13 16:03:44.550 +0530
Global LLDP information:
Status: ACTIVE
LLDP Chassis ID: ce-device9
LLDP Chassis ID Subtype: Locally Assigned Chassis Subtype
LLDP System Name: cisco-xr
LLDP advertisements are sent every 30 seconds
LLDP hold time advertised is 120 seconds
LLDP interface reinitialisation delay is 2 seconds
```

Enabling LLDP Globally

To run LLDP on the router, you must enable it globally. When you enable LLDP globally, all interfaces that support LLDP are automatically enabled for both transmit and receive operations.

You can override this default operation at the interface to disable receive or transmit operations.

The following table describes the global attributes that you can configure:

Attribute	Default	Range	Description
Holdtime	120	0-65535	Specifies the holdtime (in sec) that are sent in packets
Reinit	2	2-5	Delay (in sec) for LLDP initialization on any interface

Attribute	Default	Range	Description
Timer	30	5-65534	Specifies the rate at which LLDP packets are sent (in sec)

To enable LLDP globally, complete the following steps:

1. RP/0/RSP0/CPU0:router # configure
2. RP/0/RSP0/CPU0:router(config) #lldp
3. end or commit

Running configuration

```
RP/0/RP0/CPU0:router-5#show run lldp
Fri Dec 15 20:36:49.132 UTC
lldp
!
```

```
RP/0/RP0/CPU0:router#show lldp neighbors
Fri Dec 15 20:29:53.763 UTC
Capability codes:
  (R) Router, (B) Bridge, (T) Telephone, (C) DOCSIS Cable Device
  (W) WLAN Access Point, (P) Repeater, (S) Station, (O) Other
```

```
Device ID      Local Intf      Hold-time  Capability  Port ID
SW-NOSTG-I11-PUB.cis Mg0/RP0/CPU0/0    120        N/A         Fa0/28
```

Total entries displayed: 1

```
RP/0/RP0/CPU0:router#show lldp neighbors mgmtEth 0/RP0/CPU0/0
Fri Dec 15 20:30:54.736 UTC
Capability codes:
```

```
  (R) Router, (B) Bridge, (T) Telephone, (C) DOCSIS Cable Device
  (W) WLAN Access Point, (P) Repeater, (S) Station, (O) Other
```

```
Device ID      Local Intf      Hold-time  Capability  Port ID
SW-NOSTG-I11-PUB.cis Mg0/RP0/CPU0/0    120        N/A         Fa0/28
```

Total entries displayed: 1

Enabling LLDP Per Interface

When you enable LLDP globally, all interfaces that support LLDP are automatically enabled for both transmit and receive operations. However, if you want to enable LLDP per interface, perform the following configuration steps:

1. RP/0/RSP0/CPU0:router(config)# int gigabitEthernet 0/2/0/0
2. RP/0/RSP0/CPU0:router(config-if)# no sh
3. RP/0/RSP0/CPU0:router(config-if)#commit
4. RP/0/RSP0/CPU0:router(config-if)#lldp ?
5. RP/0/RSP0/CPU0:router(config-if)#lldp enable
6. RP/0/RSP0/CPU0:router(config-if)#commit

Running configuration

```
RP/0/RSP0/CPU0:router#sh running-config
Wed Jun 27 12:40:21.274 IST
Building configuration...
!! IOS XR Configuration 0.0.0
!! Last configuration change at Wed Jun 27 00:59:29 2018 by UNKNOWN
!
interface GigabitEthernet0/1/0/0
 shutdown
!
interface GigabitEthernet0/1/0/1
 shutdown
!
interface GigabitEthernet0/1/0/2
 shutdown
!
interface GigabitEthernet0/2/0/0
 Shutdown
!
interface GigabitEthernet0/2/0/1
 shutdown
!
interface GigabitEthernet0/2/0/2
 shutdown
!
end
```

Verification

Verifying the config

=====

```
RP/0/RSP0/CPU0:router#sh lldp interface <===== LLDP enabled only on GigEth0/2/0/0
Wed Jun 27 12:43:26.252 IST
```

```
GigabitEthernet0/2/0/0:
  Tx: enabled
  Rx: enabled
  Tx state: IDLE
  Rx state: WAIT FOR FRAME
RP/0/RSP0/CPU0:router#
```

```
RP/0/RSP0/CPU0:router# show lldp neighbors
Wed Jun 27 12:44:38.977 IST
Capability codes:
```

```
(R) Router, (B) Bridge, (T) Telephone, (C) DOCSIS Cable Device
(W) WLAN Access Point, (P) Repeater, (S) Station, (O) Other
```

```
Device ID      Local Intf      Hold-time  Capability  Port ID
ios            Gi0/2/0/0      120       R           Gi0/2/0/0      <===== LLDP
enabled only on GigEth0/2/0/0 and neighborship seen for the same.
```

Total entries displayed: 1

```
RP/0/RSP0/CPU0:router#
```

Dense Wavelength Division Multiplexing Tunable Optics

The Dense Wavelength-Division Multiplexing (DWDM) wavelengths of the DWDM-SFP10G-C module on the Cisco NCS 5500 Series Aggregation Services Routers is tunable. You can configure the DWDM ITU

wavelengths by using the `itu channel` command in the interface configuration mode. The `itu channel` command ensures that the traffic continues to flow.

The following table contains the wavelength mapping information for the DWDM module:

Channel	Frequency (THz)	Wavelength (nm)
1	191.35	1566.723
2	191.40	1566.314
3	191.45	1565.905
4	191.50	1565.496
5	191.55	1565.087
6	191.60	1564.679
7	191.65	1564.271
8	191.70	1563.863
9	191.75	1563.455
10	191.80	1563.047
11	191.85	1562.640
12	191.90	1562.233
13	191.95	1561.826
14	192.00	1561.419
15	192.05	1561.013
16	192.10	1560.606
17	192.15	1560.200
18	192.20	1559.794
19	192.25	1559.389
20	192.30	1558.983
21	192.35	1558.578
22	192.40	1558.173
23	192.45	1557.768
24	192.50	1557.363
25	192.55	1556.959
26	192.60	1556.555

Channel	Frequency (THz)	Wavelength (nm)
27	192.65	1556.151
28	192.70	1555.747
29	192.75	1555.343
30	192.80	1554.940
31	192.85	1554.537
32	192.90	1554.134
33	192.95	1553.731
34	193.00	1553.329
35	193.05	1552.926
36	193.10	1552.524
37	193.15	1552.122
38	193.20	1551.721
39	193.25	1551.319
40	193.30	1550.918
41	193.35	1550.517
42	193.40	1550.116
43	193.45	1549.715
44	193.50	1549.315
45	193.55	1548.915
46	193.60	1548.515
47	193.65	1548.115
48	193.70	1547.715
49	193.75	1547.316
50	193.80	1546.917
51	193.85	1546.518
52	193.90	1546.119
53	193.95	1545.720
54	194.00	1545.322

Channel	Frequency (THz)	Wavelength (nm)
55	194.05	1544.924
56	194.10	1544.526
57	194.15	1544.128
58	194.20	1543.730
59	194.25	1543.333
60	194.30	1542.936
61	194.35	1542.539
62	194.40	1542.142
63	194.45	1541.746
64	194.50	1541.349
65	194.55	1540.953
66	194.60	1540.557
67	194.65	1540.162
68	194.70	1539.766
69	194.75	1539.371
70	194.80	1538.976
71	194.85	1538.581
72	194.90	1538.186
73	194.95	1537.792
74	195.00	1537.397
75	195.05	1537.003
76	195.10	1536.609
77	195.15	1536.216
78	195.20	1535.822
79	195.25	1535.429
80	195.30	1535.036
81	195.35	1534.643
82	195.40	1534.250

Channel	Frequency (THz)	Wavelength (nm)
83	195.45	1533.858
84	195.50	1533.465
85	195.55	1533.073
86	195.60	1532.681
87	195.65	1532.290
88	195.70	1531.898
89	195.75	1531.507
90	195.80	1531.116
91	195.85	1530.725
92	195.90	1530.334
93	195.95	1529.944
94	196.00	1529.553
95	196.05	1529.163
96	196.10	1528.773



Note For more information on limitations of this feature and details about optical parameters, see https://www.cisco.com/c/en/us/products/collateral/interfaces-modules/dwdm-transceiver-modules/data_sheet_c78-711186.html.

Configuring the DWDM Tunable Optics

Perform the following procedure to configure the DWDM Tunable Optics module:

1. Router# enable //Enables the privileged EXEC mode. If prompted, enter your password.
2. Router# configure terminal
3. Router(config)# interface tengigabitethernet 4/11 // Specifies the 10-Gigabit Ethernet interface to be configured. slot/port—Specifies the location of the interface.
4. Router(config-if)# itu channel 28 //Sets the ITU channel. *number* specifies the ITU channel number. The acceptable values are from 1-96.

Verifying the ITU Configuration

The following example shows how to use the show controller optics command to check an ITU configuration:

```

RP/0/RP0/CPU0:ios#show controllers optics 0/0/0/16
Tue Sep  5 08:25:54.127 UTC

Controller State: Up

Transport Admin State: In Service

Laser State: Off

LED State: Off

Optics Status

    Optics Type:  SFP+ 10G DWDM Tunable
    DWDM carrier Info:  C BAND, MSA ITU Channel=49, Frequency=193.75THz,
    Wavelength=1547.316nm

    Alarm Status:
    -----
    Detected Alarms:
        LOW-RX0-PWR

    LOS/LOL/Fault Status:

    Laser Bias Current = 0.0 mA
    Actual TX Power = 0.00 dBm
    RX Power = 0.00 dBm

    Performance Monitoring: Enable

    THRESHOLD VALUES
    -----

    Parameter                High Alarm  Low Alarm  High Warning  Low Warning
    -----
    Rx Power Threshold(dBm)   -2.9       -30.9     -7.0         -26.9
    Tx Power Threshold(dBm)   5.9        -5.0      2.9         -1.0
    LBC Threshold(mA)         75.00     25.00    70.00      30.00
    Temp. Threshold(celsius)  75.00     -5.00    70.00      0.00
    Voltage Threshold(volt)   3.63       2.97     3.46        3.13

    Polarization parameters not supported by optics

    Temperature = 38.00 Celsius
    Voltage = 3.28 V

Transceiver Vendor Details

Form Factor : SFP+
Vendor Info
-----
Optics type   : SFP+ 10G DWDM Tunable
Name         : CISCO-OCLARO
OUI Number   : 00.0b.40
Part Number  : TRS7080FNCCA033
Rev Number   : 0000
Serial Number : ONT2038009E
PID         : DWDM-SFP10G-C
VID         : V01

// DWDM Channel to Frequency/Wavelength Mapping
RP/0/RP0/CPU0:ios#show controllers optics 0/0/0/16 dwdm-carrier-map
Tue Sep  5 08:26:31.175 UTC

```

DWDM Carrier Band:: (null)
 MSA ITU channel range supported: 1~96

DWDM Carrier Map table

ITU Ch Num	G.694.1 Ch Num	Frequency (THz)	Wavelength (nm)
1	-35	191.35	1566.723
2	-34	191.40	1566.314
3	-33	191.45	1565.905
4	-32	191.50	1565.496
5	-31	191.55	1565.087
6	-30	191.60	1564.679
7	-29	191.65	1564.271
8	-28	191.70	1563.863
9	-27	191.75	1563.455
10	-26	191.80	1563.047
11	-25	191.85	1562.640
12	-24	191.90	1562.233
13	-23	191.95	1561.826
14	-22	192.00	1561.419
15	-21	192.05	1561.013
16	-20	192.10	1560.606
17	-19	192.15	1560.200
18	-18	192.20	1559.794
19	-17	192.25	1559.389
20	-16	192.30	1558.983
21	-15	192.35	1558.578
22	-14	192.40	1558.173
23	-13	192.45	1557.768
24	-12	192.50	1557.363
25	-11	192.55	1556.959
26	-10	192.60	1556.555
27	-9	192.65	1556.151
28	-8	192.70	1555.747

29	-7	192.75	1555.343
30	-6	192.80	1554.940
31	-5	192.85	1554.537
32	-4	192.90	1554.134
33	-3	192.95	1553.731
34	-2	193.00	1553.329
35	-1	193.05	1552.926
36	0	193.10	1552.524
37	1	193.15	1552.122
38	2	193.20	1551.721
39	3	193.25	1551.319
40	4	193.30	1550.918
41	5	193.35	1550.517
42	6	193.40	1550.116
43	7	193.45	1549.715
44	8	193.50	1549.315
45	9	193.55	1548.915
46	10	193.60	1548.515
47	11	193.65	1548.115
48	12	193.70	1547.715
49	13	193.75	1547.316
50	14	193.80	1546.917
51	15	193.85	1546.518
52	16	193.90	1546.119
53	17	193.95	1545.720
54	18	194.00	1545.322
55	19	194.05	1544.924
56	20	194.10	1544.526
57	21	194.15	1544.128
58	22	194.20	1543.730
59	23	194.25	1543.333
60	24	194.30	1542.936

61	25	194.35	1542.539
62	26	194.40	1542.142
63	27	194.45	1541.746
64	28	194.50	1541.349
65	29	194.55	1540.953
66	30	194.60	1540.557
67	31	194.65	1540.162
68	32	194.70	1539.766
69	33	194.75	1539.371
70	34	194.80	1538.976
71	35	194.85	1538.581
72	36	194.90	1538.186
73	37	194.95	1537.792
74	38	195.00	1537.397
75	39	195.05	1537.003
76	40	195.10	1536.609
77	41	195.15	1536.216
78	42	195.20	1535.822
79	43	195.25	1535.429
80	44	195.30	1535.036
81	45	195.35	1534.643
82	46	195.40	1534.250
83	47	195.45	1533.858
84	48	195.50	1533.465
85	49	195.55	1533.073
86	50	195.60	1532.681
87	51	195.65	1532.290
88	52	195.70	1531.898
89	53	195.75	1531.507
90	54	195.80	1531.116
91	55	195.85	1530.725
92	56	195.90	1530.334

```

    93    57                195.95                1529.944
-----
    94    58                196.00                1529.553
-----
    95    59                196.05                1529.163
-----
    96    60                196.10                1528.773

```

```
// Change Frequency
```

```

RP/0/RP0/CPU0:ios#conf t
Tue Sep  5 08:34:14.312 UTC
RP/0/RP0/CPU0:ios(config)#controller optics 0/0/0/16
RP/0/RP0/CPU0:ios(config-Optics)#shutdown
RP/0/RP0/CPU0:ios(config-Optics)#dwdm-carrier 50GHz-grid frequency 19335
RP/0/RP0/CPU0:ios(config-Optics)#commit
Tue Sep  5 08:34:39.943 UTC
RP/0/RP0/CPU0:ios(config-Optics)#end
RP/0/RP0/CPU0:ios#show controllers optics 0/0/0/16
Tue Sep  5 08:34:42.824 UTC

```

```
Controller State: Administratively Down
```

```
Transport Admin State: Out Of Service
```

```
Laser State: Off
```

```
LED State: Off
```

```
Optics Status
```

```

Optics Type: SFP+ 10G DWDM Tunable
DWDM carrier Info: C BAND, MSA ITU Channel=41, Frequency=193.35THz,
Wavelength=1550.517nm

```

```
Alarm Status:
```

```
-----
```

```
Detected Alarms:
```

```
    LOW-RX0-PWR
```

```
LOS/LOL/Fault Status:
```

```
Laser Bias Current = 0.0 mA
```

```
Actual TX Power = 0.00 dBm
```

```
RX Power = 0.00 dBm
```

```
Performance Monitoring: Enable
```

```
THRESHOLD VALUES
```

```
-----
```

Parameter	High Alarm	Low Alarm	High Warning	Low Warning
Rx Power Threshold(dBm)	-2.9	-30.9	-7.0	-26.9
Tx Power Threshold(dBm)	5.9	-5.0	2.9	-1.0
LBC Threshold(mA)	75.00	25.00	70.00	30.00
Temp. Threshold(celsius)	75.00	-5.00	70.00	0.00
Voltage Threshold(volt)	3.63	2.97	3.46	3.13

```
Polarization parameters not supported by optics
```

```
Temperature = 39.00 Celsius
```

```
Voltage = 3.28 V
```

Transceiver Vendor Details

```

Form Factor : SFP+
Vendor Info
-----
Optics type   : SFP+ 10G DWDM Tunable
Name          : CISCO-OCLARO
OUI Number    : 00.0b.40
Part Number   : TRS7080FNCCA033
Rev Number    : 0000
Serial Number : ONT2038009B
PID           : DWDM-SFP10G-C
VID           : V01
    
```

// Change Wavelength

```

RP/0/RP0/CPU0:ios#conf t
Tue Sep  5 11:27:21.614 UTC
RP/0/RP0/CPU0:ios(config)#controller optics 0/0/0/16
RP/0/RP0/CPU0:ios(config-Optics)#shutdown
RP/0/RP0/CPU0:ios(config-Optics)#dwdm-carrier 50GHz-grid wavelength 1539766
RP/0/RP0/CPU0:ios(config-Optics)#commit
Tue Sep  5 11:28:14.547 UTC
RP/0/RP0/CPU0:ios(config-Optics)#end
RP/0/RP0/CPU0:ios#show controllers optics 0/0/0/16
Tue Sep  5 11:28:30.934 UTC
    
```

Controller State: Administratively Down

Transport Admin State: Out Of Service

Laser State: Off

LED State: Off

Optics Status

```

Optics Type: SFP+ 10G DWDM Tunable
DWDM carrier Info: C BAND, MSA ITU Channel=68, Frequency=194.70THz,
Wavelength=1539.766nm
    
```

Alarm Status:

```

-----
Detected Alarms:
    LOW-RX0-PWR
    
```

LOS/LOL/Fault Status:

```

Laser Bias Current = 0.0 mA
Actual TX Power = 0.00 dBm
RX Power = 0.00 dBm
    
```

Performance Monitoring: Enable

THRESHOLD VALUES

Parameter	High Alarm	Low Alarm	High Warning	Low Warning
Rx Power Threshold(dBm)	-2.9	-30.9	-7.0	-26.9
Tx Power Threshold(dBm)	5.9	-5.0	2.9	-1.0
LBC Threshold(mA)	75.00	25.00	70.00	30.00
Temp. Threshold(celsius)	75.00	-5.00	70.00	0.00
Voltage Threshold(volt)	3.63	2.97	3.46	3.13

```

Polarization parameters not supported by optics

Temperature = 38.00 Celsius
Voltage = 3.28 V

Transceiver Vendor Details

Form Factor : SFP+
Vendor Info
-----
Optics type      : SFP+ 10G DWDM Tunable
Name             : CISCO-OCLARO
OUI Number       : 00.0b.40
Part Number      : TRS7080FNCCA033
Rev Number       : 0000
Serial Number    : ONT2038009E
PID              : DWDM-SFP10G-C
VID              : V01

// Change Channel
RP/0/RP0/CPU0:ios#conf t
Tue Sep  5 08:29:03.648 UTC
RP/0/RP0/CPU0:ios(config)#controller optics 0/0/0/16
RP/0/RP0/CPU0:ios(config-Optics)#shutdown
RP/0/RP0/CPU0:ios(config-Optics)#dwdm-carrier 50GHz-grid ?
  frequency      Configure Frequency and Map to ITU Channel
  itu-ch         Configure the ITU 50GHz Grid ITU Channel
  wavelength     Configure Wavelength and Map to ITU Channel
RP/0/RP0/CPU0:ios(config-Optics)#dwdm-carrier 50GHz-grid itu-ch 84
RP/0/RP0/CPU0:ios(config-Optics)#commit

RP/0/RP0/CPU0:ios#show controllers optics 0/0/0/16
Tue Sep  5 08:29:54.851 UTC

Controller State: Administratively Down

Transport Admin State: Out Of Service

Laser State: Off

LED State: Off

Optics Status

Optics Type: SFP+ 10G DWDM Tunable
DWDM carrier Info: C BAND, MSA ITU Channel=84, Frequency=195.50THz,
Wavelength=1533.465nm

Alarm Status:
-----
Detected Alarms:
    LOW-RX0-PWR

LOS/LOL/Fault Status:

Laser Bias Current = 0.0 mA
Actual TX Power = 0.00 dBm
RX Power = 0.00 dBm

Performance Monitoring: Enable

THRESHOLD VALUES
-----

```

Parameter	High Alarm	Low Alarm	High Warning	Low Warning
Rx Power Threshold(dBm)	-2.9	-30.9	-7.0	-26.9
Tx Power Threshold(dBm)	5.9	-5.0	2.9	-1.0
LBC Threshold(mA)	75.00	25.00	70.00	30.00
Temp. Threshold(celsius)	75.00	-5.00	70.00	0.00
Voltage Threshold(volt)	3.63	2.97	3.46	3.13

Polarization parameters not supported by optics

Temperature = 38.00 Celsius
Voltage = 3.28 V

Transceiver Vendor Details

```

Form Factor : SFP+
Vendor Info
-----
Optics type   : SFP+ 10G DWDM Tunable
Name          : CISCO-OCLARO
OUI Number    : 00.0b.40
Part Number   : TRS7080FNCCA033
Rev Number    : 0000
Serial Number  : ONT2038009B
PID           : DWDM-SFP10G-C
VID           : V01

```

Priority Flow Control (PFC)

Priority flow control (PFC; IEEE 802.1Qbb), which is also referred to as Class-based Flow Control (CBFC) or Per Priority Pause (PPP), is a mechanism that prevents frame loss that is due to congestion. PFC is similar to 802.3x Flow Control (pause frames) or link-level flow control (LLFC). However, PFC functions on a per class-of-service (CoS) basis.

During congestion, PFC sends a pause frame that indicates which CoS value needs to be paused. A PFC pause frame contains a 2-octet timer value for each CoS that indicates the length of time that the traffic needs to be paused. The unit of time for the timer is specified in pause quanta. A quanta is the time that is required for transmitting 512 bits at the speed of the port. The range is from 0 to 65535.



Note The router sends out the required amount of pause frames or pause-threshold (x-off) messages to achieve lossless queues. It also sends out resume-threshold (x-on) messages.

PFC asks the peer to stop sending frames of a particular CoS value by sending a pause frame to a well-known multicast address. This pause frame is a one-hop frame that is not forwarded when received by the peer. When the congestion is mitigated, the router stops sending the PFC frames to the upstream node.



Note

- PFC Rx traffic processing is enabled only if `hw-module profile priority-flow-control` command is enabled on the line card.
- `CISCO-PFC-EXT-MIB` is supported.

The PFC feature is only supported on the following line card or fixed chassis PIDs of the NCS5500 Series:

- NC55-36X100G
- NC55-18H18F
- NC55-24X100G-SE
- NC55-36X100G-S
- NC55-24H12F-SE
- NC55-36X100G-A-SE
- NCS-55A1-36H-SE-S
- NCS-55A1-36H-S
- NCS-55A1-24H
- NCS-55A1-48Q6H

Restrictions for PFC

PFC has the following restrictions:

- PFC for transmit is not supported for internal traffic (recycle / loopback) and non-unicast traffic (broadcast / multicast).
- PFC for receive impacts all traffic meant to go out of the port. This may cause unintended drops to both unicast and non-unicast traffic because non-unicast traffic may consume buffer descriptors, thus starving unicast traffic. Hence, PFC is incompatible with sustained high rate non-unicast traffic in the system.
- PFC configuration will enable or disable both PFC transmit and receive functionalities. There is no support to enable only transmit or receive functions.
- PFC is only supported in the non-HQoS profile. For more details on this QoS prerequisite and configuration examples, please refer to *Modular QoS Configuration Guide for Cisco NCS 5500 Series Routers*.
- Link level flow control and PFC are not supported on the same interface simultaneously.
- PFC is only qualified on 40G and 100G physical interface types. PFC is not supported on breakout ports for these interface types and is not qualified on other interface types.
- Being an Ethernet feature, PFC has to be individually configured on the member interfaces of a bundle instead of the bundle interface. The user is expected to either enable or disable PFC on all members of the bundle, as a mix isn't supported.
- Pause frames are not counted in the interface's statistical information, and the **show interfaces** command displays this counter as 0. Use the **show controllers Ethernet-interface-type interface-path-id** command to retrieve the PFC statistics.

Configuring Priority Flow Control

Use the following steps to configure Priority Flow Control:

Configuration:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(configure)#interface HundredGigE0/0/0/0
RP/0/RP0/CPU0:router(config)# priority-flow-control mode on
```

Running configuration:

```
*Interface Level*
interface HundredGigE0/0/0/0
priority-flow-control mode on
```

Verification:

Sample output for show controllers hundredGigE 0/0/0/0 priority-flow-control command is:

```
RP/0/RP1/CPU0:router#show controllers hundredGigE 0/0/0/0 priority-flow-control
Mon Oct 12 12:20:53.520 UTC
```

Priority flow control information for interface HundredGigE0/0/0/0:

```
Priority Flow Control:
Total Rx PFC Frames: 0
Total Tx PFC Frames: 1764273
Rx Data Frames Dropped: 0
CoS  Status  Rx Frames  Tx Frames
---  -
0  off      0          0
1  off      0          0
2  off      0          0
3  on       0          882032
4  on       0          882241
5  off      0          0
6  off      0          0
7  off      0          0
```

Priority flow control watchdog configuration:
(D) : Default value
U : Unconfigured

Configuration Item	Global	Interface	Effective
PFC watchdog state	: U	U	Enabled(D)
Poll interval	: U	U	100(D)
Shutdown multiplier	: U	U	1(D)
Auto-restore multiplier	: U	U	10(D)

Priority flow control watchdog statistics:
SAR: Auto restore and shutdown

Traffic Class	0	1	2	3	4	5	6	7
Watchdog Events	: 0	0	0	0	0	0	0	0
Shutdown Events	: 0	0	0	0	0	0	0	0
Auto Restore Events	: 0	0	0	0	0	0	0	0
SAR Events	: 0	0	0	0	0	0	0	0
SAR Instantaneous Events	: 0	0	0	0	0	0	0	0
Total Dropped Packets	: 0	0	0	0	0	0	0	0

```
Dropped Packets      :      0      0      0      0      0      0      0
0
```

Priority flow control watchdog state machine state:

```
D - Disabled
M - Monitoring
S - Waiting For Shutdown
R - Waiting to Restore
```

```
-----
PFC Watchdog      : Enabled
Watchdog SM state : Traffic Class
                   7 6 5 4 3 2 1 0
                   - - - D D - - -
```

RP/0/RP1/CPU0:router#

Sample output for `show controllers hundredGigE 0/0/0/0 priority-flow-control statistics` command is:

```
RP/0/RP1/CPU0:router#show controllers hundredGigE 0/0/0/0 priority-flow-control statistics
```

Mon Oct 12 12:22:39.362 UTC

Priority flow control information for interface HundredGigE0/0/0/0:

```
Priority Flow Control:
Total Rx PFC Frames: 0
Total Tx PFC Frames: 1764273
Rx Data Frames Dropped: 0
CoS  Status  Rx Frames  Tx Frames
---  -
0  off      0          0
1  off      0          0
2  off      0          0
3  on       0          882032
4  on       0          882241
5  off      0          0
6  off      0          0
7  off      0          0
```

Sample output for `clear controller hundredGigE 0/0/0/0 priority-flow-control statistics [traffic-class <0-7>]` is:

```
RP/0/RP1/CPU0:router#clear controller hundredGigE 0/0/0/0 priority-flow-control statistics
traffic-class 3
```

Mon Oct 12 12:22:48.778 UTC

```
RP/0/RP1/CPU0:router#show controllers hundredGigE 0/0/0/0 priority-flow-control statistics
```

Mon Oct 12 12:22:51.097 UTC

Priority flow control information for interface HundredGigE0/0/0/0:

```
Priority Flow Control:
Total Rx PFC Frames: 0
Total Tx PFC Frames: 882241
Rx Data Frames Dropped: 0
CoS  Status  Rx Frames  Tx Frames
---  -
0  off      0          0
1  off      0          0
2  off      0          0
3  on       0          0
4  on       0          882241
5  off      0          0
6  off      0          0
```



```
7 off 0 0
RP/0/RP1/CPU0:router#
```

Optical Transport Networks

Optical Transport Network (OTN) encapsulates frames of data that allows the system to send multiple data sources on the same channel.

OTN can carry any kind of traffic and removes the restriction of the different physical network dependencies. These physical networks offer different types of services, such as Ethernet, SDH, SONET, Fiber Channel, and so on.

OTN comprises of the following switching layers:

- Time Division Multiplexing (TDM)
- Wavelength Switched Optical Network (WSON)

OTN uses the following information structures to encapsulate data:

- OTUk – where $k=1/2/2e/3/3e2/4$, is an information structure into which the system maps another information structure called ODUk ($k=1/2/2e/3/3e2/4$). The ODUk signal is the server layer signal for client signals.
- OTU2e (Data rate 11.09Gb/s) transports a 10 Gigabit Ethernet LAN PHY coming from IP/Ethernet switches and routers at full line rate (10.3 Gbit/s), as specified in G.Sup43

Restrictions and Important Guidelines

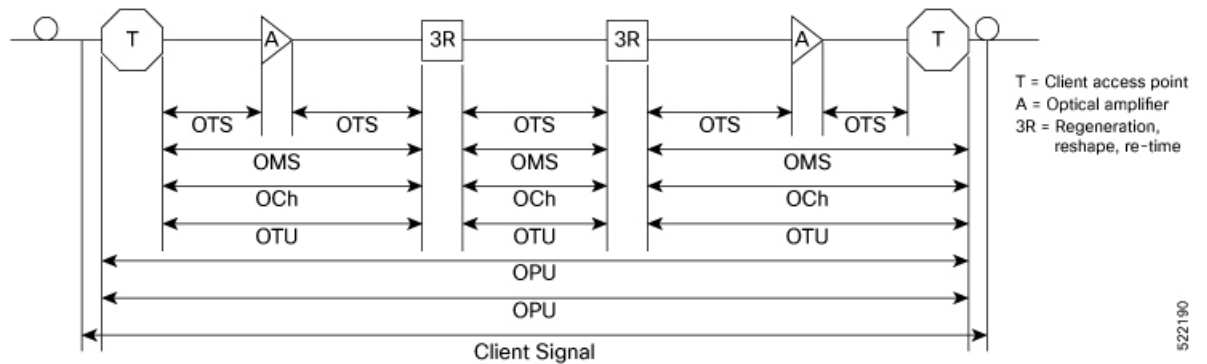
The following are some of the important guidelines and restrictions related to OTN:

- OTN is only supported on the NC55-MPA-12T-S line card.
- NC55-MPA-12T-S is supported on the following 2RU Cisco NCS 5500 Series Routers:
 - NCS-55A2-MOD-S
 - NCS-55A2-MOD-SE-S
 - NCS-55A2-MOD-HX-S
 - NCS-55A2-MOD-SE-H-S
 - NCS-55A2-MOD-HD-S
- NC55-MPA-12T-S is supported in the following Modular Line cards:
 - NC55-MOD-A-S
 - NC55-MOD-A-SE-S
- OTN isn't supported on SF_BER and SD_BER.
- OTN converts 10GE LAN PHY signal (host side) into OTU1e/OTU2e signal (line side).
- Supports 10G mapping modes

- OTU2e signal operates at 11.096 Gbps, which carries 10GE LAN PHY signal. Also known as the BMP mapping
- OTU1e signal operates at 11.049 Gbps, which carries 10GE LAN PHY signal. Also known as the BMP mapping
- Supports FEC functionality in the following modes:
 - GFEC = ITU-T G.709 / G.975, OH 7%
 - EFEC = ITU-T G.975.1 (1.4), OH 7%
 - UFEC = ITU-T G.975.1 (1.7), OH 7%
- FEC EC/UNC alarms aren't reported on NC55-MPA-12T-S.

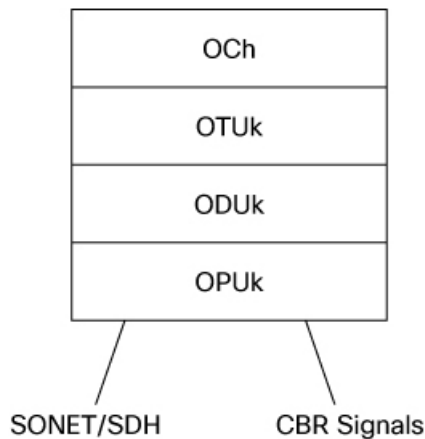
OTN Architecture

Figure 1: OTN Architecture



522190

Figure 2: OTN Layers



522191

OTN Layers and their Functions

Layer	Function	Terminated On
Optical Transport Section (OTS)	Optical transmission. Includes fiber and optical amplifier.	Optical Amplifier
Optical Multiplex Section (OMS)	Optical multiplexing. Lambda multiplexing.	MUX/DEMUX
Optical Channel (OCh)	Client Optical Adaptation	Regenerators
Optical Transport Unit (OTU)	Section Monitoring, FEC	Regenerators, Client Access Point, OTN Switch
Optical Data Unit (ODU)	Path Monitoring, Tandem Connection Monitoring	Client Access Point
Optical Channel Payload Unit (OPU)	Client Signal Adaptation	Client Access Point

Configuring OTN Interface

This section describes how you can configure OTN on an interface.

```
/* Configure "port-mode Otn framing opu2e" under controller optics of the interface. */

controller Optics0/2/1/0
port-mode Otn framing opu2e
!
```

Running Configuration

```
controller Optics0/2/1/0
port-mode Otn framing opu2e
!
```

Verification

```
RP/0/RP1/CPU0:ios#show portmode all
Sat Nov 20 21:37:01.717 UTC
Portmode Information
-----
Port Name Portmode Type Framing Mapping PT type
Rate
Optics0_2_1_0 OTN OPU2e framing type None mapping type NA
None
Optics0_2_1_11 OTN OPU2e framing type None mapping type NA
None
RP/0/RP1/CPU0:ios#

RP/0/RP1/CPU0:ios#show controllers otu2e 0/2/1/0
Sat Nov 20 21:38:17.853 UTC
Port : OTU2E 0/2/1/0
Controller State : Up
Inherited Secondary State : Normal
Configured Secondary State : Normal
Derived State : In Service
Loopback mode : None
```

```

BER Thresholds : SF = 1.0E-6 SD = 1.0E-7
Performance Monitoring : Enable
Alarm Information:
LOS = 0 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 1 BIAE = 0 SF_BER = 0
SD_BER = 0 BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0 FLEXO_GIDM = 0
FLEXO-MM = 0 FLEXO-LOM = 0 FLEXO-RDI = 0
FLEXO-LOF = 0
Detected Alarms : None
OTU TTI Received
FEC mode : STANDARD

RP/0/RP1/CPU0:ios#show controllers odu2e 0/2/1/0
Sat Nov 20 21:39:06.792 UTC
Port : ODU2E 0/2/1/0
Controller State : Up
Inherited Secondary state : Normal
Configured Secondary state : Maintenance
Derived State : Maintenance
Loopback mode : None
BER Thresholds : SF = 1.0E-6 SD = 1.0E-7
Performance Monitoring : Enable
Path Monitoring Mode : Non-Intrusive Monitor
PM TIM-CA state : Disable
Alarm Information:
AIS = 0 IAE = 0 BIAE = 0
SF_BER = 0 SD_BER = 0 BDI = 0
OCI = 0 LCK = 0 PTIM = 0
TIM = 0 CSF = 0 GFP LFD = 0
GFP LOCS = 0 GFP LOCCS = 0 GFP UPM = 0
Detected Alarms : None
ODU TTI Sent
ODU TTI Received
ODU TTI Expected
Owner : All
Resource State : ODU Resource Free
Private Line Emulation(PLE) supported : No

```

Supported Alarms

This table lists the supported OTN alarms:

ODU Alarms	OUT Alarms
AIS	OOF
OCI	AIS
LCK	LOF
BDI	LOM
	OOM
	BDI
	BIAE
	IAE

The following are the supported OTN PM counters:

- BIP
- BEI

How to Configure Interfaces in Breakout Mode

Table 7: Feature History Table

Feature name	Release Information	Feature Description
2x50GbE Breakout Ports on Cisco NCS-55A1-24H Routers	Release 7.5.2	You can now configure 2x50GbE breakout ports on all QSFP28 ports of the Cisco NCS-55A1-24H fixed port router.

Information About Breakout

The router supports transmission of traffic in the breakout mode. The breakout mode enables a 40GbE, 100GbE, or 400GbE port to be split into multiple GbE ports.

Breakout Mode options:

- 4x10GbE
- 4x25GbE
- 2x50GbE
- 8x50GbE
- 4x100GbE
- 3x100GbE
- 2x100GbE
- 1x100GbE



Note

- The supported breakout mode is dependent on the port and optic transceiver.

Configure Breakout in a Port

This example shows how to configuring a 4x10GbE breakout in a port:

```
RP/0/RP0/CPU0:Router# configure
RP/0/RP0/CPU0:Router(config)# controller optics 0/1/0/28
RP/0/RP0/CPU0:Router(config-Optics)# breakout 4x10
RP/0/RP0/CPU0:Router(config-Optics)# commit
```

```
RP/0/RP0/CPU0:Router(config-Optics)# end
RP/0/RP0/CPU0:Router#
```

Remove the Breakout Configuration

Removing the breakout configuration:

```
RP/0/RP0/CPU0:Router# configure
RP/0/RP0/CPU0:Router(config)# controller optics 0/1/0/28
RP/0/RP0/CPU0:Router(config-Optics)# no breakout 4x10
RP/0/RP0/CPU0:Router(config-Optics)# commit
RP/0/RP0/CPU0:uut(config-Optics)# end
```

Verify a Breakout Configuration

Verifying a breakout configuration:

```
RP/0/RP0/CPU0:Router# show running-config controller optics 0/1/0/28
controller Optics0/1/0/28
breakout 4x10
!
```

```
RP/0/RP0/CPU0:Router# show int br location 0/1/CPU0 | i Te0/1/0/28
Te0/1/0/27/0      up      up      ARPA 10000 10000000
Te0/1/0/27/1      up      up      ARPA 10000 10000000
Te0/1/0/27/2      up      up      ARPA 10000 10000000
Te0/1/0/27/3      up      up      ARPA 10000 10000000
Te0/1/0/28/0      up      up      ARPA 10000 10000000
Te0/1/0/28/1      up      up      ARPA 10000 10000000
Te0/1/0/28/2      up      up      ARPA 10000 10000000
Te0/1/0/28/3      up      up      ARPA 10000 10000000
```



CHAPTER 5

Configuring Ethernet OAM

This module describes the configuration of Ethernet Operations, Administration, and Maintenance (OAM) .

Feature History for Configuring Ethernet OAM

Release	Modification
Release 6.1.1	Support for the following features was introduced: <ul style="list-style-type: none">• Ethernet Link OAM• Ethernet CFM
Release 7.1.1	Support for CFM adaptive bandwidth notifications was introduced for NCS5500 platforms.
Release 7.5.1	Support for Link Loss Forwarding (LLF) was introduced.
Release 7.5.1	Support for CFM adaptive bandwidth notifications was introduced for Cisco Network Convergence System 5700 Series routers and routers with Cisco NC57 line cards operating in native mode.

- [Ethernet OAM, on page 63](#)
- [Ethernet CFM, on page 77](#)
- [CFM Adaptive Bandwidth Notifications, on page 122](#)
- [CFM Over Bundles, on page 130](#)
- [Y.1731 Performance Monitoring, on page 131](#)
- [Unidirectional Link Detection Protocol, on page 141](#)
- [Link Loss Forwarding, on page 145](#)

Ethernet OAM

To configure Ethernet OAM, you should understand the following concepts:

Ethernet Link OAM

Ethernet as a Metro Area Network (MAN) or a Wide Area Network (WAN) technology benefits greatly from the implementation of Operations, Administration and Maintenance (OAM) features. Ethernet link OAM features allow Service Providers to monitor the quality of the connections on a MAN or WAN. Service providers can monitor specific events, . Ethernet link OAM operates on a single, physical link and it can be configured to monitor either side or both sides of that link.

Ethernet link OAM can be configured in the following ways:

- A Link OAM profile can be configured, and this profile can be used to set the parameters for multiple interfaces.
- Link OAM can be configured directly on an interface.

When an interface is also using a link OAM profile, specific parameters that are set in the profile can be overridden by configuring a different value directly on the interface.

An Ethernet Link OAM profile simplifies the process of configuring EOAM features on multiple interfaces. An Ethernet OAM profile, and all of its features, can be referenced by other interfaces, allowing other interfaces to inherit the features of that Ethernet OAM profile.

Individual Ethernet link OAM features can be configured on individual interfaces without being part of a profile. In these cases, the individually configured features always override the features in the profile.

The preferred method of configuring custom EOAM settings is to create an EOAM profile in Ethernet configuration mode and then attach it to an individual interface or to multiple interfaces.

When an EOAM packet is received on any one of the AC interfaces on which EOAM is not configured, the AC interface multicasts the received EOAM packets to other AC interfaces that are part of EVPN-BD to reach the peer. When an EOAM is enabled on the bundle member in the peer, it punts the packet to the CPU in the peer. Also, the EOAM flaps the bundle member as the local or remote Key of the received EOAM does not match.

These standard Ethernet Link OAM features are supported on the router:

Neighbor Discovery

Neighbor discovery enables each end of a link to learn the OAM capabilities of the other end and establish an OAM peer relationship. Each end also can require that the peer have certain capabilities before it will establish a session. You can configure certain actions to be taken if there is a capabilities conflict or if a discovery process times out, using the **action capabilities-conflict** or **action discovery-timeout** commands.

EFD

Ethernet Fault Detection (EFD) is a mechanism that allows Ethernet OAM protocols, such as CFM, to control the `line protocol` state of an interface.

Unlike many other interface types, Ethernet interfaces do not have a line protocol, whose state is independent from that of the interface. For Ethernet interfaces, this role is handled by the physical-layer Ethernet protocol itself, and therefore if the interface is physically up, then it is available and traffic can flow.

EFD changes this to allow CFM to act as the line protocol for Ethernet interfaces. This allows CFM to control the interface state so that if a CFM defect (such as AIS or loss of continuity) is detected with an expected peer MEP, the interface can be shut down. This not only stops traffic flow, but also triggers actions in any

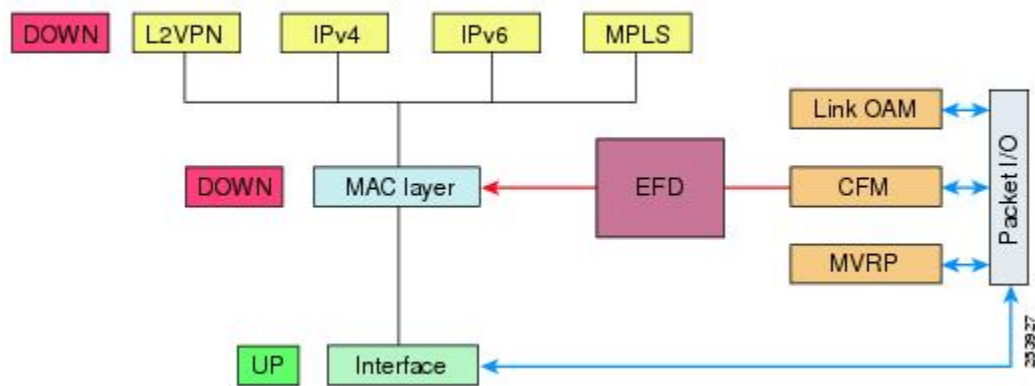
higher-level protocols to route around the problem. For example, in the case of Layer 2 interfaces, the MAC table would be cleared and MSTP would reconverge. For Layer 3 interfaces, the ARP cache would be cleared and potentially the IGP would reconverge.



Note EFD can only be used for down MEPs. When EFD is used to shut down the interface, the CFM frames continue to flow. This allows CFM to detect when the problem has been resolved, and thus bring the interface backup automatically.

This figure shows CFM detection of an error on one of its sessions EFD signaling an error to the corresponding MAC layer for the interface. This triggers the MAC to go to a down state, which further triggers all higher level protocols (Layer 2 pseudowires, IP protocols, and so on) to go down and also trigger a reconvergence where possible. As soon as CFM detects there is no longer any error, it can signal to EFD and all protocols will once again go active.

Figure 3: CFM Error Detection and EFD Trigger



MIB Retrieval

MIB retrieval enables an OAM peer on one side of an interface to get the MIB variables from the remote side of the link. The MIB variables that are retrieved from the remote OAM peer are READ ONLY.

Miswiring Detection (Cisco-Proprietary)

Miswiring Detection is a Cisco-proprietary feature that uses the 32-bit vendor field in every Information OAMPDU to identify potential miswiring cases.

SNMP Traps

SNMP traps can be enabled or disabled on an Ethernet OAM interface.

How to Configure Ethernet OAM

This section provides these configuration procedures:

Configuring Ethernet Link OAM

Custom EOAM settings can be configured and shared on multiple interfaces by creating an EOAM profile in Ethernet configuration mode and then attaching the profile to individual interfaces. The profile configuration does not take effect until the profile is attached to an interface. After an EOAM profile is attached to an interface, individual EOAM features can be configured separately on the interface to override the profile settings when desired.

This section describes how to configure an EOAM profile and attach it to an interface in these procedures:

Configuring an Ethernet OAM Profile

Perform these steps to configure an Ethernet OAM profile.

SUMMARY STEPS

1. **configure**
2. **ethernet oam profile** *profile-name*
3. **link-monitor**
4. **symbol-period window** *window*
5. **symbol-period threshold low** *threshold* **high** *threshold*
6. **frame window** *window*
7. **frame threshold low** *threshold* **high** *threshold*
8. **frame-period window** *window*
9. **frame-period threshold low***threshold* **high** *threshold*
10. **frame-seconds window** *window*
11. **frame-seconds threshold low** *threshold* **high** *threshold*
12. **exit**
13. **mib-retrieval**
14. **connection timeout** *<timeout>*
15. **hello-interval** {100ms|1s}
16. **mode** {active|passive}
17. **require-remote mode** {active|passive}
18. **require-remote mib-retrieval**
19. **action capabilities-conflict** {disable | efd | error-disable-interface}
20. **action critical-event** {disable | error-disable-interface}
21. **action discovery-timeout** {disable | efd | error-disable-interface}
22. **action dying-gasp** {disable | error-disable-interface}
23. **action high-threshold** {error-disable-interface | log}
24. **action session-down** {disable | efd | error-disable-interface}
25. **action session-up** disable
26. **action uni-directional link-fault** {disable | efd | error-disable-interface}
27. **action wiring-conflict** {disable | efd | log}
28. **uni-directional link-fault detection**

29. **commit**
30. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure terminal	Enters global configuration mode.
Step 2	ethernet oam profile <i>profile-name</i> Example: RP/0/RP0/CPU0:router(config)# ethernet oam profile Profile_1	Creates a new Ethernet Operations, Administration and Maintenance (OAM) profile and enters Ethernet OAM configuration mode.
Step 3	link-monitor Example: RP/0/RP0/CPU0:router(config-eoam)# link-monitor	Enters the Ethernet OAM link monitor configuration mode.
Step 4	symbol-period window <i>window</i> Example: RP/0/RP0/CPU0:router(config-eoam-lm)# symbol-period window 60000	(Optional) Configures the window size (in milliseconds) for an Ethernet OAM symbol-period error event. The IEEE 802.3 standard defines the window size as a number of symbols rather than a time duration. These two formats can be converted either way by using a knowledge of the interface speed and encoding. The range is 1000 to 60000. The default value is 1000.
Step 5	symbol-period threshold low <i>threshold</i> high <i>threshold</i> Example: RP/0/RP0/CPU0:router(config-eoam-lm)# symbol-period threshold ppm low 1 high 1000000	(Optional) Configures the thresholds (in symbols) that trigger an Ethernet OAM symbol-period error event. The high threshold is optional and is configurable only in conjunction with the low threshold. The range is 0 to 60000000. The default low threshold is 1.
Step 6	frame window <i>window</i> Example: RP/0/RP0/CPU0:router(config-eoam-lm)# frame window 6000	(Optional) Configures the frame window size (in milliseconds) of an OAM frame error event. The range is from 1000 to 60000. The default value is 1000.
Step 7	frame threshold low <i>threshold</i> high <i>threshold</i> Example:	(Optional) Configures the thresholds (in symbols) that triggers an Ethernet OAM frame error event. The high threshold is optional and is configurable only in conjunction with the low threshold.

	Command or Action	Purpose
	<pre>RP/0/RP0/CPU0:router(config-eoam-lm)# frame threshold low 10000000 high 60000000</pre>	<p>The range is from 0 to 60000000.</p> <p>The default low threshold is 1.</p>
Step 8	<p>frame-period window <i>window</i></p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-eoam-lm)# frame-period window 60000</pre> <pre>RP/0/RP0/CPU0:router(config-eoam-lm)# frame-period window milliseconds 60000</pre>	<p>(Optional) Configures the window size (in milliseconds) for an Ethernet OAM frame-period error event. The IEEE 802.3 standard defines the window size as number of frames rather than a time duration. These two formats can be converted either way by using a knowledge of the interface speed. Note that the conversion assumes that all frames are of the minimum size.</p> <p>The range is from 1000 to 60000.</p> <p>The default value is 1000.</p> <p>Note The only accepted values are multiples of the line card interface module-specific polling interval, that is, 1000 milliseconds for most line card interface modules.</p>
Step 9	<p>frame-period threshold low<i>threshold high</i> <i>threshold</i></p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-eoam-lm)# frame-period threshold ppm low 100 high 1000000</pre>	<p>(Optional) Configures the thresholds (in errors per million frames) that trigger an Ethernet OAM frame-period error event. The frame period window is defined in the IEEE specification as a number of received frames, in our implementation it is x milliseconds. The high threshold is optional and is configurable only in conjunction with the low threshold.</p> <p>The range is from 1 to 1000000.</p> <p>The default low threshold is 1.</p> <p>To obtain the number of frames, the configured time interval is converted to a window size in frames using the interface speed. For example, for a 1Gbps interface, the IEEE defines minimum frame size as 512 bits. So, we get a maximum of approximately 1.5 million frames per second. If the window size is configured to be 8 seconds (8000ms) then this would give us a Window of 12 million frames in the specification's definition of Errored Frame Window.</p> <p>The thresholds for frame-period are measured in errors per million frames. Hence, if you configure a window of 8000ms (that is a window of 12 million frames) and a high threshold of 100, then the threshold would be crossed if there are 1200 errored frames in that period (that is, 100 per million for 12 million).</p>
Step 10	<p>frame-seconds window <i>window</i></p> <p>Example:</p>	<p>(Optional) Configures the window size (in milliseconds) for the OAM frame-seconds error event.</p> <p>The range is 10000 to 900000.</p>

	Command or Action	Purpose
	RP/0/RP0/CPU0:router(config-eoam-lm)# frame-seconds window 900000	The default value is 60000. Note The only accepted values are multiples of the line card interface module-specific polling interval, that is, 1000 milliseconds for most line card interface modules.
Step 11	frame-seconds threshold low <i>threshold</i> high <i>threshold</i> Example: RP/0/RP0/CPU0:router(config-eoam-lm)# frame-seconds threshold low 3 high 900	(Optional) Configures the thresholds (in seconds) that trigger a frame-seconds error event. The high threshold value can be configured only in conjunction with the low threshold value. The range is 1 to 900 The default value is 1.
Step 12	exit Example: RP/0/RP0/CPU0:router(config-eoam-lm)# exit	Exits back to Ethernet OAM mode.
Step 13	mib-retrieval Example: RP/0/RP0/CPU0:router(config-eoam)# mib-retrieval	Enables MIB retrieval in an Ethernet OAM profile or on an Ethernet OAM interface.
Step 14	connection timeout <timeout> Example: RP/0/RP0/CPU0:router(config-eoam)# connection timeout 30	Configures the connection timeout period for an Ethernet OAM session, as a multiple of the hello interval. The range is 2 to 30. The default value is 5.
Step 15	hello-interval {100ms 1s} Example: RP/0/RP0/CPU0:router(config-eoam)# hello-interval 100ms	Configures the time interval between hello packets for an Ethernet OAM session. The default is 1 second (1s).
Step 16	mode {active passive} Example: RP/0/RP0/CPU0:router(config-eoam)# mode passive	Configures the Ethernet OAM mode. The default is active.
Step 17	require-remote mode {active passive} Example: RP/0/RP0/CPU0:router(config-eoam)# require-remote mode active	Requires that active mode or passive mode is configured on the remote end before the OAM session becomes active.

	Command or Action	Purpose
Step 18	require-remote mib-retrieval Example: <pre>RP/0/RP0/CPU0:router(config-eoam)# require-remote mib-retrieval</pre>	Requires that MIB-retrieval is configured on the remote end before the OAM session becomes active.
Step 19	action capabilities-conflict {disable efd error-disable-interface} Example: <pre>RP/0/RP0/CPU0:router(config-eoam)# action capabilities-conflict efd</pre>	<p>Specifies the action that is taken on an interface when a capabilities-conflict event occurs. The default action is to create a syslog entry.</p> <p>Note</p> <ul style="list-style-type: none"> If you change the default, the log keyword option is available in Interface Ethernet OAM configuration mode to override the profile setting and log the event for the interface when it occurs.
Step 20	action critical-event {disable error-disable-interface} Example: <pre>RP/0/RP0/CPU0:router(config-eoam)# action critical-event error-disable-interface</pre>	<p>Specifies the action that is taken on an interface when a critical-event notification is received from the remote Ethernet OAM peer. The default action is to create a syslog entry.</p> <p>Note</p> <ul style="list-style-type: none"> If you change the default, the log keyword option is available in Interface Ethernet OAM configuration mode to override the profile setting and log the event for the interface when it occurs.
Step 21	action discovery-timeout {disable efd error-disable-interface} Example: <pre>RP/0/RP0/CPU0:router(config-eoam)# action discovery-timeout efd</pre>	<p>Specifies the action that is taken on an interface when a connection timeout occurs. The default action is to create a syslog entry.</p> <p>Note</p> <ul style="list-style-type: none"> If you change the default, the log keyword option is available in Interface Ethernet OAM configuration mode to override the profile setting and log the event for the interface when it occurs.
Step 22	action dying-gasp {disable error-disable-interface} Example: <pre>RP/0/RP0/CPU0:router(config-eoam)# action dying-gasp error-disable-interface</pre>	<p>Specifies the action that is taken on an interface when a dying-gasp notification is received from the remote Ethernet OAM peer. The default action is to create a syslog entry.</p> <p>Note</p> <ul style="list-style-type: none"> If you change the default, the log keyword option is available in Interface Ethernet OAM configuration mode to override the profile setting and log the event for the interface when it occurs.

	Command or Action	Purpose
Step 23	<p>action high-threshold {error-disable-interface log}</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-eoam)# action high-threshold error-disable-interface</pre>	<p>Specifies the action that is taken on an interface when a high threshold is exceeded. The default is to take no action when a high threshold is exceeded.</p> <p>Note</p> <ul style="list-style-type: none"> If you change the default, the disable keyword option is available in Interface Ethernet OAM configuration mode to override the profile setting and take no action at the interface when the event occurs.
Step 24	<p>action session-down {disable efd error-disable-interface}</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-eoam)# action session-down efd</pre>	<p>Specifies the action that is taken on an interface when an Ethernet OAM session goes down.</p> <p>Note</p> <ul style="list-style-type: none"> If you change the default, the log keyword option is available in Interface Ethernet OAM configuration mode to override the profile setting and log the event for the interface when it occurs.
Step 25	<p>action session-up disable</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-eoam)# action session-up disable</pre>	<p>Specifies that no action is taken on an interface when an Ethernet OAM session is established. The default action is to create a syslog entry.</p> <p>Note</p> <ul style="list-style-type: none"> If you change the default, the log keyword option is available in Interface Ethernet OAM configuration mode to override the profile setting and log the event for the interface when it occurs.
Step 26	<p>action uni-directional link-fault {disable efd error-disable-interface}</p>	<p>Specifies the action that is taken on an interface when a link-fault notification is received from the remote Ethernet OAM peer. The default action is to create a syslog entry.</p> <p>Note</p> <ul style="list-style-type: none"> If you change the default, the log keyword option is available in Interface Ethernet OAM configuration mode to override the profile setting and log the event for the interface when it occurs.
Step 27	<p>action wiring-conflict {disable efd log}</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-eoam)# action session-down efd</pre>	<p>Specifies the action that is taken on an interface when a wiring-conflict event occurs. The default is to put the interface into error-disable state.</p> <p>Note</p> <ul style="list-style-type: none"> If you change the default, the error-disable-interface keyword option is available in Interface Ethernet OAM configuration mode to override the profile setting and put the interface into error-disable state when the event occurs.

	Command or Action	Purpose
Step 28	uni-directional link-fault detection Example: <pre>RP/0/RP0/CPU0:router(config-eoam)# uni-directional link-fault detection</pre>	Enables detection of a local, unidirectional link fault and sends notification of that fault to an Ethernet OAM peer.
Step 29	commit Example: <pre>RP/0/RP0/CPU0:router(config-if)# commit</pre>	Saves the configuration changes to the running configuration file and remains within the configuration session.
Step 30	end Example: <pre>RP/0/RP0/CPU0:router(config-if)# end</pre>	Ends the configuration session and exits to the EXEC mode.

Attaching an Ethernet OAM Profile to an Interface

Perform these steps to attach an Ethernet OAM profile to an interface:

SUMMARY STEPS

1. **configure**
2. **interface** [FastEthernet | HundredGigE | TenGigE] *interface-path-id*
3. **ethernet oam**
4. **profile** *profile-name*
5. **commit**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: <pre>RP/0/RP0/CPU0:router# configure terminal</pre>	Enters global configuration mode.
Step 2	interface [FastEthernet HundredGigE TenGigE] <i>interface-path-id</i> Example: <pre>RP/0/RP0/CPU0:router(config)# interface TenGigE 0/1/0/0</pre>	Enters interface configuration mode and specifies the Ethernet interface name and notation <i>rack/slot/module/port</i> . Note <ul style="list-style-type: none"> • The example indicates an 8-port 10-Gigabit Ethernet interface in modular services card slot 1.
Step 3	ethernet oam Example:	Enables Ethernet OAM and enters interface Ethernet OAM configuration mode.

	Command or Action	Purpose
	<pre>RP/0/RP0/CPU0:router(config-if)# ethernet oam</pre>	
Step 4	<p>profile <i>profile-name</i></p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-if-eoam)# profile Profile_1</pre>	Attaches the specified Ethernet OAM profile (<i>profile-name</i>), and all of its configuration, to the interface.
Step 5	<p>commit</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-if)# commit</pre>	Saves the configuration changes to the running configuration file and remains within the configuration session.
Step 6	<p>end</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-if)# end</pre>	Ends the configuration session and exits to the EXEC mode.

Configuring Ethernet OAM at an Interface and Overriding the Profile Configuration

Using an EOAM profile is an efficient way of configuring multiple interfaces with a common EOAM configuration. However, if you want to use a profile but also change the behavior of certain functions for a particular interface, then you can override the profile configuration. To override certain profile settings that are applied to an interface, you can configure that command in interface Ethernet OAM configuration mode to change the behavior for that interface.

In some cases, only certain keyword options are available in interface Ethernet OAM configuration due to the default settings for the command. For example, without any configuration of the **action** commands, several forms of the command have a default behavior of creating a syslog entry when a profile is created and applied to an interface. Therefore, the **log** keyword is not available in Ethernet OAM configuration for these commands in the profile because it is the default behavior. However, the **log** keyword is available in Interface Ethernet OAM configuration if the default is changed in the profile configuration so you can retain the action of creating a syslog entry for a particular interface.

To see all of the default Ethernet OAM configuration settings, see the [Verifying the Ethernet OAM Configuration](#).

To configure Ethernet OAM settings at an interface and override the profile configuration, perform these steps:

SUMMARY STEPS

1. **configure**
2. **interface** [HundredGigE | TenGigE] *interface-path-id*
3. **ethernet oam**
4. *interface-Ethernet-OAM-command*
5. **commit**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure terminal	Enters global configuration mode.
Step 2	interface [HundredGigE TenGigE] <i>interface-path-id</i> Example: RP/0/RP0/CPU0:router (config)# interface TenGigE 0/1/0/0	Enters interface configuration mode and specifies the Ethernet interface name and notation <i>rack/slot/module/port</i> . Note • The example indicates an 8-port 10-Gigabit Ethernet interface in modular services card slot 1.
Step 3	ethernet oam Example: RP/0/RP0/CPU0:router(config-if)# ethernet oam	Enables Ethernet OAM and enters interface Ethernet OAM configuration mode.
Step 4	<i>interface-Ethernet-OAM-command</i> Example: RP/0/RP0/CPU0:router (config-if- <i>eoam</i>)# action capabilities-conflict error-disable-interface	Configures a setting for an Ethernet OAM configuration command and overrides the setting for the profile configuration, where <i>interface-Ethernet-OAM-command</i> is one of the supported commands on the platform in interface Ethernet OAM configuration mode.
Step 5	commit Example: RP/0/RP0/CPU0:router (config-if)# commit	Saves the configuration changes to the running configuration file and remains within the configuration session.
Step 6	end Example: RP/0/RP0/CPU0:router (config-if)# end	Ends the configuration session and exits to the EXEC mode.

Verifying the Ethernet OAM Configuration

Use the **show ethernet oam configuration** command to display the values for the Ethernet OAM configuration for a particular interface, or for all interfaces. The following example shows the default values for Ethernet OAM settings:

```
RP/0/RP0/CPU0:router# show ethernet oam configuration
Thu Aug  5 22:07:06.870 DST
GigabitEthernet0/4/0/0:
  Hello interval:                               1s
  Mib retrieval enabled:                         N
  Uni-directional link-fault detection enabled:  N
  Configured mode:                               Active
  Connection timeout:                             5
  Symbol period window:                           0
```

Symbol period low threshold:	1
Symbol period high threshold:	None
Frame window:	1000
Frame low threshold:	1
Frame high threshold:	None
Frame period window:	1000
Frame period low threshold:	1
Frame period high threshold:	None
Frame seconds window:	60000
Frame seconds low threshold:	1
Frame seconds high threshold:	None
High threshold action:	None
Link fault action:	Log
Dying gasp action:	Log
Critical event action:	Log
Discovery timeout action:	Log
Capabilities conflict action:	Log
Wiring conflict action:	Error-Disable
Session up action:	Log
Session down action:	Log
Require remote mode:	Ignore
Require remote MIB retrieval:	N

Configuration Examples for Ethernet OAM

This section provides the following configuration examples:

Configuration Examples for EOAM Interfaces

This section provides the following configuration examples:

Configuring an Ethernet OAM Profile Globally: Example

This example shows how to configure an Ethernet OAM profile globally:

```

configure terminal
 ethernet oam profile Profile_1
  link-monitor
  symbol-period window 60000
  symbol-period threshold ppm low 10000000 high 60000000
  frame window 60
  frame threshold ppm low 10000000 high 60000000
  frame-period window 60000
  frame-period threshold ppm low 100 high 12000000
  frame-seconds window 900000
  frame-seconds threshold low 3 high 900
  exit
 mib-retrieval
 connection timeout 30
 require-remote mode active
 require-remote mib-retrieval
 action dying-gasp error-disable-interface
 action critical-event error-disable-interface
 action discovery-timeout error-disable-interface
 action session-down error-disable-interface
 action capabilities-conflict error-disable-interface
 action wiring-conflict error-disable-interface
 action remote-loopback error-disable-interface
 commit

```

Configuring Ethernet OAM Features on an Individual Interface: Example

This example shows how to configure Ethernet OAM features on an individual interface:

```
configure terminal
interface TenGigE 0/1/0/0
  ethernet oam
    link-monitor
      symbol-period window 60000
      symbol-period threshold ppm low 10000000 high 60000000
      frame window 60
      frame threshold ppm low 10000000 high 60000000
      frame-period window 60000
      frame-period threshold ppm low 100 high 12000000
      frame-seconds window 900000
      frame-seconds threshold low 3 high 900
    exit
  mib-retrieval
    connection timeout 30
    require-remote mode active
    require-remote mib-retrieval
    action link-fault error-disable-interface
    action dying-gasp error-disable-interface
    action critical-event error-disable-interface
    action discovery-timeout error-disable-interface
    action session-down error-disable-interface
    action capabilities-conflict error-disable-interface
    action wiring-conflict error-disable-interface
    action remote-loopback error-disable-interface
  commit
```

Configuring Ethernet OAM Features to Override the Profile on an Individual Interface: Example

This example shows the configuration of Ethernet OAM features in a profile followed by an override of that configuration on an interface:

```
configure terminal
ethernet oam profile Profile_1
  mode passive
  action dying-gasp disable
  action critical-event disable
  action discovery-timeout disable
  action session-up disable
  action session-down disable
  action capabilities-conflict disable
  action wiring-conflict disable
  action remote-loopback disable
  action uni-directional link-fault error-disable-interface
  commit

configure terminal
interface TenGigE 0/1/0/0
  ethernet oam
    profile Profile_1
    mode active
    action dying-gasp log
    action critical-event log
    action discovery-timeout log
    action session-up log
    action session-down log
    action capabilities-conflict log
    action wiring-conflict log
```

```

action remote-loopback log
action uni-directional link-fault log
uni-directional link-fault detection
commit

```

Clearing Ethernet OAM Statistics on an Interface: Example

This example shows how to clear Ethernet OAM statistics on an interface:

```

RP/0/RP0/CPU0:router# clear ethernet oam statistics interface gigabitethernet 0/1/5/1

```

Enabling SNMP Server Traps on a Router: Example

This example shows how to enable SNMP server traps on a router:

```

configure terminal
snmp-server traps ethernet oam events

```

Ethernet CFM

Table 8: Feature History Table

Feature Name	Release	Description
Cisco NC57 Native Mode: CFM	Release 7.3.1	This feature is now supported on routers that have the Cisco NC57 line cards installed and operate in the native mode. To enable the native mode, use the hw-module profile npu native-mode-enable command in the configuration mode. Ensure that you reload the router after configuring the native mode.
Cisco NC57 Compatibility Mode: CFM	Release 7.4.1	This feature is now supported on routers that have the Cisco NC57 line cards installed and operate in the compatibility mode.

Feature Name	Release	Description
Support for Link Loss Forwarding on Cisco NCS 5500 Series Routers	Release 7.5.1	This feature, now available on Cisco NCS 5500 Series Routers, enables high availability between two bridged interfaces by disabling both interfaces if any one of them fails. This functionality allows a fault detected on one side of a CFM-protected network to propagate to the other side, enabling the device to re-route around the failure at that end. In earlier releases, a failure on one bridged interface did not disable the other interface, and connected devices remained unaware of the link loss.

Ethernet Connectivity Fault Management (CFM) is a service-level OAM protocol that provides tools for monitoring and troubleshooting end-to-end Ethernet services per VLAN. This includes proactive connectivity monitoring, fault verification, and fault isolation. CFM uses standard Ethernet frames and can be run on any physical media that is capable of transporting Ethernet service frames. Unlike most other Ethernet protocols which are restricted to a single physical link, CFM frames can transmit across the entire end-to-end Ethernet network.



Note Enable a maximum of 32 VLAN ranges per NPU. Else, when you reload the device, all CFM sessions over the 802.1Q VLAN interface might go down. Also, the corresponding bundle interface might go down. If more than 32 VLAN ranges exist on an NPU, remove the additional VLAN ranges and reload the device to address the issue. This is not applicable for NCS 5700 line cards.



Note Up MEP with Cisco NC57 line cards installed and operate in the native and compatibility modes as a part of Layer 2 service. When you have have NC57 line card (compatibility mode) interface as core facing (ingress) and NC57 line card as the AC (egress) interface, the up mep CFM session does not come up.

CFM is defined in two standards:

- IEEE 802.1ag—Defines the core features of the CFM protocol.
- ITU-T Y.1731—Redefines, but maintains compatibility with the features of IEEE 802.1ag, and defines some additional features.

Ethernet CFM supports these functions of ITU-T Y.1731:

- ETH-CC, ETH-RDI, ETH-LB, ETH-LT, ETH-BNM, ETH-CSF—These are equivalent to the corresponding features defined in IEEE 802.1ag.



Note The Linktrace responder procedures defined in IEEE 802.1ag are used rather than the procedures defined in Y.1731; however, these are interoperable.

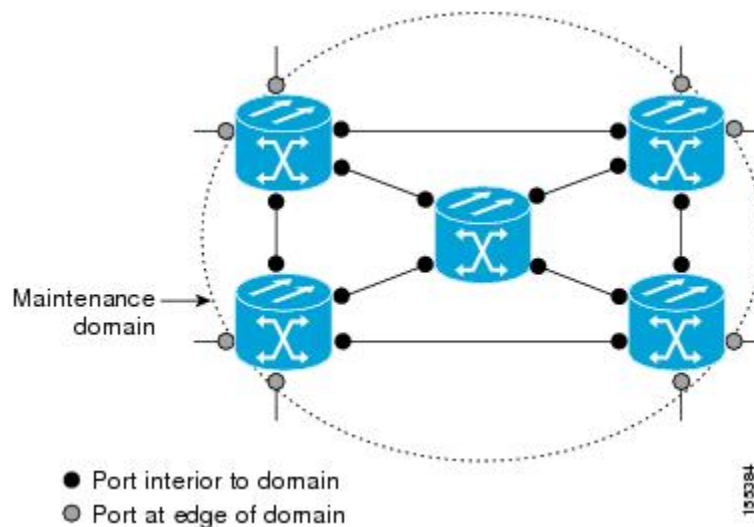
- ETH-AIS—The reception of ETH-LCK messages is also supported.

To understand how the CFM maintenance model works, you need to understand these concepts and features:

Maintenance Domains

A maintenance domain describes a management space for the purpose of managing and administering a network. A domain is owned and operated by a single entity and defined by the set of interfaces internal to it and at its boundary, as shown in this figure.

Figure 4: CFM Maintenance Domain



A maintenance domain is defined by the bridge ports that are provisioned within it. Domains are assigned maintenance levels, in the range of 0 to 7, by the administrator. The level of the domain is useful in defining the hierarchical relationships of multiple domains.

CFM maintenance domains allow different organizations to use CFM in the same network, but independently. For example, consider a service provider who offers a service to a customer, and to provide that service, they use two other operators in segments of the network. In this environment, CFM can be used in the following ways:

- The customer can use CFM between their CE devices, to verify and manage connectivity across the whole network.
- The service provider can use CFM between their PE devices, to verify and manage the services they are providing.
- Each operator can use CFM within their operator network, to verify and manage connectivity within their network.

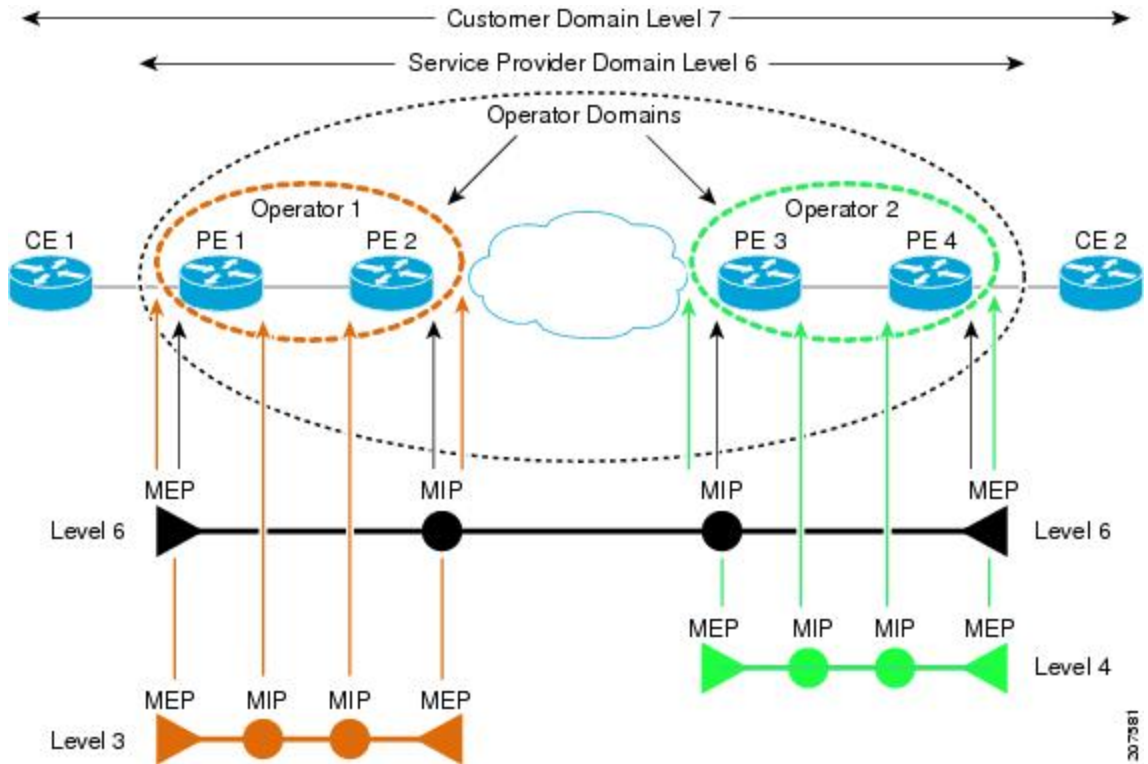
Each organization uses a different CFM maintenance domain.

This figure shows an example of the different levels of maintenance domains in a network.



Note In CFM diagrams, the conventions are that triangles represent MEPs, pointing in the direction that the MEP sends CFM frames, and circles represent MIPs.

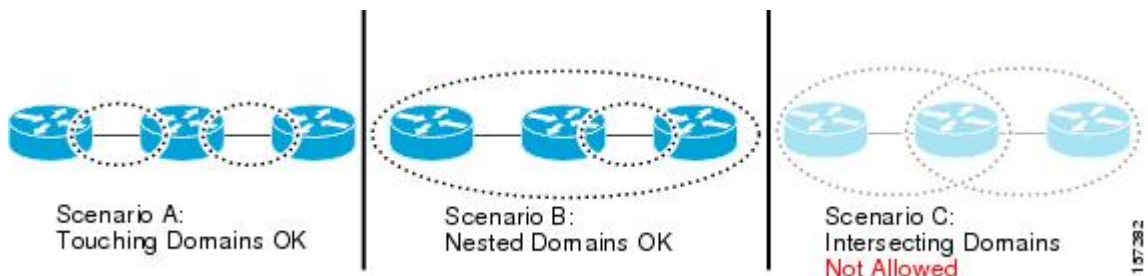
Figure 5: Different CFM Maintenance Domains Across a Network



To ensure that the CFM frames for each domain do not interfere with each other, each domain is assigned a maintenance level, between 0 and 7. Where domains are nested, as in this example, the encompassing domain must have a higher level than the domain it encloses. In this case, the domain levels must be negotiated between the organizations involved. The maintenance level is carried in all CFM frames that relate to that domain.

CFM maintenance domains may touch or nest, but cannot intersect. This figure illustrates the supported structure for touching and nested domains, and the unsupported intersection of domains.

Figure 6: Supported CFM Maintenance Domain Structure



Services

A CFM service allows an organization to partition its CFM maintenance domain, according to the connectivity within the network. For example, if the network is divided into a number of virtual LANs (VLANs), a CFM service is created for each of these. CFM can then operate independently in each service. It is important that the CFM services match the network topology, so that CFM frames relating to one service cannot be received in a different service. For example, a service provider may use a separate CFM service for each of their customers, to verify and manage connectivity between that customer's end points.

A CFM service is always associated with the maintenance domain that it operates within, and therefore with that domain's maintenance level. All CFM frames relating to the service carry the maintenance level of the corresponding domain.



Note CFM Services are referred to as *Maintenance Associations* in IEEE 802.1ag and as *Maintenance Entity Groups* in ITU-T Y.1731.

Maintenance Points

A CFM Maintenance Point (MP) is an instance of a particular CFM service on a specific interface. CFM only operates on an interface if there is a CFM maintenance point on the interface; otherwise, CFM frames are forwarded transparently through the interface.

A maintenance point is always associated with a particular CFM service, and therefore with a particular maintenance domain at a particular level. Maintenance points generally only process CFM frames at the same level as their associated maintenance domain. Frames at a higher maintenance level are always forwarded transparently, while frames at a lower maintenance level are normally dropped. This helps enforce the maintenance domain hierarchy, and ensures that CFM frames for a particular domain cannot leak out beyond the boundary of the domain.

There are two types of MP:

- Maintenance End Points (MEPs)—Created at the edge of the domain. Maintenance end points (MEPs) are members of a particular service within a domain and are responsible for sourcing and sinking CFM frames. They periodically transmit continuity check messages and receive similar messages from other MEPs within their domain. They also transmit traceroute and loopback messages at the request of the administrator. MEPs are responsible for confining CFM messages within the domain.
- Maintenance Intermediate Points (MIPs)—Created in the middle of the domain. Unlike MEPS, MIPs do allow CFM frames at their own level to be forwarded.

MIP Creation

Unlike MEPs, MIPs are not explicitly configured on each interface. MIPs are created automatically according to the algorithm specified in the CFM 802.1ag standard. The algorithm, in brief, operates as follows for each interface:

- The bridge-domain or cross-connect for the interface is found, and all services associated with that bridge-domain or cross-connect are considered for MIP auto-creation.

- The level of the highest-level MEP on the interface is found. From among the services considered above, the service in the domain with the lowest level that is higher than the highest MEP level is selected. If there are no MEPs on the interface, the service in the domain with the lowest level is selected.
- The MIP auto-creation configuration (**mip auto-create** command) for the selected service is examined to determine whether a MIP should be created.



Note Configuring a MIP auto-creation policy for a service does not guarantee that a MIP will automatically be created for that service. The policy is only considered if that service is selected by the algorithm first.

MEP and CFM Processing Overview

The boundary of a domain is an interface, rather than a bridge or host. Therefore, MEPs can be sub-divided into two categories:

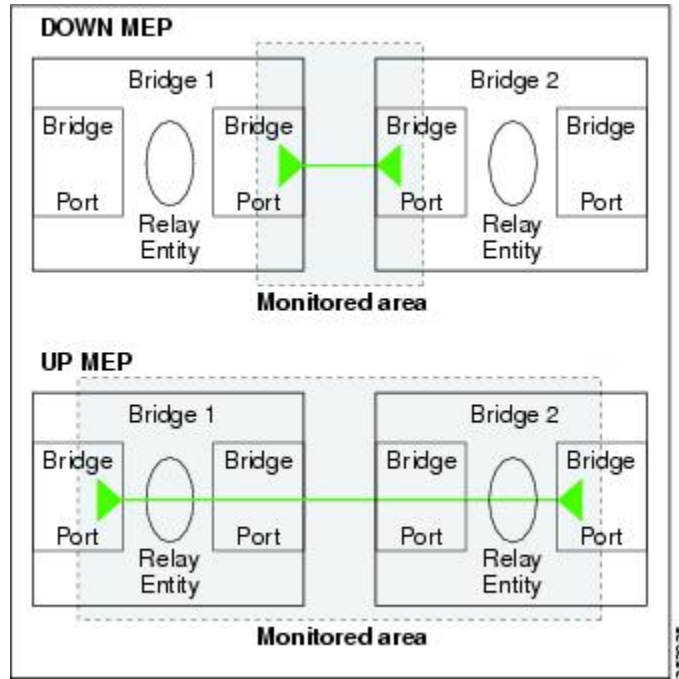
- Down MEPs—Send CFM frames from the interface where they are configured, and process CFM frames received on that interface. Down MEPs transmit AIS messages upward (toward the cross-connect).
- Up MEPs—Send frames into the bridge relay function, as if they had been received on the interface where the MEP is configured. They process CFM frames that have been received on other interfaces, and have been switched through the bridge relay function as if they are going to be sent out of the interface where the MEP is configured. Up MEPs transmit AIS messages downward (toward the wire). However, AIS packets are only sent when there is a MIP configured on the same interface as the MEP and at the level of the MIP.



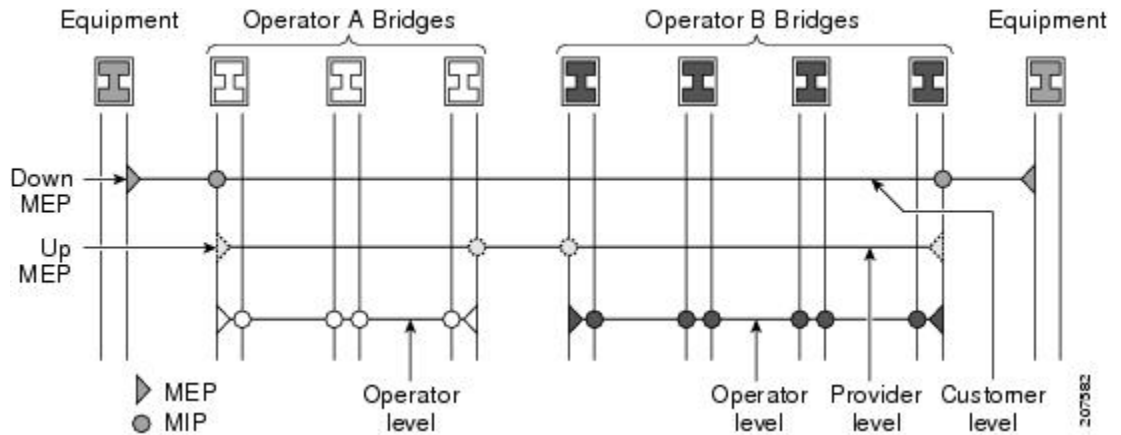
Note The terms *Down MEP* and *Up MEP* are defined in the IEEE 802.1ag and ITU-T Y.1731 standards, and refer to the direction that CFM frames are sent from the MEP. The terms should not be confused with the operational status of the MEP.

This figure illustrates the monitored areas for Down and Up MEPs.

Figure 7: Monitored Areas for Down and Up MEPs



This figure shows maintenance points at different levels. Because domains are allowed to nest but not intersect (see [Supported CFM Maintenance Domain Structure](#)), a MEP at a low level always corresponds with a MEP or MIP at a higher level. In addition, only a single MIP is allowed on any interface—this is generally created in the lowest domain that exists at the interface and that does not have a MEP.



MIPs and Up MEPs can only exist on switched (Layer 2) interfaces, because they send and receive frames from the bridge relay function. Down MEPs can be created on switched (Layer 2) interfaces.

MEPs continue to operate normally if the interface they are created on is blocked by the Spanning Tree Protocol (STP); that is, CFM frames at the level of the MEP continue to be sent and received, according to the direction of the MEP. MEPs never allow CFM frames at the level of the MEP to be forwarded, so the STP block is maintained.

MIPs also continue to receive CFM frames at their level if the interface is STP blocked, and can respond to any received frames. However, MIPs do not allow CFM frames at the level of the MIP to be forwarded if the interface is blocked.



Note A separate set of CFM maintenance levels is created every time a VLAN tag is pushed onto the frame. Therefore, if CFM frames are received on an interface which pushes an additional tag, so as to “tunnel” the frames over part of the network, the CFM frames will not be processed by any MPs within the tunnel, even if they are at the same level. For example, if a CFM MP is created on an interface with an encapsulation that matches a single VLAN tag, any CFM frames that are received at the interface that have two VLAN tags will be forwarded transparently, regardless of the CFM level.

CFM Protocol Messages

The CFM protocol consists of a number of different message types, with different purposes. All CFM messages use the CFM EtherType, and carry the CFM maintenance level for the domain to which they apply.

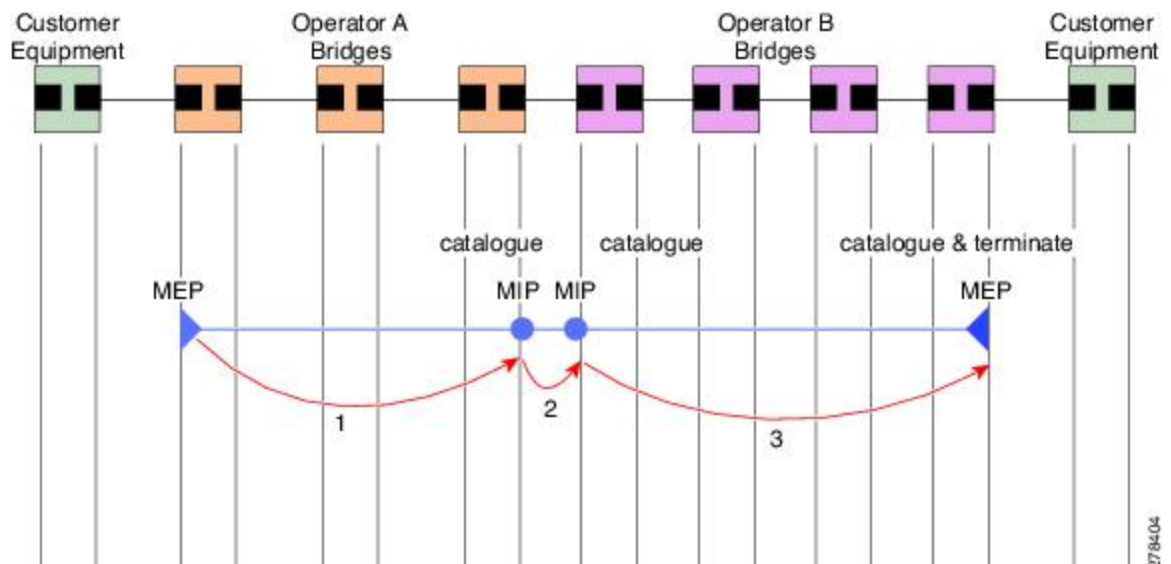
This section describes the following CFM messages:

Continuity Check (IEEE 802.1ag and ITU-T Y.1731)

Continuity Check Messages (CCMs) are “heartbeat” messages exchanged periodically between all the MEPs in a service. Each MEP sends out multicast CCMs, and receives CCMs from all the other MEPs in the service—these are referred to as *peer MEPs*. This allows each MEP to discover its peer MEPs, and to verify that there is connectivity between them.

MIPs also receive CCMs. MIPs use the information to build a MAC learning database that is used when responding to Linktrace. For more information about Linktrace, see the [Linktrace \(IEEE 802.1ag and ITU-T Y.1731\)](#).

Figure 8: Continuity Check Message Flow



278404

All the MEPs in a service must transmit CCMs at the same interval. The interval at which CCMs are being transmitted is called CCM interval. IEEE 802.1ag defines 7 possible intervals that can be used:

- 3.3ms
- 10ms
- 100ms
- 1s
- 10s
- 1 minute
- 10 minutes

A MEP detects a loss of connectivity with one of its peer MEPs when a certain number of CCMs have been missed. This occurs when sufficient time has passed during which a certain number of CCMs were expected, given the CCM interval. This number is called the *loss threshold*, and is usually set to 3.

CFM is supported only on interfaces which have Layer 2 transport feature enabled.

Maintenance Association Identifier (MAID)

Table 9: Feature History Table

Feature Name	Release	Description
48 byte string-based MAID support for Offloaded Endpoints	Release 7.5.1	<p>This feature is supported on Cisco Network Convergence System 5700 Series routers and routers with the Cisco NC57 line cards operating in native mode. This feature extends MAID functionality to support the flexible format for hardware offloaded MEPs. This removes the restrictions on the type of MAID supported for sessions with less than 1 minute time intervals.</p> <p>To enable the feature in native mode, run the Ethernet CFM Show Command for flexible packet format: Examples command in the System Admin Config mode, and reload the router.</p>

Continuity Check Messages (CCM) are essential for detecting various defects in network services. They carry critical information that helps in the identification and maintenance of the service. This is a breakdown of the information contained in CCM messages:

- **Maintenance Domain Identifier (MDID):** A configured identifier unique to the domain of the transmitting Maintenance End Point (MEP). It is crucial for the identification of the maintenance domain.

- **Short MA Name (SMAN):** A configured identifier specific to the service of the transmitting MEP. It is used to identify the service within the maintenance domain.
- **Maintenance Association Identifier (MAID):** A combination of MDID and SMAN. Together, these identifiers form the MAID, which is a composite identifier that must be uniformly configured across all MEPs within the same service.



Note MDID **only** supports **null** value and SMAN supports ITU Carrier Code (ICC) or a numerical. No other values are supported.

Supported MAID Formats for Offloaded MEPs

- No Domain Name Format
 - MD Name Format = 1-NoDomainName
 - Short MA Name Format = 3 - 2 bytes integer value
 - Short MA Name Length = 2 - fixed length
 - Short MA Name = 2 bytes of integer
- 1731 Maid Format
 - MD Name Format = 1-NoDomainName
 - MA Name Format(MEGID Format) = 32
 - MEGID Length = 13 - fixed length
 - MEGID(ICCCode) = 6 Bytes
 - MEGID(UMC) = 7 Bytes
 - ITU Carrier Code (ICC) - Number of different configurable ICC code - 15 (for each NPU)
 - Unique MEG ID Code (UMC) - 4

These are some examples:

- Configuring domain ID null: **ethernet cfm domain SMB level 3 id null**
- Configuring SMAN: **ethernet cfm domain SMB level 3 id null service 901234AB xconnect group 99999 p2p 99999 id number 1**

This table summarizes the supported values and parameters for MDID and SMAN. This table only details the MAID restriction on the hardware offload feature. There is no MAID restriction for software offload or non-offloaded MEPs.

For Cisco NCS 5500 series routers, "id null" has to be explicitly configured for the domain ID, for hardware offloaded sessions.

Format	MDID	SMAN	Support	Comment
	No	2 byte integer	Yes	Up to 2000 entries

Format	MDID	SMAN	Support	Comment
	No	13 bytes ICCCode (6 bytes) and UMC (7 bytes)	Yes	Up to 15 unique ICC Up to 4K UMC values
48 bytes string based	1-48 bytes of MDID and SMAN		No	Most commonly used

Guidelines and Restrictions for MAID

- Configure each MEP within the service with a distinct MEP ID, which is a unique numeric identifier.
- Configure MEP CrossCheck for all MEPs with intervals of less than 1 minute, as Dynamic Remote MEPs are not supported for these.
- In a Remote Defect Indication (RDI), each MEP includes sequence number in the CCMs it is sending, if it has detected a defect relating to the CCMs it is receiving. This notifies all the MEPs in the service that a defect has been detected somewhere in the service. Sequence numbering is not supported for MEPs with CCM intervals of less than 1 minute.
- CCM Tx/Rx statistics counters are not supported for MEPs with less than 1 minute intervals.
- Sender TLV and Cisco Proprietary TLVs are not supported for MEPs with less than 1min intervals.
- Starting from Cisco IOS XR Software Release 7.5.1, MAID supports the flexible packet format of MEG IDs on hardware offloaded MEPs for the following Cisco NC57 line cards:
 - NC57-24DD
 - NCS-57C3-MODS-SYS

This feature is supported only on Cisco NC57 line cards installed and operate in native mode. It removes the restrictions on the type of MAID that are supported for sessions with less than 1 minute time intervals. This helps in interoperating with the devices that already support the flexible format configuration.

Examples:

- Configuring domain ID: **ethernet cfm domain SMB level 3 id string** or
ethernet cfm domain SMB level 3
- Configuring SMAN: **ethernet cfm domain SMB level 3 id null service 901234AB xconnect group 99999 p2p 99999 id string** or
ethernet cfm domain SMB level 3 id null service 901234AB xconnect group 99999 p2p 99999
- The status of the interface where the MEP is operating (for example, up - when the interface is up, or down - when the interface is down) should not be confused with the direction of any MEPs on the interface (Up MEPs/Down MEPs).

Defect Identification using CCM Analysis

These defects can be detected from the received CCMs:

- Interval mismatch: The CCM interval in the received CCM does not match the interval that the MEP is configured to send CCMs.
- Level mismatch: A MEP receives a CCM carrying a lower maintenance level than the MEP's own configured level.
- Loop: A CCM is received with a source MAC address that matches the MAC address of the MEP's operating interface, indicating a loop.
- Configuration error: A received CCM contains a MEP ID that duplicates the MEP ID of the receiving MEP, signaling a configuration issue.
- Cross-connect error: A CCM with a non-matching MAID is received, often pointing to a VLAN misconfiguration that causes service leakage.
- Peer interface down: A CCM is received that indicates the interface on the peer is down.
- Remote defect indication: A CCM is received carrying a remote defect indication. This does not trigger the local MEP to send out CCMs with a remote defect indication.

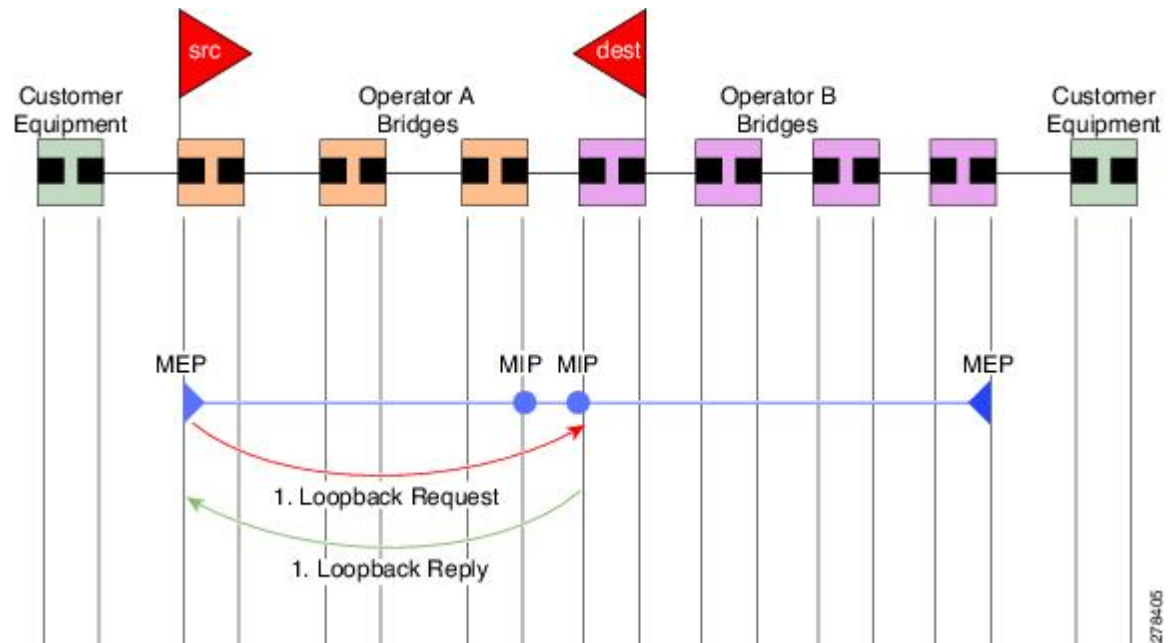
By monitoring the sequence numbers in CCMs from peer MEPs, out-of-sequence CCMs can be identified, although these are not classified as CCM defects.

Loopback (IEEE 802.1ag and ITU-T Y.1731)

Loopback Messages (LBM) and Loopback Replies (LBR) are used to verify connectivity between a local MEP and a particular remote MP. At the request of the administrator, a local MEP sends unicast LBMs to the remote MP. On receiving each LBM, the target maintenance point sends an LBR back to the originating MEP. Loopback indicates whether the destination is reachable or not—it does not allow hop-by-hop discovery of the path. It is similar in concept to an ICMP Echo (ping). Since loopback messages are destined for unicast addresses, they are forwarded like normal data traffic, while observing the maintenance levels. At each device that the loopback reaches, if the outgoing interface is known (in the bridge's forwarding database), then the frame is sent out on that interface. If the outgoing interface is not known, then the message is flooded on all interfaces.

This figure shows an example of CFM loopback message flow between a MEP and MIP.

Figure 9: Loopback Messages



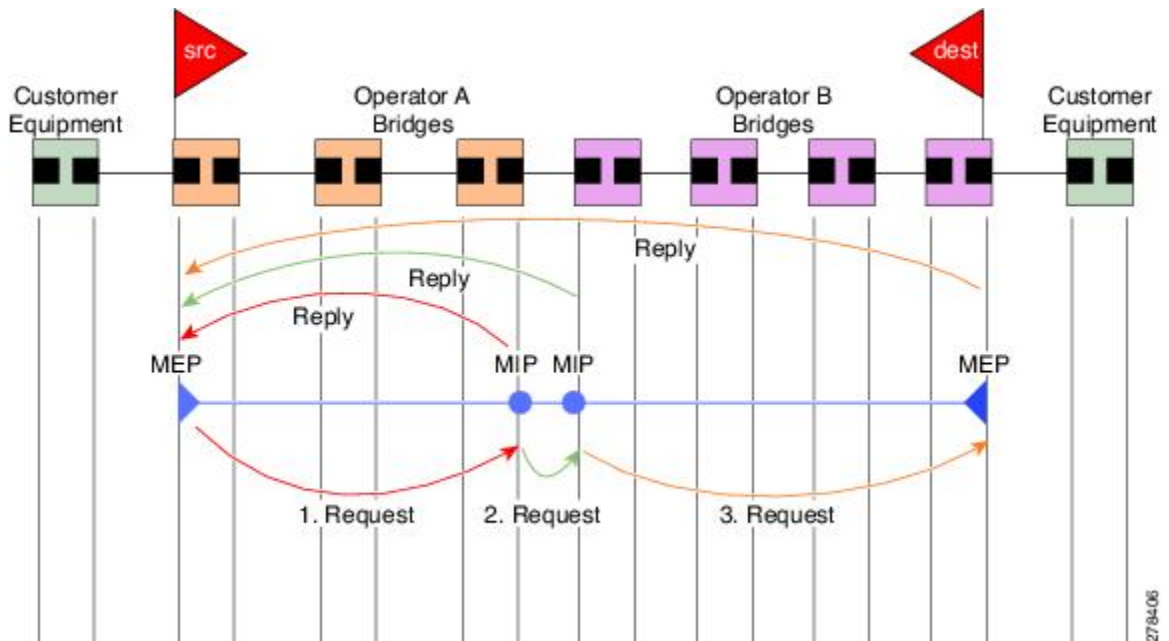
Loopback messages can be padded with user-specified data. This allows data corruption to be detected in the network. They also carry a sequence number which allows for out-of-order frames to be detected.

Linktrace (IEEE 802.1ag and ITU-T Y.1731)

Linktrace Messages (LTM) and Linktrace Replies (LTR) are used to track the path (hop-by-hop) to a unicast destination MAC address. At the request of the operator, a local MEP sends an LTM. Each hop where there is a maintenance point sends an LTR back to the originating MEP. This allows the administrator to discover connectivity data about the path. It is similar in concept to IP traceroute, although the mechanism is different. In IP traceroute, successive probes are sent, whereas CFM Linktrace uses a single LTM which is forwarded by each MP in the path. LTMs are multicast, and carry the unicast target MAC address as data within the frame. They are intercepted at each hop where there is a maintenance point, and either retransmitted or dropped to discover the unicast path to the target MAC address.

This figure shows an example of CFM linktrace message flow between MEPs and MIPs.

Figure 10: Linktrace Message Flow



The linktrace mechanism is designed to provide useful information even after a network failure. This allows it to be used to locate failures, for example after a loss of continuity is detected. To achieve this, each MP maintains a CCM Learning Database. This maps the source MAC address for each received CCM to the interface through which the CCM was received. It is similar to a typical bridge MAC learning database, except that it is based only on CCMs and it times out much more slowly—on the order of days rather than minutes.



Note In IEEE 802.1ag, the CCM Learning Database is referred to as the MIP CCM Database. However, it applies to both MIPs and MEPs.

In IEEE 802.1ag, when an MP receives an LTM message, it determines whether to send a reply using the following steps:

1. The target MAC address in the LTM is looked up in the bridge MAC learning table. If the MAC address is known, and therefore the egress interface is known, then an LTR is sent.
2. If the MAC address is not found in the bridge MAC learning table, then it is looked up in the CCM learning database. If it is found, then an LTR is sent.
3. If the MAC address is not found, then no LTR is sent (and the LTM is not forwarded).

If the target MAC has never been seen previously in the network, the linktrace operation will not produce any results.



Note IEEE 802.1ag and ITU-T Y.1731 define slightly different linktrace mechanisms. In particular, the use of the CCM learning database and the algorithm described above for responding to LTM messages are specific to IEEE 802.1ag. IEEE 802.1ag also specifies additional information that can be included in LTRs. Regardless of the differences, the two mechanisms are interoperable.

Configurable Logging

CFM supports logging of various conditions to syslog. Logging can be enabled independently for each service, and when the following conditions occur:

- New peer MEPs are detected, or loss of continuity with a peer MEP occurs.
- Changes to the CCM defect conditions are detected.
- Cross-check “missing” or “unexpected” conditions are detected.
- AIS condition detected (AIS messages received) or cleared (AIS messages no longer received).
- EFD used to shut down an interface, or bring it back up.

Flexible VLAN Tagging for CFM

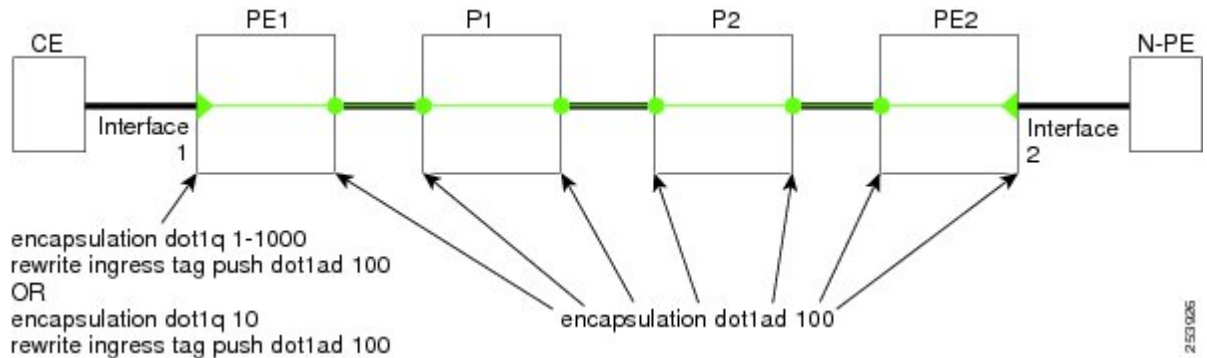
The Flexible VLAN Tagging for CFM feature ensures that CFM packets are sent with the right VLAN tags so that they are appropriately handled as a CFM packet by the remote device. When packets are received by an edge router, they are treated as either CFM packets or data packets, depending on the number of tags in the header. The system differentiates between CFM packets and data packets based on the number of tags in the packet, and forwards the packets to the appropriate paths based on the number of tags in the packet.

CFM frames are normally sent with the same VLAN tags as the corresponding customer data traffic on the interface, as defined by the configured encapsulation and tag rewrite operations. Likewise, received frames are treated as CFM frames if they have the correct number of tags as defined by the configured encapsulation and tag rewrite configuration, and are treated as data frames (that is, they are forwarded transparently) if they have more than this number of tags.

In most cases, this behavior is as desired, since the CFM frames are then treated in exactly the same way as the data traffic flowing through the same service. However, in a scenario where multiple customer VLANs are multiplexed over a single multipoint provider service (for example, N:1 bundling), a different behavior might be desirable.

This figure shows an example of a network with multiple VLANS using CFM.

Figure 11: Service Provider Network With Multiple VLANs and CFM



This figure shows a provider's access network, where the S-VLAN tag is used as the service delimiter. PE1 faces the customer, and PE2 is at the edge of the access network facing the core. N:1 bundling is used, so the interface encapsulation matches a range of C-VLAN tags. This could potentially be the full range, resulting in all:1 bundling. There is also a use case where only a single C-VLAN is matched, but the S-VLAN is nevertheless used as the service delimiter—this is more in keeping with the IEEE model, but limits the provider to 4094 services.

CFM is used in this network with a MEP at each end of the access network, and MIPs on the boxes within the network (if it is native Ethernet). In the normal case, CFM frames are sent by the up MEP on PE1 with two VLAN tags, matching the customer data traffic. This means that at the core interfaces and at the MEP on PE2, the CFM frames are forwarded as if they were customer data traffic, since these interfaces match only on the S-VLAN tag. So, the CFM frames sent by the MEP on PE1 are not seen by any of the other MIPs.

Flexible VLAN tagging changes the encapsulation for CFM frames that are sent and received at Up MEPs. Flexible VLAN tagging allows the frames to be sent from the MEP on PE1 with just the S-VLAN tag that represents the provider service. If this is done, the core interfaces will treat the frames as CFM frames and they will be seen by the MIPs and by the MEP on PE2. Likewise, the MEP on PE1 should handle received frames with only one tag, as this is what it will receive from the MEP on PE2.

To ensure that CFM packets from Up MEPs are routed to the appropriate paths successfully, tags may be set to a specific number in a domain service, using the **tags** command. Currently, tags can only be set to one (1).

Configuring Ethernet CFM

To configure Ethernet CFM, perform the following tasks:



Note CFM is not supported for the following:

- L3 Interfaces and Sub-Interfaces
- Bundle Member Ports
- Bridge Domain
- CFM over BGP-VPLS is supported only for NCS 5700 line cards.

In addition, for NC57 line cards, CFM is also not supported for:

- EVPN
- EVPN-Virtual Private Wire Service (VPWS)

Configuring a CFM Maintenance Domain

To configure a CFM maintenance domain, perform the following steps:

SUMMARY STEPS

1. **configure**
2. **ethernet cfm**
3. **traceroute cache hold-time** *minutes* **size** *entries*
4. **domain** *domain-name* **level** *level-value* [**id** **null**] [**dns** *DNS-name*] [**mac** *H.H.H*] [**string** *string*]
5. **end** or **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	ethernet cfm Example: RP/0/RP0/CPU0:router(config)# <code>ethernet cfm</code>	Enters Ethernet Connectivity Fault Management (CFM) configuration mode.
Step 3	traceroute cache hold-time <i>minutes</i> size <i>entries</i> Example: RP/0/RP0/CPU0:router(config-cfm)# <code>traceroute cache hold-time 1 size 3000</code>	(Optional) Sets the maximum limit of traceroute cache entries or the maximum time limit to hold the traceroute cache entries. The default is 100 minutes and 100 entries.
Step 4	domain <i>domain-name</i> level <i>level-value</i> [id null] [dns <i>DNS-name</i>] [mac <i>H.H.H</i>] [string <i>string</i>]	Creates and names a container for all domain configurations and enters CFM domain configuration mode.

	Command or Action	Purpose
	<p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-cfm)# domain Domain_One level 1 id string D1</pre>	<p>The level must be specified.</p> <p>The id is the maintenance domain identifier (MDID) and is used as the first part of the maintenance association identifier (MAID) in CFM frames. If the MDID is not specified, the domain name is used as the MDID by default.</p>
Step 5	<p>end or commit</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-cfm-dmn)# commit</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> When you use 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. 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 Services for a CFM Maintenance Domain

You can configure up to 32000 CFM services for a maintenance domain. To configure services for a CFM maintenance domain, perform the following steps:

SUMMARY STEPS

- configure**
- ethernet cfm**
- domain** *domain-name* **level** *level-value* [**id** [null]] [**dns** *DNS-name*] [**mac** *H.H.H*] [**string** *string*]]
- service** *service-name* {**down-meps** | **xconnect group** *xconnect-group-name* **m2mp** | **p2p** *xconnect-name*} [**id** [**icc-based** *icc-string umc-string*] | [[**number** *number*]]
- end or 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	ethernet cfm Example: <pre>RP/0/RP0/CPU0:router(config)# ethernet cfm</pre>	Enters Ethernet CFM configuration mode.
Step 3	domain <i>domain-name</i> level <i>level-value</i> [id [null]] [dns <i>DNS-name</i>] [mac <i>H.H.H</i>] [string <i>string</i>] Example: <pre>RP/0/RP0/CPU0:router(config-cfm)# domain Domain_One level 1 id string D1</pre>	<p>Creates and names a container for all domain configurations at a specified maintenance level, and enters CFM domain configuration mode.</p> <p>The id is the maintenance domain identifier (MDID) and is used as the first part of the maintenance association identifier (MAID) in CFM frames. If the MDID is not specified, the domain name is used as the MDID by default.</p>
Step 4	service <i>service-name</i> {down-meps xconnect group <i>xconnect-group-name</i> m2mp p2p <i>xconnect-name</i>} [id [icc-based <i>icc-string</i> <i>umc-string</i>] [number <i>number</i>]] Example: <pre>RP/0/RP0/CPU0:router(config-cfm-dmn)# service ABC xconnect group X1 p2p ADB</pre>	<p>Configures and associates a service with the domain and enters CFM domain service configuration mode. You can specify that the service is used only for down MEPs, or associate the service with a bridge domain where MIPs and up MEPs will be created.</p> <p>The id sets the short MA name.</p>
Step 5	end or commit Example: <pre>RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# commit</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> When you use 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. 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.

Enabling and Configuring Continuity Check for a CFM Service

To configure Continuity Check for a CFM service, complete the following steps:

SUMMARY STEPS

1. **configure**
2. **ethernet cfm**
3. **domain** *domain-name* **level** *level-value* [**id** [null] [**dns** *DNS-name*] [**mac** *H.H.H*] [**string** *string*]]
4. **service** *service-name* {**down-meps** | **xconnect group** *xconnect-group-name* **p2p** *xconnect-name*} [**id** [**icc-based** *icc-string umc-string*] | [**number** *number*]
5. **continuity-check interval** *time* [**loss-threshold** *threshold*]
6. **continuity-check archive hold-time** *minutes*
7. **continuity-check loss auto-traceroute**
8. **end** or **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	ethernet cfm Example: RP/0/RP0/CPU0:router(config)# ethernet cfm	Enters Ethernet Connectivity Fault Management (CFM) configuration mode.
Step 3	domain <i>domain-name</i> level <i>level-value</i> [id [null] [dns <i>DNS-name</i>] [mac <i>H.H.H</i>] [string <i>string</i>]] Example: RP/0/RP0/CPU0:router(config-cfm)# domain Domain_One level 1 id string D1	<p>Creates and names a container for all domain configurations and enters the CFM domain configuration mode.</p> <p>The level must be specified.</p> <p>The id is the maintenance domain identifier (MDID) and is used as the first part of the maintenance association identifier (MAID) in CFM frames. If the MDID is not specified, the domain name is used as the MDID by default.</p>
Step 4	service <i>service-name</i> { down-meps xconnect group <i>xconnect-group-name</i> p2p <i>xconnect-name</i> } [id [icc-based <i>icc-string umc-string</i>] [number <i>number</i>] Example:	Configures and associates a service with the domain and enters CFM domain service configuration mode. You can specify that the service is used only for down MEPs, or associate the service with a bridge domain or xconnect where MIPs and up MEPs will be created.

	Command or Action	Purpose
	<pre>RP/0/RP0/CPU0:router(config-cfm-dmn)# service ABC xconnect group X1 p2p ADB</pre>	The id sets the short MA name.
Step 5	<p>continuity-check interval <i>time</i> [loss-threshold <i>threshold</i>]</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# continuity-check interval 100m loss-threshold 10</pre>	(Optional) Enables Continuity Check and specifies the time interval at which CCMs are transmitted or to set the threshold limit for when a MEP is declared down.
Step 6	<p>continuity-check archive hold-time <i>minutes</i></p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# continuity-check archive hold-time 100</pre>	(Optional) Configures how long information about peer MEPs is stored after they have timed out.
Step 7	<p>continuity-check loss auto-traceroute</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# continuity-check loss auto-traceroute</pre>	(Optional) Configures automatic triggering of a traceroute when a MEP is declared down.
Step 8	<p>end or commit</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# commit</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> When you use 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. 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 Automatic MIP Creation for a CFM Service

For more information about the algorithm for creating MIPs, see the **MIP Creation** section.

To configure automatic MIP creation for a CFM service, complete the following steps:

SUMMARY STEPS

1. **configure**
2. **ethernet cfm**
3. **domain** *domain-name* **level** *level-value* [**id** [null] [dns *DNS-name*] [**mac** *H.H.H*] [**string** *string*]]
4. **service** *service-name* {**down-meps** | **xconnect group** *xconnect-group-name* **p2p** *xconnect-name*} [**id** [**icc-based***icc-string umc-string*] | [**number** *number*]
5. **mip auto-create** {**all** | **lower-mep-only**} {**ccm-learning**}
6. **end** or **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	ethernet cfm Example: RP/0/RP0/CPU0:router# ethernet cfm	Enters the Ethernet Connectivity Fault Management (CFM) configuration mode.
Step 3	domain <i>domain-name</i> level <i>level-value</i> [id [null] [dns <i>DNS-name</i>] [mac <i>H.H.H</i>] [string <i>string</i>]] Example: RP/0/RP0/CPU0:router(config-cfm)# domain Domain_One level 1 id string D1	Creates and names a container for all domain configurations and enters the CFM domain configuration mode. The level must be specified. The only supported option is id [null] for less than 1min interval MEPS. The id is the maintenance domain identifier (MDID) and is used as the first part of the maintenance association identifier (MAID) in CFM frames. If the MDID is not specified, the domain name is used as the MDID by default.
Step 4	service <i>service-name</i> { down-meps xconnect group <i>xconnect-group-name</i> p2p <i>xconnect-name</i> } [id [icc-based <i>icc-string umc-string</i>] [number <i>number</i>] Example: RP/0/RP0/CPU0:router(config-cfm-dmn)# service ABC xconnect group X1 p2p ADB	Configures and associates a service with the domain and enters CFM domain service configuration mode. You can specify that the service is used only for down MEPs, or associate the service with a bridge domain where MIPs and up MEPs will be created. The id sets the short MA name.
Step 5	mip auto-create { all lower-mep-only } { ccm-learning } Example: RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# mip auto-create all ccm-learning	(Optional) Enables the automatic creation of MIPs in a bridge domain. ccm-learning option enables CCM learning for MIPs created in this service. This must be used only in services with a relatively long CCM interval of at least 100 ms. CCM learning at MIPs is disabled by default.
Step 6	end or commit Example:	Saves configuration changes.

	Command or Action	Purpose
	<pre>RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# commit</pre>	<ul style="list-style-type: none"> When you use 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. 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 Cross-Check on a MEP for a CFM Service

To configure cross-check on a MEP for a CFM service and specify the expected set of MEPs, complete the following steps:

SUMMARY STEPS

- configure**
- ethernet cfm**
- domain** *domain-name level level-value* [**id** [**null**] [**dns** *DNS-name*] [**mac** *H.H.H*] [**string** *string*]]
- service** *service-name* {**bridge group** *bridge-domain-group bridge-domain bridge-domain-name* | **down-meps** | **xconnect group** *xconnect-group-name p2p xconnect-name*} [**id** [**icc-based** *icc-string umc-string*] | [**string** *text*] | [**number** *number*] | [**vlan-id** *id-number*] | [**vpn-id** *oui-vpnid*]]
- mep crosscheck**
- mep-id** *mep-id-number* [**mac-address** *mac-address*]
- end** or **commit**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	ethernet cfm Example: <pre>RP/0/RP0/CPU0:router# ethernet cfm</pre>	Enters the Ethernet Connectivity Fault Management (CFM) configuration mode.
Step 3	domain <i>domain-name</i> level <i>level-value</i> [id [null] [dns <i>DNS-name</i>] [mac <i>H.H.H</i>] [string <i>string</i>]] Example: <pre>RP/0/RP0/CPU0:router(config-cfm)# domain Domain_One level 1 id string D1</pre>	<p>Creates and names a container for all domain configurations and enters the CFM domain configuration mode.</p> <p>The level must be specified.</p> <p>The id is the maintenance domain identifier (MDID) and is used as the first part of the maintenance association identifier (MAID) in CFM frames. If the MDID is not specified, the domain name is used as the MDID by default.</p>
Step 4	service <i>service-name</i> { bridge group <i>bridge-domain-group</i> bridge-domain <i>bridge-domain-name</i> down-meps xconnect group <i>xconnect-group-name</i> p2p <i>xconnect-name</i> } [id [icc-based <i>icc-string umc-string</i>] [string <i>text</i>] [number <i>number</i>] [vlan-id <i>id-number</i>] [vpn-id <i>oui-vpnid</i>]] Example: <pre>RP/0/RP0/CPU0:router(config-cfm-dmn)# service Bridge_Service bridge group B1 bridge-domain B1</pre>	<p>Configures and associates a service with the domain and enters CFM domain service configuration mode. You can specify that the service is used only for down MEPs, or associate the service with a bridge domain or xconnect where MIPs and up MEPs will be created.</p> <p>The id sets the short MA name.</p>
Step 5	mep crosscheck Example: <pre>RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# mep crosscheck mep-id 10</pre>	Enters CFM MEP crosscheck configuration mode.
Step 6	mep-id <i>mep-id-number</i> [mac-address <i>mac-address</i>] Example: <pre>RP/0/RP0/CPU0:router(config-cfm-xcheck)# mep-id 10</pre>	<p>Enables cross-check on a MEP.</p> <p>Note</p> <ul style="list-style-type: none"> Repeat this command for every MEP that you want included in the expected set of MEPs for cross-check.
Step 7	end or commit Example: <pre>RP/0/RP0/CPU0:router(config-cfm-xcheck)# commit</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> When you use 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.

Configuring Other Options for a CFM Service

To configure other options for a CFM service, complete the following steps:

SUMMARY STEPS

1. **configure**
2. **ethernet cfm**
3. **domain** *domain-name* **level** *level-value* [**id** **null**] [**dns** *DNS-name*] [**mac** *H.H.H*] [**string** *string*]]
4. **service** *service-name* {**bridge group** *bridge-domain-group* **bridge-domain** *bridge-domain-name* | **down-meps** | **xconnect group** *xconnect-group-name* **p2p** *xconnect-name*} [**id** [**icc-based** *icc-string* *umc-string*] | [**string** *text*] | [**number** *number*] | [**vlan-id** *id-number*] | [**vpn-id** *oui-vpnid*]]
5. **maximum-meps** *number*
6. **log** {**ais**|**continuity-check errors**|**continuity-check mep changes**|**crosscheck errors**|**efd**}
7. **end** or **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	ethernet cfm Example: RP/0/RP0/CPU0:router# ethernet cfm	Enters the Ethernet Connectivity Fault Management (CFM) configuration mode.
Step 3	domain <i>domain-name</i> level <i>level-value</i> [id null] [dns <i>DNS-name</i>] [mac <i>H.H.H</i>] [string <i>string</i>]] Example: RP/0/RP0/CPU0:router(config-cfm)# domain Domain_One level 1 id string D1	Creates and names a container for all domain configurations and enters the CFM domain configuration mode. The level must be specified. The id is the maintenance domain identifier (MDID) and is used as the first part of the maintenance association

	Command or Action	Purpose
		identifier (MAID) in CFM frames. If the MDID is not specified, the domain name is used as the MDID by default.
Step 4	<p>service <i>service-name</i> {bridge group <i>bridge-domain-group</i> bridge-domain <i>bridge-domain-name</i> down-meps xconnect group <i>xconnect-group-name</i> p2p <i>xconnect-name</i>} [id [icc-based <i>icc-string umc-string</i>] [string text] [number number] [vlan-id id-number] [vpn-id oui-vpnid]]</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-cfm-dmn)# service Bridge_Service bridge group BD1 bridge-domain B1</pre>	<p>Configures and associates a service with the domain and enters CFM domain service configuration mode. You can specify that the service is used only for down MEPS, or associate the service with a bridge domain or xconnect where MIPs and up MEPS will be created.</p> <p>The id sets the short MA name.</p>
Step 5	<p>maximum-meps <i>number</i></p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# maximum-meps 1000</pre>	(Optional) Configures the maximum number (2 to 8190) of MEPS across the network, which limits the number of peer MEPS recorded in the database.
Step 6	<p>log {ais continuity-check errors continuity-check mep changes crosscheck errors efd}</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# log continuity-check errors</pre>	(Optional) Enables logging of certain types of events.
Step 7	<p>end or commit</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# commit</pre>	<p>Saves configuration changes.</p> <ul style="list-style-type: none"> • When you use 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. • 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 CFM MEPs

- For every subinterface configured under a Layer 3 parent interface, you must associate a unique 802.1Q or 802.1ad tag. Else, it leads to unknown network behavior.

SUMMARY STEPS

1. **configure**
2. **interface** {**HundredGigE** | **TenGigE**} *interface-path-id*
3. **interface** {**HundredGigE** | **TenGigE** | **Bundle-Ether**} *interface-path-id.subinterface*
4. **vrf vrf-name**
5. **interface** {**HundredGigE** | **TenGigE**} *interface-path-id*
6. **ethernet cfm**
7. **mep domain** *domain-name* **service** *service-name* **mep-id** *id-number*
8. **cos** *cos*
9. **end** or **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	interface { HundredGigE TenGigE } <i>interface-path-id</i> Example: RP/0/RP0/CPU0:router(config)# <code>interface TenGigE 0/0/0/1</code>	Type of Ethernet interface on which you want to create a MEP. Enter HundredGigE or TenGigE and the physical interface or virtual interface. Note <ul style="list-style-type: none"> • Use the show interfaces command to see a list of all interfaces currently configured on the router.
Step 3	interface { HundredGigE TenGigE Bundle-Ether } <i>interface-path-id.subinterface</i> Example: RP/0/RP0/CPU0:router(config)# <code>interface TenGigE 0/0/0/1</code>	Type of Ethernet interface on which you want to create a MEP. Enter HundredGigE , TenGigE , or Bundle-Ether and the physical interface or virtual interface followed by the subinterface path ID. Naming convention is <i>interface-path-id.subinterface</i> . The period in front of the subinterface value is required as part of the notation.
Step 4	vrf vrf-name Example: RP/0/RP0/CPU0:router(config-if)# <code>vrf vrf_A</code>	Configures a VRF instance and enters VRF configuration mode.

	Command or Action	Purpose
Step 5	interface { HundredGigE TenGigE } <i>interface-path-id</i> Example: RP/0/RP0/CPU0:router(config)# interface TenGigE 0/0/0/1	Type of Ethernet interface on which you want to create a MEP. Enter HundredGigE or TenGigE and the physical interface or virtual interface. Note <ul style="list-style-type: none"> Use the show interfaces command to see a list of all interfaces currently configured on the router.
Step 6	ethernet cfm Example: RP/0/RP0/CPU0:router(config-if)# ethernet cfm	Enters interface Ethernet CFM configuration mode.
Step 7	mep domain <i>domain-name</i> service <i>service-name</i> mep-id <i>id-number</i> Example: RP/0/RP0/CPU0:router(config-if-cfm)# mep domain Dm1 service Sv1 mep-id 1	Creates a maintenance end point (MEP) on an interface and enters interface CFM MEP configuration mode.
Step 8	cos <i>cos</i> Example: RP/0/RP0/CPU0:router(config-if-cfm-mep)# cos 7	(Optional) Configures the class of service (CoS) (from 0 to 7) for all CFM packets generated by the MEP on an interface. If not configured, the CoS is inherited from the Ethernet interface. Note For Ethernet interfaces, the CoS is carried as a field in the VLAN tag. Therefore, CoS only applies to interfaces where packets are sent with VLAN tags. If the cos (CFM) command is executed for a MEP on an interface that does not have a VLAN encapsulation configured, it will be ignored.
Step 9	end or commit Example: RP/0/RP0/CPU0:router(config-if-cfm-mep)# commit	Saves configuration changes. <ul style="list-style-type: none"> When you use 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. Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • 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 Y.1731 AIS

This section has the following step procedures:

Configuring AIS in a CFM Domain Service

Use the following procedure to configure Alarm Indication Signal (AIS) transmission for a CFM domain service and configure AIS logging.

SUMMARY STEPS

1. **configure**
2. **ethernet cfm**
3. **domain** *name level level*
4. **service name bridge group** *name bridge-domain name*
5. **service name xconnect group** *xconnect-group-name p2p xconnect-name*
6. **ais transmission** [**interval** {**1s**|**1m**}][**cos** *cos*]
7. **log ais**
8. **end** or **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	ethernet cfm Example: RP/0/RP0/CPU0:router(config)# <code>ethernet cfm</code>	Enters Ethernet CFM global configuration mode.
Step 3	domain <i>name level level</i> Example: RP/0/RP0/CPU0:router(config-cfm)# <code>domain D1 level 1</code>	Specifies the domain and domain level.

	Command or Action	Purpose
Step 4	<p>service name bridge group name bridge-domain name</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-cfm-dmn)# service S1 bridge group BG1 bridge-domain BD2</pre>	Specifies the service, bridge group, and bridge domain.
Step 5	<p>service name xconnect group xconnect-group-name p2p xconnect-name</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-cfm-dmn)# service S1 xconnect group XG1 p2p X2</pre>	Specifies the service and cross-connect group and name.
Step 6	<p>ais transmission [interval {1s 1m}][cos cos]</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# ais transmission interval 1m cos 7</pre>	Configures Alarm Indication Signal (AIS) transmission for a Connectivity Fault Management (CFM) domain service.
Step 7	<p>log ais</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# log ais</pre>	Configures AIS logging for a Connectivity Fault Management (CFM) domain service to indicate when AIS or LCK packets are received.
Step 8	<p>end or commit</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-sla-prof-stat-cfg)# 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> 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 AIS on a CFM Interface

To configure AIS on a CFM interface, perform the following steps:

SUMMARY STEPS

1. **configure**
2. **interface gigabitethernet *interface-path-id***
3. **ethernet cfm**
4. **ais transmission up interval 1m cos *cos***
5. **end** or **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	interface gigabitethernet <i>interface-path-id</i> Example: RP/0/RP0/CPU0:router# interface TenGigE 0/0/0/2	Enters interface configuration mode.
Step 3	ethernet cfm Example: RP/0/RP0/CPU0:router(config)# ethernet cfm	Enters Ethernet CFM interface configuration mode.
Step 4	ais transmission up interval 1m cos <i>cos</i> Example: RP/0/RP0/CPU0:router(config-if-cfm)# ais transmission up interval 1m cos 7	Configures Alarm Indication Signal (AIS) transmission on a Connectivity Fault Management (CFM) interface.
Step 5	end or commit Example: RP/0/RP0/CPU0:router(config-sla-prof-stat-cfg)# 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.

Configuring Flexible VLAN Tagging for CFM

Use this procedure to set the number of tags in CFM packets in a CFM domain service.

SUMMARY STEPS

1. **configure**
2. **ethernet cfm**
3. **domain name level level**
4. **service name bridge group name bridge-domain name**
5. **tags number**
6. **end** or **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	ethernet cfm Example: RP/0/RP0/CPU0:router(config)# ethernet cfm	Enters Ethernet CFM global configuration mode.
Step 3	domain name level level Example: RP/0/RP0/CPU0:router(config-cfm)# domain D1 level 1	Specifies the domain and domain level.
Step 4	service name bridge group name bridge-domain name Example:	Specifies the service, bridge group, and bridge domain.

	Command or Action	Purpose
	RP/0/RP0/CPU0:router(config-cfm-dmn)# service S2 bridge group BG1 bridge-domain BD2	
Step 5	<p>tags number</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# tags 1</pre>	Specifies the number of tags in CFM packets. Currently, the only valid value is 1.
Step 6	<p>end or commit</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# 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> 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 CFM Configuration

To verify the CFM configuration, use one or more of the following commands:

show ethernet cfm configuration-errors [domain domain-name] [interface interface-path-id]	Displays information about errors that are preventing configured CFM operations from becoming active, as well as any warnings that have occurred.
show ethernet cfm local maintenance-points domain name [service name] interface type interface-path-id [mep mip]	Displays a list of local maintenance points.



Note After you configure CFM, the error message, *cfmd[317]: %L2-CFM-5-CCM_ERROR_CCMS_MISSED : Some received CCMs have not been counted by the CCM error counters*, may display. This error message does not have any functional impact and does not require any action from you.

Troubleshooting Tips

To troubleshoot problems within the CFM network, perform these steps:

SUMMARY STEPS

1. To verify connectivity to a problematic MEP, use the **ping ethernet cfm** command as shown in this example:
2. If the results of the **ping ethernet cfm** command show a problem with connectivity to the peer MEP, use the **traceroute ethernet cfm** command to help further isolate the location of the problem as shown in the following example:

DETAILED STEPS

Step 1 To verify connectivity to a problematic MEP, use the **ping ethernet cfm** command as shown in this example:

```
RP/0/RP0/CPU0:router# ping ethernet cfm domain D1 service S1 mep-id 16 source
interface TenGigE 0/0/0/1
```

```
Type escape sequence to abort.
Sending 5 CFM Loopbacks, timeout is 2 seconds -
Domain foo (level 2), Service foo
Source: MEP ID 1, interface TenGigE0/0/0/1
Target: 0001.0002.0003 (MEP ID 16):
  Running (5s) ...
Success rate is 60.0 percent (3/5), round-trip min/avg/max = 1251/1349/1402 ms
Out-of-sequence: 0.0 percent (0/3)
Bad data: 0.0 percent (0/3)
Received packet rate: 1.4 pps
```

Step 2 If the results of the **ping ethernet cfm** command show a problem with connectivity to the peer MEP, use the **traceroute ethernet cfm** command to help further isolate the location of the problem as shown in the following example:

```
RP/0/RP0/CPU0:router# traceroute ethernet cfm domain D1 service S1 mep-id 16 source
interface TenGigE 0/0/0/2
```

```
Traceroutes in domain D1 (level 4), service S1
Source: MEP-ID 1, interface TenGigE0/0/0/2
=====
Traceroute at 2009-05-18 12:09:10 to 0001.0203.0402,
TTL 64, Trans ID 2:

Hop Hostname/Last          Ingress MAC/name          Egress MAC/Name          Relay
-----
 1 ios
   0000-0001.0203.0400    0001.0203.0400 [Down]
   TenGigE0/0/0/2
 2 abc
   ios                    0001.0203.0401 [Ok]
   Not present
 3 bcd
   abc                    0001.0203.0402 [Ok]
   TenGigE0/0
Replies dropped: 0
```

If the target was a MEP, verify that the last hop shows “Hit” in the Relay field to confirm connectivity to the peer MEP.

If the Relay field contains “MPDB” for any of the hops, then the target MAC address was not found in the bridge MAC learning table at that hop, and the result is relying on CCM learning. This result can occur under normal conditions, but

it can also indicate a problem. If you used the **ping ethernet cfm** command before using the **traceroute ethernet cfm** command, then the MAC address should have been learned. If “MPDB” is appearing in that case, then this indicates a problem at that point in the network.

Configuration Examples for Ethernet CFM

This section includes the following examples:

Ethernet CFM Domain Configuration: Example

This example shows how to configure a basic domain for Ethernet CFM:

```
configure
ethernet cfm
  traceroute cache hold-time 1 size 3000
  domain Domain_One level 1 id string D1
commit
```

Ethernet CFM Service Configuration: Example

This example shows how to create a service for an Ethernet CFM domain:

```
service Bridge_Service bridge group BD1 bridge-domain B1
service Cross_Connect_1 xconnect group XG1 p2p X1
commit
```

Flexible Tagging for an Ethernet CFM Service Configuration: Example

This example shows how to set the number of tags in CFM packets from down MEPs in a CFM domain service:

```
configure
ethernet cfm
  domain D1 level 1
    service S2 bridge group BG1 bridge-domain BD2
      tags 1
commit
```

Continuity Check for an Ethernet CFM Service Configuration: Example

This example shows how to configure continuity-check options for an Ethernet CFM service:

```
continuity-check archive hold-time 100
continuity-check loss auto-traceroute
continuity-check interval 100ms loss-threshold 10
commit
```

MIP Creation for an Ethernet CFM Service Configuration: Example

This example shows how to enable MIP auto-creation for an Ethernet CFM service:

```
RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# mip auto-create all
RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# commit
```

Cross-check for an Ethernet CFM Service Configuration: Example

This example shows how to configure cross-check for MEPs in an Ethernet CFM service:

```
mep crosscheck
mep-id 10
mep-id 20
commit
```

Other Ethernet CFM Service Parameter Configuration: Example

This example shows how to configure other Ethernet CFM service options:

```
maximum-meps 4000
log continuity-check errors
commit
exit
exit
exit
```

MEP Configuration: Example

This example shows how to configure a MEP for Ethernet CFM on an interface:

```
interface TenGigE 0/0/0/1
 ethernet cfm
 mep domain Dm1 service Sv1 mep-id 1
 commit
```

Ethernet CFM Show Command: Examples

These examples show how to verify the configuration of Ethernet Connectivity Fault Management (CFM):

Example 1

This example shows how to display all the maintenance points that have been created on an interface:

```
RP/0/RP0/CPU0:router# show ethernet cfm local maintenance-points
```

Domain/Level	Service	Interface	Type	ID	MAC
fig/5	bay	Gi0/10/0/12	Dn MEP	2	44:55:66
fig/5	bay	Gi0/0/1/0	MIP		55:66:77
fred/3	barney	Gi0/1/0/0	Dn MEP	5	66:77:88!

Example 2

This example shows how to display all the CFM configuration errors on all domains:

```
RP/0/RP0/CPU0:router# show ethernet cfm configuration-errors
```

```
Domain fig (level 5), Service bay
```


* MIP creation configured using bridge-domain blort, but bridge-domain blort does not exist.

* An Up MEP is configured for this domain on interface TenGigE0/0/0/3 and an Up MEP is also configured for domain blort, which is at the same level (5).

* A MEP is configured on interface TenGigE0/0/0/1 for this domain/service, which has CC interval 100ms, but the lowest interval supported on that interface is 1s

Example 3

This example shows how to display operational state for local maintenance end points (MEPs):

```
RP/0/RP0/CPU0:router# show ethernet cfm local meps
```

```
A - AIS received           I - Wrong interval
R - Remote Defect received V - Wrong Level
L - Loop (our MAC received) T - Timed out (archived)
C - Config (our ID received) M - Missing (cross-check)
X - Cross-connect (wrong MAID) U - Unexpected (cross-check)
P - Peer port down
```

```
Domain foo (level 6), Service bar
  ID Interface (State)      Dir MEPS/Err RD Defects AIS
-----
  100 Gi1/1/0/1 (Up)       Up    0/0  N  A    L7
```

```
Domain fred (level 5), Service barney
  ID Interface (State)      Dir MEPS/Err RD Defects AIS
-----
   2 Gi0/1/0/0 (Up)       Up    3/2  Y  RPC   L6
```

```
Domain foo (level 6), Service bar
  ID Interface (State)      Dir MEPS/Err RD Defects AIS
-----
  100 Gi1/1/0/1 (Up)       Up    0/0  N  A
```

```
Domain fred (level 5), Service barney
  ID Interface (State)      Dir MEPS/Err RD Defects AIS
-----
   2 Gi0/1/0/0 (Up)       Up    3/2  Y  RPC
```

Example 4

This example shows how to display operational state of other maintenance end points (MEPs) detected by a local MEP:

```
RP/0/RP0/CPU0:router# show ethernet cfm peer meps
```

```
Flags:
> - Ok           I - Wrong interval
R - Remote Defect received V - Wrong level
L - Loop (our MAC received) T - Timed out
C - Config (our ID received) M - Missing (cross-check)
X - Cross-connect (wrong MAID) U - Unexpected (cross-check)
```

```
Domain fred (level 7), Service barney
Down MEP on TenGigE0/0/0/1, MEP-ID 2
=====
St   ID MAC address   Port   Up/Downtime   CcmRcvd SeqErr   RDI Error
-----
>   1 0011.2233.4455 Up     00:00:01     1234    0    0    0
R>  4 4455.6677.8899 Up     1d 03:04     3456    0   234  0
L   2 1122.3344.5566 Up     3w 1d 6h     3254    0    0   3254
```

```

C      2 7788.9900.1122 Test    00:13          2345      6    20  2345
X      3 2233.4455.6677 Up     00:23          30        0     0   30
I      3 3344.5566.7788 Down   00:34      12345     0   300  1234
V      3 8899.0011.2233 Blocked 00:35        45        0     0   45
  T    5 5566.7788.9900         00:56        20        0     0   0
M      6                                0          0     0   0
U>    7 6677.8899.0011 Up     00:02        456        0     0   0

```

```

Domain fred (level 7), Service fig
Down MEP on TenGigE0/0/0/12, MEP-ID 3

```

```

=====
St   ID MAC address      Port    Up/DownTime  CcmRcvd SeqErr   RDI  Error
---
>   1 9900.1122.3344 Up      03:45      4321     0     0    0

```

Example 5

This example shows how to display operational state of other maintenance end points (MEPs) detected by a local MEP with details:

```

RP/0/RP0/CPU0:router# show ethernet cfm peer meps detail

```

```

Domain dom3 (level 5), Service ser3
Down MEP on TenGigE0/0/0/1 MEP-ID 1

```

```

=====
Peer MEP-ID 10, MAC 0001.0203.0403
  CFM state: Wrong level, for 00:01:34
  Port state: Up
  CCM defects detected:    V - Wrong Level
  CCMs received: 5
    Out-of-sequence:      0
    Remote Defect received: 5
    Wrong Level:          0
    Cross-connect (wrong MAID): 0
    Wrong Interval:      5
    Loop (our MAC received): 0
    Config (our ID received): 0
Last CCM received 00:00:06 ago:
  Level: 4, Version: 0, Interval: 1min
  Sequence number: 5, MEP-ID: 10
  MAID: String: dom3, String: ser3
  Port status: Up, Interface status: Up

```

```

Domain dom4 (level 2), Service ser4
Down MEP on TenGigE0/0/0/2 MEP-ID 1

```

```

=====
Peer MEP-ID 20, MAC 0001.0203.0402
  CFM state: Ok, for 00:00:04
  Port state: Up
  CCMs received: 7
    Out-of-sequence:      1
    Remote Defect received: 0
    Wrong Level:          0
    Cross-connect (wrong MAID): 0
    Wrong Interval:      0
    Loop (our MAC received): 0
  Config (our ID received): 0
Last CCM received 00:00:04 ago:
  Level: 2, Version: 0, Interval: 10s
  Sequence number: 1, MEP-ID: 20
  MAID: String: dom4, String: ser4
  Chassis ID: Local: ios; Management address: 'Not specified'
  Port status: Up, Interface status: Up

```

```

Peer MEP-ID 21, MAC 0001.0203.0403
CFM state: Ok, for 00:00:05
Port state: Up
CCMs received: 6
  Out-of-sequence:          0
  Remote Defect received:   0
  Wrong Level:              0
  Cross-connect (wrong MAID): 0
  Wrong Interval:          0
  Loop (our MAC received):  0
  Config (our ID received): 0
Last CCM received 00:00:05 ago:
Level: 2, Version: 0, Interval: 10s
Sequence number: 1, MEP-ID: 21
MAID: String: dom4, String: ser4
Port status: Up, Interface status: Up

Peer MEP-ID 601, MAC 0001.0203.0402
CFM state: Timed Out (Standby), for 00:15:14, RDI received
Port state: Down
CCM defects detected:   Defects below ignored on local standby MEP
                        I - Wrong Interval
                        R - Remote Defect received
                        T - Timed Out
                        P - Peer port down
CCMs received: 2
  Out-of-sequence:          0
  Remote Defect received:   2
  Wrong Level:              0

  Wrong Interval:          2
  Loop (our MAC received):  0
  Config (our ID received): 0
Last CCM received 00:15:49 ago:
Level: 2, Version: 0, Interval: 10s
Sequence number: 1, MEP-ID: 600
MAID: DNS-like: dom5, String: ser5
Chassis ID: Local: ios; Management address: 'Not specified'
Port status: Up, Interface status: Down

```

Ethernet CFM Show Command for flexible packet format: Examples

The flexible packet format supports the following two types of formats:

- MDID String format
- MDID Invalid format



Note To enable the feature in native mode, use the **hw-module profile oam 48byte-cfm-maid-enable** command in the System Admin Config mode. Ensure that you reload the router after configuring the native mode.

```

Router(config)#hw-module profile oam ?
  48byte-cfm-maid-enable  Enable 48byte cfm maid feature
  sat-enable              enable SAT feature
Router(config)#hw-module profile oam 48byte-cfm-maid-enable
In order to make the oam profile take effect, the router must be manually reloaded.
Router(config)#commit

```

```

Router(config)#hw-module profile npu native-mode-enable
Tue Nov 16 06:48:34.027 UTC
In order to activate this new npu profile, you must manually reload the chassis
Router(config)#commit

```

MDID String format: Example

Configuration

```

Router(config)#ethernet cfm
Router(config-cfm)#domain test level 3 id string test_domain
Router(config-cfm-dmn)#service test down-meps id string test_service
Router(config-cfm-dmn-svc)#mep crosscheck mep-id 4
Router(config-cfm-dmn-svc)#log continuity-check mep changes
Router(config-cfm-dmn-svc)#continuity-check interval 10ms
Router(config-cfm-dmn-svc)#commit
Router(config-cfm-dmn-svc)#root
Router(config)#interface TenGigE0/0/0/0.1 12tr

```

```

Router(config-subif)#encapsulation dot1q 1

```

```

Router(config-subif)#ethernet cfm
Router(config-if-cfm)#mep domain test service test mep-id 3
Router(config-if-cfm-mep)#commit

```

Verification

```

Router#show ethernet cfm peer meps

```

```

Tue Nov 16 06:46:13.859 UTC
Flags:
> - Ok                               I - Wrong interval
R - Remote Defect received           V - Wrong level
L - Loop (our MAC received)          T - Timed out
C - Config (our ID received)         M - Missing (cross-check)
X - Cross-connect (wrong MAID)       U - Unexpected (cross-check)
* - Multiple errors received         S - Standby

```

```

Domain test (level 3), Service test
Down MEP on TenGigE0/0/0/0.1 MEP-ID 3

```

```

=====
St   ID MAC Address      Port    Up/Downtime    CcmRcvd SeqErr  RDI Error
-----
>   4 d46a.355c.b814 Up      00:02:30      0      0      0      0

```

```

Router#show ethernet cfm peer meps detail

```

```

Tue Nov 16 06:46:29.169 UTC
Domain test (level 3), Service test
Down MEP on TenGigE0/0/0/0.1 MEP-ID 3
=====
Peer MEP-ID 4, MAC d46a.355c.b814
CFM state: Ok, for 00:02:46
Received CCM handling offloaded to hardware
Port state: Up
CCMs received: 0
Out-of-sequence: 0
Remote Defect received: 0
Wrong level: 0
Cross-connect (wrong MAID): 0

```

```

Wrong interval:          0
Loop (our MAC received): 0
Config (our ID received): 0
Last CCM received:
Level: 3, Version: 0, Interval: 10ms
Sequence number: 0, MEP-ID: 4
MAID: String: test_domain, String: test_service
Port status: Up, Interface status: Up

```

```
Router#show ethernet cfm local meps verbose
```

```
Tue Nov 16 06:46:41.783 UTC
Domain test (level 3), Service test
Down MEP on TenGigE0/0/0/0.1 MEP-ID 3
```

```
=====
Interface state: Up      MAC address: b0c5.3cff.c080
Peer MEPs: 1 up, 0 with errors, 0 timed out (archived)
Cross-check errors: 0 missing, 0 unexpected

```

```
CCM generation enabled: Yes, 10ms (Remote Defect detected: No)
                        CCM processing offloaded to hardware

```

```
AIS generation enabled: No
Sending AIS:             No
Receiving AIS:          No
Sending CSF:             No
Receiving CSF:          No

```

```
No packets sent/received
```

```
Router#
```

```
Router#show run interface tenGigE 0/0/0/0.1
```

```
Tue Nov 16 06:47:09.035 UTC
interface TenGigE0/0/0/0.1 l2transport
encapsulation dot1q 1
ethernet cfm
  mep domain test service test mep-id 3
  !
  !
  !
```

```
Router#show run ethernet cfm
```

```
Tue Nov 16 06:47:23.800 UTC
ethernet cfm
domain test level 3 id string test_domain
  service test down-meps id string test_service
  continuity-check interval 10ms
  mep crosscheck
  mep-id 4
  !
  log continuity-check mep changes
  !
  !
  !
-----
```

MDID Invalid format: Example

Configuration

```
Router#show run ethernet cfm
```

```
Tue Nov 16 06:57:14.099 UTC
```

```

ethernet cfm
domain test level 3
  service test down-meps
    continuity-check interval 10ms
    mep crosscheck
    mep-id 4
  !
  log continuity-check mep changes
!
!
!

```

Verification

```
Router#show ethernet cfm peer meps
```

```
Tue Nov 16 06:57:19.027 UTC
```

```
Flags:
```

```

> - Ok                      I - Wrong interval
R - Remote Defect received  V - Wrong level
L - Loop (our MAC received) T - Timed out
C - Config (our ID received) M - Missing (cross-check)
X - Cross-connect (wrong MAID) U - Unexpected (cross-check)
* - Multiple errors received S - Standby

```

```
Domain test (level 3), Service test
Down MEP on TenGigE0/0/0/0.1 MEP-ID 3
```

```

=====
St   ID MAC Address      Port   Up/Downtime   CcmRcvd SeqErr   RDI Error
---
>   4 d46a.355c.b814 Up     00:00:24     0       0       0       0

```

```
Router#show ethernet cfm peer meps detail
```

```
Tue Nov 16 06:57:23.567 UTC
```

```
Domain test (level 3), Service test
Down MEP on TenGigE0/0/0/0.1 MEP-ID 3
```

```

=====
Peer MEP-ID 4, MAC d46a.355c.b814
CFM state: Ok, for 00:00:29
Received CCM handling offloaded to hardware
Port state: Up
CCMs received: 0
  Out-of-sequence:          0
  Remote Defect received:   0
  Wrong level:              0
  Cross-connect (wrong MAID): 0
  Wrong interval:          0
  Loop (our MAC received):  0
  Config (our ID received): 0
Last CCM received:
  Level: 3, Version: 0, Interval: 10ms
  Sequence number: 0, MEP-ID: 4
  MAID: String: test, String: test
  Port status: Up, Interface status: Up

```

```
Router#show ethernet cfm local meps
```

```
Tue Nov 16 06:57:36.672 UTC
```

```
Defects (from at least one peer MEP):
```

```

A - AIS received           I - Wrong interval
R - Remote Defect received V - Wrong level
L - Loop (our MAC received) T - Timed out
C - Config (our ID received) M - Missing (cross-check)
X - Cross-connect (wrong MAID) U - Unexpected (cross-check)
P - Peer port down        F - CSF received

```

```

Domain test (level 3), Service test
  ID Interface (State)          Dir MEPS/Err RD Defects AIS
-----
   3 Te0/0/0/0.1 (Up)          Dn      1/0   N

Router#show ethernet cfm local meps verbose
Tue Nov 16 06:57:39.015 UTC
Domain test (level 3), Service test
Down MEP on TenGigE0/0/0/0.1 MEP-ID 3
=====
Interface state: Up          MAC address: b0c5.3cff.c080
Peer MEPS: 1 up, 0 with errors, 0 timed out (archived)
Cross-check errors: 0 missing, 0 unexpected

CCM generation enabled: Yes, 10ms (Remote Defect detected: No)
                        CCM processing offloaded to hardware
AIS generation enabled: No
Sending AIS:            No
Receiving AIS:         No
Sending CSF:           No
Receiving CSF:        No

No packets sent/received
Router#

```

AIS for CFM Configuration: Examples

Example 1

This example shows how to configure Alarm Indication Signal (AIS) transmission for a CFM domain service:

```

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ethernet cfm
RP/0/RP0/CPU0:router(config-cfm)# domain D1 level 1
RP/0/RP0/CPU0:router(config-cfm-dmn)# service S1 bridge group BG1 bridge-domain BD2
RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# ais transmission interval 1m cos 7

RP/0/RP0/CPU0:routerconfigure
RP/0/RP0/CPU0:router(config)# ethernet cfm
RP/0/RP0/CPU0:router(config-cfm)# domain D1 level 1
RP/0/RP0/CPU0:router(config-cfm-dmn)# service Cross_Connect_1 xconnect group XG1 p2p X1
RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# ais transmission interval 1m cos 7

```

Example 2

This example shows how to configure AIS logging for a Connectivity Fault Management (CFM) domain service to indicate when AIS or LCK packets are received:

```

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ethernet cfm
RP/0/RP0/CPU0:router(config-cfm)# domain D1 level 1
RP/0/RP0/CPU0:router(config-cfm-dmn)# service S2 bridge group BG1 bridge-domain BD2
RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# log ais

RP/0/RP0/CPU0:routerconfigure
RP/0/RP0/CPU0:router(config)# ethernet cfm
RP/0/RP0/CPU0:router(config-cfm)# domain D1 level 1

```

```
RP/0/RP0/CPU0:router(config-cfm-dmn)# service Cross_Connect_1 xconnect group XG1 p2p X1
RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# log ais
```

This example shows how to configure AIS transmission on a CFM interface.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface TenGigE 0/1/0/2
RP/0/RP0/CPU0:router(config-if)# ethernet cfm
RP/0/RP0/CPU0:router(config-if-cfm)# ais transmission up interval 1m cos 7
```

AIS for CFM Show Commands: Examples

This section includes the following examples:

show ethernet cfm interfaces ais Command: Example

This example shows how to display the information published in the Interface AIS table:

```
RP/0/RP0/CPU0:router# show ethernet cfm interfaces ais

Defects (from at least one peer MEP):
A - AIS received           I - Wrong interval
R - Remote Defect received V - Wrong Level
L - Loop (our MAC received) T - Timed out (archived)
C - Config (our ID received) M - Missing (cross-check)
X - Cross-connect (wrong MAID) U - Unexpected (cross-check)
P - Peer port down        D - Local port down
```

Interface (State)	AIS Dir	Trigger		Via Levels	Transmission			
		L	Defects		L	Int	Last started	Packets
TenGigE0/0/0/0 (Up)	Dn	5	RPC	6	7	1s	01:32:56 ago	5576
TenGigE0/0/0/0 (Up)	Up	0	M	2,3	5	1s	00:16:23 ago	983
TenGigE0/0/0/1 (Dn)	Up		D		7	60s	01:02:44 ago	3764
TenGigE0/0/0/2 (Up)	Dn	0	RX	1!				

show ethernet cfm local meps Command: Examples

Example 1: Default

This example shows how to display statistics for local maintenance end points (MEPs):

```
RP/0/RP0/CPU0:router# show ethernet cfm local meps

A - AIS received           I - Wrong interval
R - Remote Defect received V - Wrong Level
L - Loop (our MAC received) T - Timed out (archived)
C - Config (our ID received) M - Missing (cross-check)
X - Cross-connect (wrong MAID) U - Unexpected (cross-check)
P - Peer port down
```

```
Domain foo (level 6), Service bar
  ID Interface (State)      Dir MEPs/Err RD Defects AIS
-----
  100 Gi1/1/0/1 (Up)       Up    0/0  N A    7

Domain fred (level 5), Service barney
  ID Interface (State)      Dir MEPs/Err RD Defects AIS
```



```
-----
2 Gi0/1/0/0 (Up)      Up      3/2   Y  RPC      6
```

Example 2: Domain Service

This example shows how to display statistics for MEPs in a domain service:

```
RP/0/RP0/CPU0:router# show ethernet cfm local meps domain foo service bar detail
```

```
Domain foo (level 6), Service bar
Down MEP on TenGigE0/0/0/1, MEP-ID 100
=====
Interface state: Up      MAC address: 1122.3344.5566
Peer MEPs: 0 up, 0 with errors, 0 timed out (archived)

CCM generation enabled: No
AIS generation enabled: Yes (level: 7, interval: 1s)
Sending AIS:           Yes (started 01:32:56 ago)
Receiving AIS:         Yes (from lower MEP, started 01:32:56 ago)

Domain fred (level 5), Service barney
Down MEP on TenGigE0/0/0/1, MEP-ID 2
=====
Interface state: Up      MAC address: 1122.3344.5566
Peer MEPs: 3 up, 2 with errors, 0 timed out (archived)
Cross-check defects: 0 missing, 0 unexpected

CCM generation enabled: Yes (Remote Defect detected: Yes)
CCM defects detected:  R - Remote Defect received
                       P - Peer port down
                       C - Config (our ID received)
AIS generation enabled: Yes (level: 6, interval: 1s)
Sending AIS:           Yes (to higher MEP, started 01:32:56 ago)
Receiving AIS:         No
```

Example 4: Detail

This example shows how to display detailed statistics for MEPs in a domain service:

```
RP/0/RP0/CPU0:router# show ethernet cfm local meps detail

Domain foo (level 6), Service bar
Down MEP on TenGigE0/0/0/1, MEP-ID 100
=====
Interface state: Up      MAC address: 1122.3344.5566
Peer MEPs: 0 up, 0 with errors, 0 timed out (archived)

CCM generation enabled: No
AIS generation enabled: Yes (level: 7, interval: 1s)
Sending AIS:           Yes (started 01:32:56 ago)
Receiving AIS:         Yes (from lower MEP, started 01:32:56 ago)

Domain fred (level 5), Service barney
Down MEP on TenGigE0/0/0/1, MEP-ID 2
=====
Interface state: Up      MAC address: 1122.3344.5566
Peer MEPs: 3 up, 2 with errors, 0 timed out (archived)
Cross-check defects: 0 missing, 0 unexpected

CCM generation enabled: Yes (Remote Defect detected: Yes)
CCM defects detected:  R - Remote Defect received
                       P - Peer port down
```

show ethernet cfm local meps detail Command: Example

```

C - Config (our ID received)
AIS generation enabled: Yes (level: 6, interval: 1s)
Sending AIS:           Yes (to higher MEP, started 01:32:56 ago)
Receiving AIS:         No

```

show ethernet cfm local meps detail Command: Example

Use the **show ethernet cfm local meps detail** command to display MEP-related EFD status information. This example shows that EFD is triggered for MEP-ID 100:

```

RP/0/RP0/CPU0:router# show ethernet cfm local meps detail

Domain foo (level 6), Service bar
Down MEP on TenGigE0/0/0/1, MEP-ID 100
=====
Interface state: Up      MAC address: 1122.3344.5566
Peer MEPs: 0 up, 0 with errors, 0 timed out (archived)
Cross-check errors: 2 missing, 0 unexpected

CCM generation enabled: No
AIS generation enabled: Yes (level: 7, interval: 1s)
Sending AIS:           Yes (started 01:32:56 ago)
Receiving AIS:         Yes (from lower MEP, started 01:32:56 ago)
EFD triggered:         Yes

Domain fred (level 5), Service barney
Down MEP on TenGigE0/0/0/1, MEP-ID 2
=====
Interface state: Up      MAC address: 1122.3344.5566
Peer MEPs: 3 up, 0 with errors, 0 timed out (archived)
Cross-check errors: 0 missing, 0 unexpected

CCM generation enabled: Yes (Remote Defect detected: No)
AIS generation enabled: Yes (level: 6, interval: 1s)
Sending AIS:           No
Receiving AIS:         No
EFD triggered:         No

```

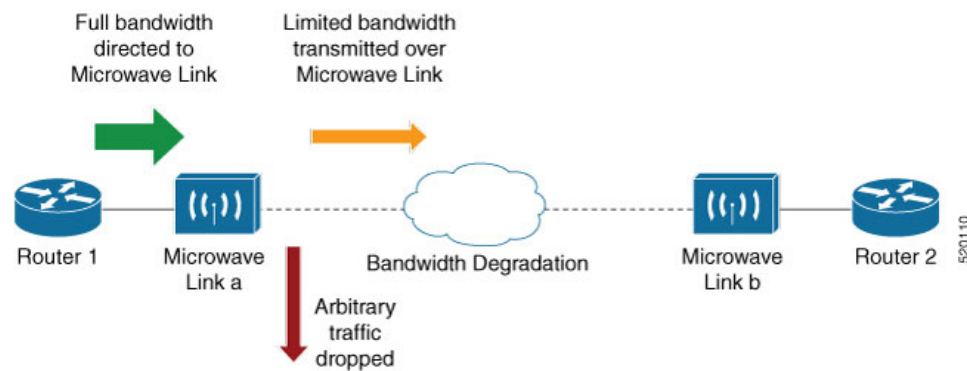


Note You can also verify that EFD has been triggered on an interface using the **show interfaces** and **show interfaces brief** commands. When an EFD trigger has occurred, these commands will show the interface status as *up* and the line protocol state as *down*.

CFM Adaptive Bandwidth Notifications

Microwave links are used in carrier ethernet networks, because they are cheaper than laying fibre either in dense metro areas or rural locations. However, the disadvantage of microwave links is that the signal is affected by atmospheric conditions and can degrade.

Modern microwave devices support adaptive modulation schemes to prevent a complete loss of signal. This allows them to continue to operate during periods of degradation, but at a reduced bandwidth. However, to fully take advantage of this, it's necessary to convey the decrease in bandwidth to the head-end router so that appropriate actions can be taken. Otherwise, the link may become saturated and traffic dropped arbitrarily as shown in the following figure:



A generic solution to this is a Connectivity Fault Management (CFM) extension to send Bandwidth Notifications Messages (BNM) to Maintenance Endpoints (MEPs) on the corresponding interface on the head-end router. To be flexible in the actions taken, the choice of solution uses Embedded Event Manager (EEM) to invoke operator written TCL scripts. For information on EEM, see [Embedded Event Manager, on page 127](#).

Bandwidth Notification Messages

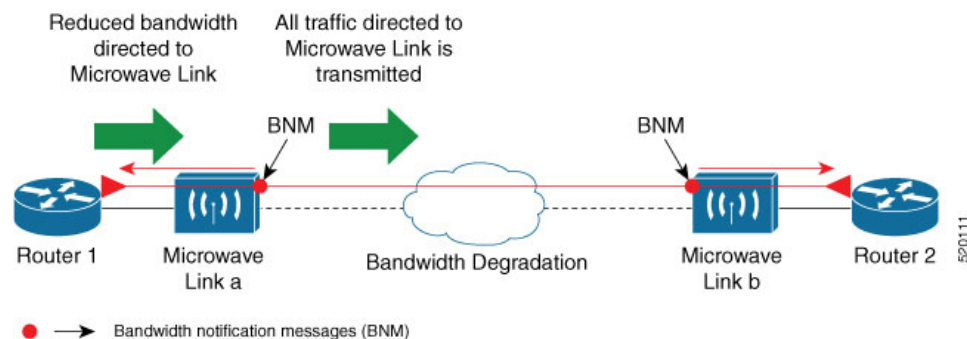
The two types of messages used to notify the head-end router are:

- G.8013 Bandwidth Notification Messages (G.8013 BNM)
- Cisco proprietary Bandwidth Vendor-Specific Messages (Cisco BW-VSM)

Both the message types contain the following information:

- Source MAC
- Port ID
- Maintenance Domain (MD) Level
- Transmission period
- Nominal Bandwidth
- Current Bandwidth

During signal degradation, periodic BNMs are sent to the head-end router containing the current bandwidth (sampled over a period of time) and nominal bandwidth (full bandwidth when there is no degradation). This allows the router to reduce the bandwidth directed to the link as shown in the figure below:



When degradation in bandwidth is detected, depending on the topology, the degradation may affect one or more paths in the network. Therefore, in more complex topologies, the head-end router may need information about links in each affected path. The BNM transmission period and a Link ID are used to differentiate between messages from the same source MAC address which refer to different links.

Restrictions for CFM Bandwidth Notifications

The list of restrictions for CFM Bandwidth Notifications is:

- Up to 200 unique BNM enabled links learnt from BNMs are supported per line card. Any BNMs for links over this limit will be discarded.

To reset CFM BNM enabled links for the specified interfaces, use the `clear ethernet cfm interface [<interface>] bandwidth-notifications { all | state <state> } [location { all | <node> }]` command. An archive timer is used to clean up any BNM enabled links whose loss timer expired at least 24 hours ago.

- Over process restart:
 - Loss threshold, wait-to-restore, and hold-off timers are restarted. This may cause links to take longer to transition between states than they would have otherwise.
 - Archive timers are restarted. This may cause historical statistics for links to persist longer than they would have otherwise.
 - Queued events for EEM scripts which have been rate-limited are not preserved. Scripts with at least one link in DEGRADED state, or BNMs have changed over process restart, and are invoked. Rate-limit timers are restarted. This may cause scripts to be invoked when they would otherwise have been filtered by the damping or conformance-testing algorithms. If the last link returns to its nominal bandwidth within the rate-limit period but before the process restart, then the script will not be invoked after the process restart. Thus, actions taken by the script may not reflect the (increased) latest bandwidths of any links which returned to their nominal bandwidths within the rate-limit period.

Bandwidth Reporting

Received BNMs are used to identify BNM enabled links within a Maintenance Entity Group (MEG), and should be uniquely identifiable within the MEG by Port-ID or MAC address. Each link has an associated nominal bandwidth, and a Reported Bandwidth (RBW), which are notified to the operator. The link is considered to be in OK state when the RBW is equal to the nominal bandwidth and DEGRADED if RBW is less than nominal.

Devices sending BNMs can detect changes in bandwidth many times a second. For example, changes caused by an object passing through a microwave link's line of sight. The protocol for sending BNMs is designed to mitigate fluctuating current bandwidth by sampling across a 'monitoring-interval' and applying basic damping to degradation events. To help mitigate this further, a damping algorithm is used. This algorithm is applied on the receiving device, and is distinct from any damping performed by the sender. For more information on this, see [Damping Algorithm, on page 125](#).

An operator may be interested in more than one BNM enabled link, and needs the ability to register on a set of BNM enabled links which affect the path to a node in the network. To do this, the state and RBW for each link of interest are put into a conformance testing algorithm, which both filters and rate-limits changes to

publish events notifying the operator only of significant changes. For more information on this, see [Conformance Testing Algorithm, on page 126](#).

The following diagram shows how a received BNM flows through the damping and conformance testing algorithm to invoke operator scripts:



Note

- Port ID takes precedence over MAC address. This means that BNM with same port ID but different MAC addresses are counted as same BNMs.
- If BNM reported bandwidth is equal to the threshold, then EEM will not be invoked.
- If a degraded link having bandwidth higher than the threshold receives BNM with bandwidth less than the threshold, it doesn't wait for the hold-off timer and instantly changes the bandwidth by invoking EEM script.

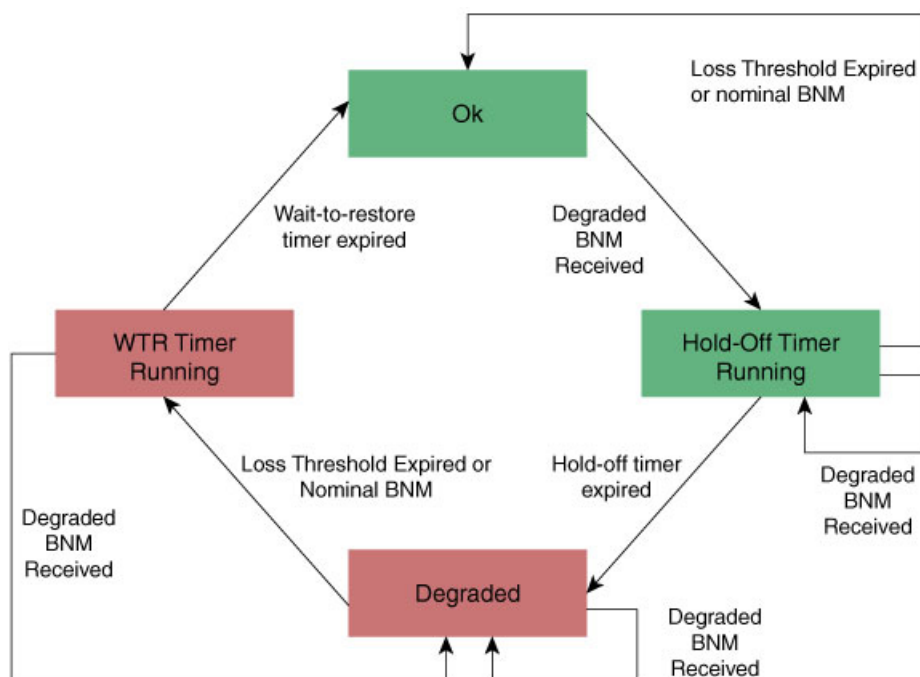
Damping Algorithm

A damping algorithm is applied to each unique BNM enabled link for which BNMs are received. The table below describes the timers used for this purpose:

Timers	Description
loss threshold (in packet numbers)	This timer handles the case when BNMs stop being received. This timer is (re)started whenever any BNM is received for the link. The value is equal to the expected period between BNMs (as indicated by the last BNM) multiplied by the configured loss threshold. When the loss threshold timer expires, as the link may have changed or been removed entirely, bandwidth information is no longer available, therefore the link is considered to have been restored to its previously notified nominal bandwidth.
hold-off (in seconds)	This timer is used to damp transient transitions from OK to DEGRADED state. It is started when the first degraded BNM is received, and is stopped if the loss threshold timer expires or the current bandwidth returns to the nominal bandwidth. If the timer expires, then the BNM enabled link enters DEGRADED state. The value of this timer is configurable. If it is zero, then the link immediately enters degraded state and the timer is not started.

Timers	Description
wait-to-restore (WTR, in seconds)	This timer is used to damp transient transitions from DEGRADED to OK state. It is started when a BNM Enabled Link is in DEGRADED state and either the loss threshold timer expires or a BNM is received that indicates the current bandwidth has returned to the nominal bandwidth. If a degraded BNM is received while the timer is running then it is stopped and the BNM Enabled Link remains in DEGRADED state. If this timer expires then the link returns to OK state.

The following internal state transition diagram shows how damping algorithm takes place:



520113

Conformance Testing Algorithm

The conformance testing algorithm comprises of two parts:

1. Filtering bandwidth changes.

Filtering is done so that events are published whenever either:

- Any link which was in OK state or had a RBW more than or equal to the specified threshold, has transitioned to DEGRADED state and has a RBW less than the specified threshold.
- Any link which was in DEGRADED state and had a RBW less than the specified threshold, is still in DEGRADED state and has a RBW less than the specified threshold, but the old and new RBWs are different.
- Any link which was in DEGRADED state and had a RBW less than the specified threshold, has transitioned to OK state or has a RBW more than or equal to the specified threshold.

2. Rate-limiting bandwidth changes

Rate-limiting is done by only publishing events at most once within any rate-limit period. If there is a change in bandwidth (which passes the filter) within this rate-limit period, a timer is started to expire at the end of the period. Upon timer expiry, an event is published which reflects the latest state and bandwidth of all links of interest which are in DEGRADED state.

Embedded Event Manager

The Embedded Event Manager (EEM) consists of an EEM server that monitors various real-time events in the system using programs called Event Detectors (EDs) and triggers registered policies (for example, TCLscripts) to run. The EEM supports at least 200 script registrations.

Typical actions taken in response to signal degradation events include:

- Signaling to G.8032 to switch some flows to alternative paths
- Modifying QoS configuration to adjust traffic shaping to the new bandwidth
- Adjusting IGP metrics to switch some traffic to an alternative path

The following variables can be queried within the TCL script:

EEM Variables	Comment
<code>interface, level, direction</code>	Identify the MEP in the registration
<code>min_reported_bandwidth</code>	Minimum reported bandwidth across all links in the registration that are currently in DEGRADED state, and below the specified threshold
<code>bnm_enabled_links [{ MAC address Port ID }]</code>	Array of BNM enabled links, with each one containing the following elements: <ul style="list-style-type: none"> • <code>reported_bw</code>: Reported Bandwidth • <code>nominal_bw</code>: Nominal BW in last BNM
<code>event_type</code>	Either 'DEGRADED' or 'OK' DEGRADED indicates that at least one BNM enabled link in the registration is in DEGRADED state with a reported bandwidth less than the threshold. This means that the <code>event_type</code> could be 'OK' if all BNM enabled links in the registration which are in DEGRADED state have a reported bandwidth greater than or equal to the threshold.

The command for EEM TCL scripts registering for CFM Bandwidth Notification events is `interface <interface name> level <level> direction <direction> {mac-addresses { <addr1> [, ..., <addr20>] } | port-ids { <id1> [, ..., <id20>] } threshold <bandwidth> [ratelimit <time>]`.

To configure EEM, use the following commands:

```
event manager directory user policy disk0:/
event manager directory user library disk0:/
event manager policy EEMscript7.tcl username root persist-time 3600
aaa authorization eventmanager default local
```

Individual scripts located in the specified directory can then be configured with:

```
event manager policy <script_name> username lab persist-time <time>
```

Event Publishing

CFM publishes events for a given EEM registration after applying the damping and conformance testing algorithms as described in [Damping Algorithm, on page 125](#) and [Conformance Testing Algorithm, on page 126](#) respectively. The set of BNM Enabled Links published in an event are those in DEGRADED state and whose RBW is less than the specified threshold.

Configure CFM Bandwidth Notifications

Use the following steps to configure CFM bandwidth notifications:

- Configure a CFM domain at the level BNMs are expected to be received at, and a CFM service in the direction (either up or down-MEPs) the BNMs are expected to be received.
- Configure a CFM MEP on the interface expected to receive BNMs in the domain and service above.

Configuration consists of two parts:

- Configuring global CFM. This is similar to Continuity Check Message (CCM) and other CFM configurations.

Global CFM configuration:

```
ethernet cfm
domain DM1 level 2 id null
  service SR1 down-meps
  !
!
domain dom1 level 1
  service ser1 down-meps
  !
!
```

- Configuration related to CFM-BNMs under interfaces. This is optional and used for changing default values.

Interface configuration:

```
Interface TenGigE0/0/1/1
ethernet cfm
  mep domain DM1 service SR1 mep-id 3001
  !
  bandwidth-notifications
  hold-off 0
  wait-to-restore 60
  loss-threshold 10
  log changes
  !
!
l2transport
!
!
```



```

interface TenGigE0/0/0/3
 ethernet cfm
   mep domain dom1 service ser1 mep-id 11
   !
   bandwidth-notifications
   hold-off 10
   wait-to-restore 40
   log changes
   !
!
!
l2transport
!
!
!

```

Running Configuration

```

RP/0/RP0/CPU0:router#show running-configuration
!! IOS XR Configuration 7.1.1.104I
!! Last configuration change at Mon Jun 24 21:26:46 2019 by root
!
hostname R2_cXR
logging console debugging
logging buffered 125000000
event manager directory user policy harddisk:/tcl/
event manager directory user library harddisk:/tcl/
event manager policy EEMmac_level1.tcl username root persist-time 3600
event manager policy EEMport_level1.tcl username root persist-time 3600
aaa authorization exec default local group tacacs+
aaa authorization eventmanager default local
!
 ethernet cfm
  domain DM0 level 1 id null
  service SR0 down-meps
  continuity-check interval 1m
  mep crosscheck
  mep-id 1003
  !
  ais transmission interval 1s cos 4
  log ais
  log continuity-check errors
  log crosscheck errors
  log continuity-check mep changes
  !
!
  domain DM1 level 2 id null
  service SR1 down-meps id number 1
  continuity-check interval 1m
  mep crosscheck
  mep-id 431
  !
  ais transmission interval 1m
  log ais
  log continuity-check errors
  log crosscheck errors
  log continuity-check mep changes
  !
!
  domain dom1 level 3 id string domain3
  service ser1 xconnect group XG1 p2p XC1 id number 2300
  mip auto-create all
  continuity-check interval 1m
  mep crosscheck
  mep-id 2030
  !
interface Loopback0
 ipv4 address 30.30.30.30 255.255.255.255

```

```

!
interface MgmtEth0/RSP0/CPU0/0
  ipv4 address 5.18.9.102 255.255.0.0
!
interface MgmtEth0/RSP0/CPU0/1
  shutdown
!
interface TenGigE0/0/0/0
  shutdown
!
interface TenGigE0/0/0/3.1 12transport
  encapsulation dot1q 6
  ethernet cfm
  mep domain DM1 service SR1 mep-id 231
!
  bandwidth-notifications
  hold-off 50
  wait-to-restore 50
  loss-threshold 100
  log changes
!

```

Verification

```

RP/0/RP0/CPU0:router#show ethernet cfm interfaces bandwidth-notifications detail
BNM Enabled Links at Level 3 (Down MEP) for GigabitEthernet/1
  MAC Address 000a.000a.000a
    State (OK):
      Nominal Bandwidth:                3000 Mbps
      Reported Bandwidth:                1000 Mbps
      Elapsed time in this state:        00:00:13.000
      Transitions into degraded state:   5000
      Hold-off:                          111s remaining
    Last BNM received 00:00:10 ago
      Nominal Bandwidth:                1000 Mbps
      Current Bandwidth:                2000 Mbps
      Interval:                          10s
      Packet-type:                       Cisco BW-VSM
      Packets received:                  20000

  Port ID 7 (MAC Address 000c.000c.000c)
    State (DEGRADED):
      Nominal Bandwidth:                6000 Mbps
      Reported Bandwidth:                2000 Mbps
      Elapsed time in this state:        00:00:39.000
      Transitions into degraded state:   10000
      Wait-to-restore:                  111s remaining
    Last BNM received 00:00:33 ago
      Nominal Bandwidth:                2000 Mbps
      Current Bandwidth:                4000 Mbps
      Interval:                          1min
      Packet-type:                       Cisco BW-VSM
      Packets received:                  40000

```

CFM Over Bundles

CFM over bundle supports the following:

- CFM Maintenance Points—Up Maintenance-association End Points (MEP), Down MEP, and MIP, which includes L2 bundle main and sub-interfaces.

- CCM interval of 100 microsecond, 1second, 10 seconds, and 1 minute. CCM interval of 10 minutes is supported only in the versions earlier than IOS XR 7.3.2.
- RP OIR/VM reload, without impacting learned CFM peer MEPs.
- Process restart without impacting CFM sessions.
- CFM MEPs on bundle interfaces as software-offloaded-MEPs with all possible rewrite and encapsulation combinations supported by L2 sub-interfaces.
- CCM learning on MIP over bundle interfaces. CCM database learning supports investigation of one CCM out of 50 that goes over MIP.
- Static and dynamic Remote MEPs.

Restrictions for Configuration of CFM on Bundles

Following are the restrictions for configuring CFM:

- Only Layer 2 bundle Ethernet interfaces and sub-interfaces are supported except for those matching the VLAN tag `any`.
- CCM interval of 3.3 milliseconds and 10 milliseconds are not supported.
- CCM interval of 10 minutes is not supported from IOS XR 7.3.2.
- Supports 5000 pps rates of CCM traffic for bundle interfaces. For example, for CCM interval of 100 milliseconds, the number of MEPs can be 500.
- Ethernet CFM is not supported with MEP that are configured on default and untagged encapsulated sub-interfaces that are part of a single physical interface.

Y.1731 Performance Monitoring

Table 10: Feature History Table

Feature Name	Release	Description
Cisco NC57 Native Mode: Y.1731 Loss and Delay Measurement	Release 7.3.1	This feature is now supported on routers that have the Cisco NC57 line cards installed and operate in the native mode.

Y.1731 Performance Monitoring (PM) provides a standard Ethernet PM function that includes measurement of Ethernet frame delay, frame delay variation, frame loss, and frame throughput measurements. This is specified by the ITU-T Y-1731 standard and interpreted by the Metro Ethernet Forum (MEF) standards group.

The router supports the following:

- Delay Measurement (DM)
- Synthetic Loss Measurement (SLM)

Two-Way Delay Measurement for Scalability

Use the Ethernet frame delay measurement to measure frame delay and frame delay variations. The system measures the Ethernet frame delay by using the Delay Measurement Message (DMM) method.

Restrictions for Configuring Two-Way Delay Measurement

Follow the guidelines and restrictions listed here when you configure two-way delay measurement:

- Y.1731 PM does not support One-Way DMM since PTP support is not available in the Release 6.3.1 for NCS 5500.
- System supports software-based timestamping for Two-Way DMM for NCS5502 and NCS5508 routers. The restriction is only applicable to UP MEP (Maintenance association End Point), which requires core NPU (Network Processor) and access NPU to have ToD (Time of Day) in sync to support 64-bit hardware-based timestamping. After you enable PTP (Precision Time Protocol) and sync all NPUs, the restriction is removed.

Configuring Two-Way Delay Measurement

Perform the following steps to configure two-way delay measurement:

```
RP/0/RP0/CPU0:router (config) # ethernet sla
profile DMM type cfm-delay-measurement
  probe
    send burst every 5 seconds packet count 5 interval 1 seconds
  !
  schedule
    every 1 minutes for 40 seconds
  !
  statistics
    measure round-trip-delay
      buckets size 1 probes
      buckets archive 5
    !
    measure round-trip-jitter
      buckets size 1 probes
      buckets archive 1
    !
  !
  !
  !
interface TenGigE0/0/0/10.1 l2transport
encapsulation dot1q 1
ethernet cfm
  mep domain DOWN0 service s10 mep-id 2001
  sla operation profile DMM target mep-id 6001
!
```

Configuring an On-Demand Ethernet SLA Operation for CFM Delay Measurement

To configure an on-demand Ethernet SLA operation for CFM delay measurement, use this command in privileged EXEC configuration mode:

```
RP/0/RP0/CPU0:router (config) #
ethernet sla on-demand operation type cfm-synthetic-loss-measurement probe domain D1 source
interface TenGigE 0/6/1/0 target mac-address 2.3.4
```

Running Configuration

```
RP/0/RP0/CPU0:router# show ethernet cfm peer meps
```

```
Mon Sep 11 12:09:44.534 UTC
```

```
Flags:
```

```
> - Ok
R - Remote Defect received
L - Loop (our MAC received)
C - Config (our ID received)
X - Cross-connect (wrong MAID)
* - Multiple errors received

I - Wrong interval
V - Wrong level
T - Timed out
M - Missing (cross-check)
U - Unexpected (cross-check)
S - Standby
```

```
Domain UP6 (level 6), Service s6
```

```
Up MEP on FortyGigE0/0/1/2.1 MEP-ID 1
```

```
=====
St   ID MAC Address   Port   Up/Downtime   CcmRcvd SeqErr   RDI Error
-----
> 4001 70e4.227c.2865 Up     00:01:27           0     0     0     0
```

```
Domain DOWN0 (level 0), Service s10
```

```
Down MEP on TenGigE0/0/0/10.1 MEP-ID 2001
```

```
=====
St   ID MAC Address   Port   Up/Downtime   CcmRcvd SeqErr   RDI Error
-----
> 6001 70e4.227c.287a Up     00:02:11           0     0     0     0
```

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

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

```
Mon Sep 11 12:10:18.467 UTC
```

```
interface FortyGigE0/0/1/2.1 l2transport
 encapsulation dot1q 1
 ethernet cfm
  mep domain UP6 service s6 mep-id 1
  sla operation profile DMM target mep-id 6001
  sla operation profile test-slm target mep-id 6001
!
!
!
l2vpn
xconnect group g1
 p2p p1
  interface TenGigE0/0/0/10.1
  interface FortyGigE0/0/1/2.1
!
!
end
```

Verification

```
One-way Delay (Source->Dest)
```

```
~~~~~
```

```
1 probes per bucket
```

```
Bucket started at 12:11:19 UTC Mon 11 September 2017 lasting 10s
```

```
Pkts sent: 10; Lost: 0 (0.0%); Corrupt: 0 (0.0%);
```

```
Misordered: 0 (0.0%); Duplicates: 0 (0.0%)
```

```
Result count: 10
```

```
Min: 1912765.961ms; Max: 1912765.961ms; Mean: 1912765.961ms; StdDev: -2147483.648ms
```

```
One-way Delay (Dest->Source)
```

```
~~~~~
```

```
1 probes per bucket
```

```

Bucket started at 12:11:19 UTC Mon 11 September 2017 lasting 10s
  Pkts sent: 10; Lost: 0 (0.0%); Corrupt: 0 (0.0%);
    Misordered: 0 (0.0%); Duplicates: 0 (0.0%)
  Result count: 10
  Min: -1912765.952ms; Max: -1912765.951ms; Mean: -1912765.951ms; StdDev: -2147483.648ms

```

```

Round Trip Jitter
~~~~~
1 probes per bucket

```

```

Bucket started at 12:11:19 UTC Mon 11 September 2017 lasting 10s
  Pkts sent: 10; Lost: 0 (0.0%); Corrupt: 0 (0.0%);
    Misordered: 0 (0.0%); Duplicates: 0 (0.0%)
  Result count: 9
  Min: 0.000ms; Max: 0.001ms; Mean: 0.000ms; StdDev: 0.000ms

```

```

One-way Jitter (Source->Dest)
~~~~~
1 probes per bucket

```

```

Bucket started at 12:11:19 UTC Mon 11 September 2017 lasting 10s
  Pkts sent: 10; Lost: 0 (0.0%); Corrupt: 0 (0.0%);
    Misordered: 0 (0.0%); Duplicates: 0 (0.0%)
  Result count: 9
  Min: 0.000ms; Max: 0.000ms; Mean: 0.000ms; StdDev: 0.000ms

```

```

One-way Jitter (Dest->Source)
~~~~~
1 probes per bucket

```

```

Bucket started at 12:11:19 UTC Mon 11 September 2017 lasting 10s
  Pkts sent: 10; Lost: 0 (0.0%); Corrupt: 0 (0.0%);
    Misordered: 0 (0.0%); Duplicates: 0 (0.0%)
  Result count: 9
  Min: 0.000ms; Max: 0.001ms; Mean: 0.000ms; StdDev: 0.000ms

```

```

RP/0/RP0/CPU0:ios#ethernet sla on-demand operation type cfm-syn probe domain DOWN0 source
interface tenGigE 0/0/0/10.1 target mep-id 6001
Mon Sep 11 12:12:39.259 UTC
Warning: Burst configuration is present and so this profile cannot be represented in the
MEF-SOAM-PM-MIB configuration tables. However, the statistics are still collected
On-demand operation 2 succesfully created
 / - Completed - statistics will be displayed shortly.
RP/0/RP0/CPU0:ios#show ethernet sla statistics on-demand id 2

```

```

Mon Sep 11 12:13:24.825 UTC
Source: Interface TenGigE0/0/0/10.1, Domain DOWN0
Destination: Target MEP-ID 6001

```

```

=====
On-demand operation ID #2, packet type 'cfm-synthetic-loss-measurement'
Started at 12:12:41 UTC Mon 11 September 2017, runs once for 10s
Frame Loss Ratio calculated every 10s

```

```

One-way Frame Loss (Source->Dest)
~~~~~
1 probes per bucket

```

```

Bucket started at 12:12:41 UTC Mon 11 September 2017 lasting 10s
  Pkts sent: 100; Lost: 0 (0.0%); Corrupt: 0 (0.0%);
    Misordered: 0 (0.0%); Duplicates: 0 (0.0%)

```

```
Result count: 1
Min: 0.000%; Max: 0.000%; Mean: 0.000%; StdDev: 0.000%; Overall: 0.000%
```

```
One-way Frame Loss (Dest->Source)
~~~~~
```

```
1 probes per bucket
```

```
Bucket started at 12:12:41 UTC Mon 11 September 2017 lasting 10s
```

```
Pkts sent: 100; Lost: 0 (0.0%); Corrupt: 0 (0.0%);
Misordered: 0 (0.0%); Duplicates: 0 (0.0%)
```

```
Result count: 1
Min: 0.000%; Max: 0.000%; Mean: 0.000%; StdDev: 0.000%; Overall: 0.000%
```

```
RP/0/RP0/CPU0:ios#show ethernet cfm local meps verbose
```

```
Mon Sep 11 12:13:04.461 UTC
```

```
Domain UP6 (level 6), Service s6
```

```
Up MEP on FortyGigE0/0/1/2.1 MEP-ID 1
```

```
=====
Interface state: Up      MAC address: 008a.960f.c4a8
Peer MEPS: 1 up, 0 with errors, 0 timed out (archived)
Cross-check errors: 0 missing, 0 unexpected
```

```
CCM generation enabled: Yes, 1s (Remote Defect detected: No)
                        CCM processing offloaded to hardware
```

```
AIS generation enabled: No
Sending AIS:           No
Receiving AIS:        No
```

```
No packets sent/received
```

```
Domain DOWN0 (level 0), Service s10
```

```
Down MEP on TenGigE0/0/0/10.1 MEP-ID 2001
```

```
=====
Interface state: Up      MAC address: 008a.960f.c428
Peer MEPS: 1 up, 0 with errors, 0 timed out (archived)
Cross-check errors: 0 missing, 0 unexpected
```

```
CCM generation enabled: Yes, 1s (Remote Defect detected: No)
                        CCM processing offloaded to hardware
```

```
AIS generation enabled: No
Sending AIS:           No
Receiving AIS:        No
```

```
Packet      Sent      Received
-----
DMM          10         0
DMR           0         10
SLM          100         0
SLR           0         100
```

Synthetic Loss Measurement

The synthetic loss measurement mechanism defined in Y.1731 can only be used in point-to-point networks, and only works when there is sufficient flow of data traffic. The difficulties with the Y.1731 loss measurement mechanism was recognized across the industry and hence an alternative mechanism has been defined and standardized for measuring loss of traffic.

This alternative mechanism does not measure the loss of the actual data traffic, but instead injects synthetic CFM frames and measures the loss of these synthetic frames. You can perform a statistical analysis to give an approximation of the loss of data traffic. This technique is called Synthetic Loss Measurement (SLM).

SLM has been included in the latest version of the Y.1731 standard. Use SLA to perform the following measurements:

- One-way loss (Source to Destination)
- One-way loss (Destination to Source)

SLM supports the following:

- All Layer 2 transport interfaces, such as physical, bundle interfaces, Layer2 sub-interfaces, pseudowire Head-end interfaces or attachment circuits.
- Up and Down MEPs.
- Transparent passing of the SLM packets through the MIP without punting it to the software.
- 1000 pps of SLM/SLR traffic.

Configuring Synthetic Loss Measurement

The following section describes how you can configure Synthetic Loss Measurement:

```
RP/0/RP0/CPU0:router (config)# ethernet sla
profile test-slm type cfm-synthetic-loss-measurement
probe
  send packet every 1 seconds
  synthetic loss calculation packets 24
!
schedule
  every 3 minutes for 120 seconds
!
statistics
  measure one-way-loss-sd
  buckets size 1 probes
  buckets archive 5
!
  measure one-way-loss-ds
  buckets size 1 probes
  buckets archive 5
!
!
!
!
interface TenGigE0/0/0/10.1 12transport
encapsulation dot1q 1
ethernet cfm
  mep domain DOWN0 service s10 mep-id 2001
  sla operation profile test-slm target mep-id 6001
!
```

Configuring an On-Demand Ethernet SLA Operation for CFM Synthetic Loss Measurement

To configure an on-demand Ethernet SLA operation for CFM synthetic loss measurement, use this command in privileged EXEC configuration mode:

```
RP/0/RP0/CPU0:router (config)# ethernet sla on-demand operation type
cfm-synthetic-loss-measurement probe domain D1 source interface TenGigE
0/6/1/0 target mac-address 2.3.4
```


Running Configuration

```

RP/0/RP0/CPU0:router# show ethernet sla statistics on-demand id 1
Mon Sep 11 12:12:00.699 UTC
Source: Interface TenGigE0/0/0/10.1, Domain DOWN0
Destination: Target MEP-ID 6001
=====
On-demand operation ID #1, packet type 'cfm-delay-measurement'
RP/0/RP0/CPU0:router#
RP/0/RP0/CPU0:router# show running-config
Mon Sep 11 12:10:18.467 UTC
Building configuration...
!! IOS XR Configuration version = 6.4.1.14I
!! Last configuration change at Mon Sep 11 12:08:16 2017 by root
!
logging console disable
telnet vrf default ipv4 server max-servers 10
username root
group root-lr
group cisco-support
secret 5 $1$QJT3$94M5/wK5J0v/lpAu/wz31/
!
line console
exec-timeout 0 0
!
ethernet cfm
domain UP6 level 6 id null
  service s6 xconnect group g1 p2p p1 id number 6
  mip auto-create all ccm-learning
  continuity-check interval 1s
  mep crosscheck
  mep-id 4001
  !
!
domain DOWN0 level 0 id null
  service s10 down-meps id number 10
  continuity-check interval 1s
  mep crosscheck
  mep-id 6001
  !
!
!
profile test-slm type cfm-synthetic-loss-measurement
  probe
    send packet every 1 seconds
    synthetic loss calculation packets 24
  !
  schedule
    every 3 minutes for 120 seconds
  !
  statistics
    measure one-way-loss-sd
      buckets size 1 probes
      buckets archive 5
    !
    measure one-way-loss-ds
      buckets size 1 probes
      buckets archive 5
  !
interface MgmtEth0/RP0/CPU0/0
shutdown
!

```

```
interface TenGigE0/0/0/0
shutdown
!
interface TenGigE0/0/0/1
shutdown
!
interface TenGigE0/0/0/2
shutdown
!
interface TenGigE0/0/0/3
shutdown
!
interface TenGigE0/0/0/4
shutdown
!
interface TenGigE0/0/0/5
shutdown
!
interface TenGigE0/0/0/6
shutdown
!
interface TenGigE0/0/0/7
shutdown
!
interface TenGigE0/0/0/8
shutdown
!
interface TenGigE0/0/0/9
shutdown
!
interface TenGigE0/0/0/10.1 l2transport
encapsulation dot1q 1
ethernet cfm
    mep domain DOWN0 service s10 mep-id 2001
    sla operation profile DMM target mep-id 6001
    sla operation profile test-slm target mep-id 6001
!
!
!
interface TenGigE0/0/0/11
shutdown
!
interface TenGigE0/0/0/12
shutdown
!
interface TenGigE0/0/0/13
shutdown
!
interface TenGigE0/0/0/14
shutdown
!
interface TenGigE0/0/0/15
shutdown
!
interface TenGigE0/0/0/16
shutdown
!
interface TenGigE0/0/0/17
shutdown
!
interface TenGigE0/0/0/18
shutdown
!
interface TenGigE0/0/0/19
```

```
shutdown
!
interface TenGigE0/0/0/20
shutdown
!
interface TenGigE0/0/0/21
shutdown
!
interface TenGigE0/0/0/22
shutdown
!
interface TenGigE0/0/0/23
shutdown
!
interface TenGigE0/0/0/24
shutdown
!
interface TenGigE0/0/0/25
shutdown
!
interface TenGigE0/0/0/26
shutdown
!
interface TenGigE0/0/0/27
shutdown
!
interface TenGigE0/0/0/28
shutdown
!
interface TenGigE0/0/0/29
shutdown
!
interface TenGigE0/0/0/30
shutdown
!
!
interface TenGigE0/0/0/31
shutdown
!
interface TenGigE0/0/0/32
shutdown
!
interface TenGigE0/0/0/33
shutdown
!
interface TenGigE0/0/0/34
shutdown
!
interface TenGigE0/0/0/35
shutdown
!
interface TenGigE0/0/0/36
shutdown
!
interface TenGigE0/0/0/37
shutdown
!
interface TenGigE0/0/0/38
shutdown
!
interface TenGigE0/0/0/39
shutdown
!
interface TenGigE0/0/1/0/1
```

```

shutdown
!
interface TenGigE0/0/1/0/2
 shutdown
!
interface TenGigE0/0/1/0/3
 shutdown
!
controller Optics0/0/1/0
 breakout 4x10
!
interface HundredGigE0/0/1/1
 shutdown
!
interface FortyGigE0/0/1/2.1 l2transport
 encapsulation dot1q 1
 ethernet cfm
  mep domain UP6 service s6 mep-id 1
   sla operation profile DMM target mep-id 6001
   sla operation profile test-slm target mep-id 6001
!
!
!
l2vpn
 xconnect group g1
  p2p p1
   interface TenGigE0/0/0/10.1
   interface FortyGigE0/0/1/2.1
!
!
!
end

```

Verification

Round Trip Delay

~~~~~

1 probes per bucket

Bucket started at 12:11:19 UTC Mon 11 September 2017 lasting 10s

Pkts sent: 10; Lost: 0 (0.0%); Corrupt: 0 (0.0%);  
 Misordered: 0 (0.0%); Duplicates: 0 (0.0%)

Result count: 10

Min: 0.009ms; Max: 0.010ms; Mean: 0.009ms; StdDev: 0.000ms

One-way Delay (Source->Dest)

~~~~~

1 probes per bucket

Bucket started at 12:11:19 UTC Mon 11 September 2017 lasting 10s

Pkts sent: 10; Lost: 0 (0.0%); Corrupt: 0 (0.0%);
 Misordered: 0 (0.0%); Duplicates: 0 (0.0%)

Result count: 10

Min: 1912765.961ms; Max: 1912765.961ms; Mean: 1912765.961ms; StdDev: -2147483.648ms

One-way Delay (Dest->Source)

~~~~~

1 probes per bucket

Bucket started at 12:11:19 UTC Mon 11 September 2017 lasting 10s

Pkts sent: 10; Lost: 0 (0.0%); Corrupt: 0 (0.0%);  
 Misordered: 0 (0.0%); Duplicates: 0 (0.0%)

```
Result count: 10
Min: -1912765.952ms; Max: -1912765.951ms; Mean: -1912765.951ms; StdDev: -2147483.648ms
```

```
Round Trip Jitter
~~~~~
1 probes per bucket
```

```
Bucket started at 12:11:19 UTC Mon 11 September 2017 lasting 10s
 Pkts sent: 10; Lost: 0 (0.0%); Corrupt: 0 (0.0%);
 Misordered: 0 (0.0%); Duplicates: 0 (0.0%)
 Result count: 9
 Min: 0.000ms; Max: 0.001ms; Mean: 0.000ms; StdDev: 0.000ms
```

```
One-way Jitter (Source->Dest)
~~~~~
1 probes per bucket
```

```
Bucket started at 12:11:19 UTC Mon 11 September 2017 lasting 10s
  Pkts sent: 10; Lost: 0 (0.0%); Corrupt: 0 (0.0%);
    Misordered: 0 (0.0%); Duplicates: 0 (0.0%)
  Result count: 9
  Min: 0.000ms; Max: 0.000ms; Mean: 0.000ms; StdDev: 0.000ms
```

```
One-way Jitter (Dest->Source)
~~~~~
1 probes per bucket
```

```
Bucket started at 12:11:19 UTC Mon 11 September 2017 lasting 10s
 Pkts sent: 10; Lost: 0 (0.0%); Corrupt: 0 (0.0%);
 Misordered: 0 (0.0%); Duplicates: 0 (0.0%)
 Result count: 9
 Min: 0.000ms; Max: 0.001ms; Mean: 0.000ms; StdDev: 0.000ms
```

## Unidirectional Link Detection Protocol

Unidirectional Link Detection (UDLD) is a single-hop physical link protocol for monitoring an ethernet link, including both point-to-point and shared media links. This is a Cisco-proprietary protocol to detect link problems, which are not detected at the physical link layer. This protocol is specifically targeted at possible wiring errors, when using unbundled fiber links, where there can be a mismatch between the transmitting and receiving connections of a port.

### Limitations

- UDLD must not be enabled on a Switched Port Analyzer (SPAN) source or a destination port.
- UDLD must not be enabled on a port that acts as a source or destination port for SPAN.

## Types of Fault Detection

UDLD can detect these types of faults:

- Transmit faults — These are cases where there is a failure in transmitting packets from the local port to the peer device, but packets are being received from the peer. These faults are caused by failure of the

physical link (where notification at layer 1 of unidirectional link faults is not supported by the media) as well as packet path faults on the local or peer device.

- **Miswiring faults** — These are cases where the receiving and transmitting sides of a port on the local device are connected to different peer ports (on the same device or on different devices). This can occur when using unbundled fibers to connect fiber optic ports.
- **Loopback faults** — These are cases where the receiving and transmitting sides of a port are connected to each other, creating a loopback condition. This can be an intentional mode of operation, for certain types of testing, but UDLD must not be used in these cases.
- **Receive faults** — The protocol includes a heartbeat signal that is transmitted at a negotiated periodic interval to the peer device. Missed heartbeats can therefore be used to detect failures on the receiving side of the link (where they do not result in interface state changes). These could be caused by a unidirectional link with a failure only affecting the receiving side, or by a link which has developed a bidirectional fault. This detection depends on reliable, regular packet transmission by the peer device. For this reason, the UDLD protocol has two (configurable) modes of operation which determine the behavior on a heartbeat timeout. These modes are described in the section [UDLD Modes of Operation, on page 142](#).

## UDLD Modes of Operation

UDLD can operate in these modes:

- **Normal mode:** In this mode, if a `Receive Fault` is detected, the user is informed and no further action is taken.
- **Aggressive mode:** In this mode, if a `Receive Fault` is detected, the user is informed and the affected port is disabled.




---

**Note** The difference of behavior between normal and aggressive modes is only seen in case of neighbor timeout. In all other cases, irrespective of the normal or aggressive mode, the system error disables a link once a unidirectional link is detected.

---

## Configure UDLD

UDLD is configured for each interface. The interface must be a physical ethernet interface.

Perform the following steps to configure UDLD protocol on an interface:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface TenGigE 0/1/0/0
```




---

**Note** The example indicates an 8-port 10-Gigabit Ethernet interface in modular services card slot 1.

---

## Running Configuration

```

RP/0/RSP0/CPU0:router(config-if)# ethernet udldRP/0/RSP0/CPU0:router(config-if-udld)# mode?
RP/0/RP0/CPU0:IOS(config)#interface tenGigE 0/0/0/0

RP/0/RP0/CPU0:IOS(config-if)#ethernet udld
RP/0/RP0/CPU0:IOS(config-if-udld)#mode ?

 aggressive Run UDLD in aggressive mode
 normal Run UDLD in normal mode

RP/0/RP0/CPU0:IOS(config-if-udld)#mode aggressive
RP/0/RP0/CPU0:IOS(config-if-udld)#message-time ?

 <7-90> 'Mslow' message time (in seconds) to use for the UDLD protocol

RP/0/RP0/CPU0:IOS(config-if-udld)#message-time 50
RP/0/RP0/CPU0:IOS(config-if-udld)#destination mac-address ?
 H.H.H
 A valid multicast MAC address

 cisco-l2cp Use the Cisco L2CP MAC address (used by CDP)
 ieee-slow-protocols Use the IEEE slow protocol destination MAC address
RP/0/RP0/CPU0:IOS(config-if-udld)#destination mac-address 0100.5e01.0101

RP/0/RP0/CPU0:IOS(config-if-udld)#logging disable
RP/0/RP0/CPU0:IOS(config-if-udld)#commit

RP/0/RP0/CPU0:IOS(config-if-udld)#end

RP/0/RP0/CPU0:IOS#sh run interface tenGigE 0/0/0/0
interface TenGigE0/0/0/0

 ethernet udld

 mode aggressive

 message-time 50

 destination mac-address 0100.5e01.0101

 logging disable

 !

 !

```

## Verification

```

RP/0/RP0/CPU0:IOS#sh ethernet udld interfaces

Device ID: 00:8a:96:e1:20:d8

Device name: IOS

Interface TenGigE0/0/0/0

```

```

Port state: Up
Main FSM state: Advertising
Detection FSM state: Unknown
Message interval: 7 seconds
Timeout interval: 5 seconds
Destination MAC: 01:00:5e:01:01:01

```

```
RP/0/RP0/CPU0:IOS#sh ethernet udld statistics tenGigE 0/0/0/0
```

```
Interface TenGigE0/0/0/0
```

```
Counters last cleared: 00:01:18 ago
```

```
Main FSM transitions (to each state)
```

```
Link up: 1
```

```
Detection: 0
```

```
Advertise: 1
```

```
Port shutdown: 0
```

```
UDLD inactive: 0
```

```
Detection FSM transitions (to each state)
```

```
Unknown: 0
```

```
Bidirectional: 0
```

```
Unidirectional: 0
```

```
Neighbor mismatch: 0
```

```
Loopback: 0
```

```
Rx packet counts
```

```
Probe: 0
```

```
Echo: 0
```

```
Flush: 0
```

```
Invalid packets (dropped): 0
```

```
Tx packet counts
```

```
Probe: 19
```

```
Echo: 0
```

```
Flush: 0
```

```
Unable to send (dropped): 0
```



```
RP/0/RP0/CPU0:IOS#
RP/0/RP0/CPU0:IOS#sh ethernet udd daemon database
Interface TenGigE0/0/0/0
```

| Item                         | Value                  |
|------------------------------|------------------------|
| Interface handle             | Te0/0/0/0 (0x00000200) |
| Name                         | Te0/0/0/0              |
| Name (long internal format)  | TenGigE0_0_0_0         |
| Configured ?                 | TRUE                   |
| Caps add in progress ?       | FALSE                  |
| Caps remove in progress ?    | FALSE                  |
| Caps added ?                 | TRUE                   |
| Protocol start pending ?     | FALSE                  |
| Protocol running ?           | TRUE                   |
| Registered for packet I/O ?  | TRUE                   |
| Aggressive mode ?            | TRUE                   |
| Logging enabled ?            | FALSE                  |
| Error disabled on start ?    | FALSE                  |
| Error disabled during ISSU ? | FALSE                  |
| Attributes read ?            | TRUE                   |
| Pending state down nfn ?     | FALSE                  |
| Message time                 | 50                     |

## Link Loss Forwarding

The Cisco ASR9000 Series Routers support Link Loss Forwarding (LLF). LLF uses CFM to transmit notification of a signal loss or fault across the network. If a local AC goes down, LLF sends signals across to the neighboring device.

The following packet types indicate a fault in a network:

- Continuity Check Message (CCM). Also known as CFM
- Alarm Indication Signal (AIS)
- Client Signal Frame (CSF)

When the system receives a CCM or AIS with fault indication, or a CSF error packet, CFM communicates with Ether-MA to TX-disable the interface.

Ether-MA handles owner channel communication and resyncs from CFMD, L2VPN, and other Ether-MA processes.

## Restrictions for LLF

- LLF isn't permitted on sub-interfaces.
- LLF is only permitted on up MEPS.
- The system runs a damping timer to govern transitions from an interface being TX-disabled to an interface being TX-enabled. The following restrictions apply on such a scenario:

- The period of the damping timer is given by three times the configured CCM interval. However, you can't configure the damping timer.
- The system doesn't provide damping for transitions from TX-enabled to TX-disabled.

## Configure Link Loss Forwarding

This section describes how to configure LLF on a network by using the `propagate-remote-status` config command.

```
/* Enable LLF */
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface GigabitEthernet0/2/0/0
RP/0/RP0/CPU0:router(config-cfm)# ethernet cfm
RP/0/RP0/CPU0:router(config-cfm)# mep domain foo service bar mep-id 1
RP/0/RP0/CPU0:router(config-cfm)# propagate remote-status
RP/0/RP0/CPU0:router(config-cfm)# commit
```

### Optional Configuration for Client Signal Fail (CSF)



**Note** CSF configuration is required for inter-operation with certain client-end setups that contain devices from other clients.

```
ethernet cfm
 domain <domain> level <level> service <service> <type>
 csf [<interval> {1s | 1m}] [cos <cos>]
 log csf
```

### Running Configuration

```
ethernet cfm
 domain dom1 level 1
 service ser1 bridge group up-meps bridge-domain up-mep
 continuity-check interval 1m
 csf interval 1m cos 4
 csf-logging
 !
 !
 !
interface GigabitEthernet0/2/0/0
 ethernet cfm
 mep domain dom1 service ser1 mep-id 1
 propagate-remote-status
 !
 !
 !
```

### Verification

```
show ethernet cfm interfaces [<interface>] llf [location <node>]
Defects (from at least one peer MEP):
A - AIS received I - Wrong interval
R - Remote Defect received V - Wrong Level
L - Loop (our MAC received) T - Timed out (archived)
C - Config (our ID received) M - Missing (cross-check)
X - Cross-connect (wrong MAID) U - Unexpected (cross-check)
P - Peer port down F - CSF received
```

```
GigabitEthernet0/1/0/0
MEP Defects Restore Timer

100 R Not running
101 None 10s remaining
102 RPF Not running

GigabitEthernet0/1/0/1
MEP Defects Restore Timer

110 None 3s remaining

GigabitEthernet0/1/0/2
MEP Defects Restore Timer

120 P Not running
```





## CHAPTER 6

# Configuring Integrated Routing and Bridging

This module describes the configuration of Integrated Routing and Bridging (IRB). IRB provides the ability to exchange traffic between bridging services and a routed interface using a Bridge-Group Virtual Interface (BVI).

### Feature History for IRB

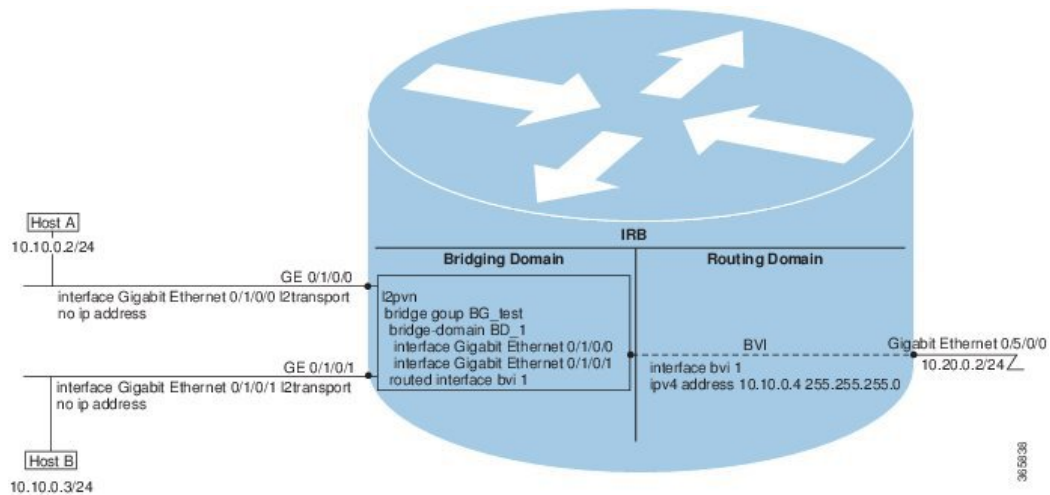
| Release       | Modification                 |
|---------------|------------------------------|
| Release 6.1.1 | This feature was introduced. |

- [IRB Introduction, on page 149](#)
- [Bridge-Group Virtual Interface, on page 150](#)
- [Supported Features on a BVI, on page 150](#)
- [BVI Interface and Line Protocol States, on page 151](#)
- [Prerequisites for Configuring IRB, on page 151](#)
- [Restrictions for Configuring IRB, on page 152](#)
- [How to Configure IRB, on page 153](#)
- [Additional Information on IRB, on page 159](#)
- [Packet Flows Using IRB, on page 159](#)
- [Configuration Examples for IRB, on page 161](#)

## IRB Introduction

IRB provides the ability to route between a bridge group and a routed interface using a BVI. The BVI is a virtual interface within the router that acts like a normal routed interface. A BVI is associated with a single bridge domain and represents the link between the bridging and the routing domains on the router. To support receipt of packets from a bridged interface that are destined to a routed interface, the BVI must be configured with the appropriate IP addresses and relevant Layer 3 attributes.

Figure 12: IRB Functional View and Configuration Elements



## Bridge-Group Virtual Interface

The BVI is a virtual interface within the router that acts like a normal routed interface. The BVI does not support bridging itself, but acts as a gateway for the corresponding bridge-domain to a routed interface within the router.

BVI supports only Layer 3 attributes, and has the following characteristics:

- Uses a MAC address taken from the local chassis MAC address pool, unless overridden at the BVI interface.
- Is configured as an interface type using the **interface bvi** command and uses an IPv4 address that is in the same subnet as the hosts on the segments of the bridged domain. The BVI also supports secondary addresses.
- The BVI identifier is independent of the bridge-domain identifier. These identifiers do not need to correlate like they do in Cisco IOS software.
- Is associated to a bridge group using the **routed interface bvi** command.
- BVI interfaces support a number range of 1 to 4294967295.

## Supported Features on a BVI

- These interface commands are supported on a BVI:
  - **arp purge-delay**
  - **arp timeout**
  - **bandwidth** (The default is 10 Gbps and is used as the cost metric for routing protocols for the BVI)
  - **ipv4**

- **ipv6**
  - **mac-address**
  - **shutdown**
- The BVI supports IP helper addressing and secondary IP addressing.
  - BVI does not support MTU configuration using **mtu** command, which is for physical interfaces. However, **ip mtu** and **ipv6 mtu** commands, which are logical interface commands, are supported.

## BVI Interface and Line Protocol States

Like typical interface states on the router, a BVI has both an Interface and Line Protocol state.

- The BVI interface state is Up when the following occurs:
  - The BVI interface is created.
  - The bridge-domain that is configured with the **routed interface bvi** command has at least one available active bridge port (Attachment circuit [AC] or pseudowire [PW]).



---

**Note** A BVI will be moved to the Down state if all of the bridge ports (Ethernet flow points [EFPs]) associated with the bridge domain for that BVI are down. However, the BVI will remain up if at least one bridgeport is up, even if all EFPs are down.

---

- These characteristics determine when the the BVI line protocol state is up:
  - The bridge-domain is in Up state.
  - The BVI IP address is not in conflict with any other IP address on another active interface in the router.

## Prerequisites for Configuring IRB

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.

Before configuring IRB, be sure that these tasks and conditions are met:

- Know the IP addressing and other Layer 3 information to be configured on the bridge virtual interface (BVI).
- Complete MAC address planning if you decide to override the common global MAC address for all BVIs.
- Be sure that the BVI network address is being advertised by running static or dynamic routing on the BVI interface.

# Restrictions for Configuring IRB

Before configuring IRB, consider these restrictions:

- Only one BVI can be configured in any bridge domain.
- The same BVI can not be configured in multiple bridge domains.
- MTU configuration and fragmentation of packets is not supported on BVI interfaces.
- The following areas are *not* supported on the BVI:
  - Access Control Lists (ACLs). However, Layer 2 ACLs can be configured on each Layer 2 port of the bridge domain.
  - IP fast reroute (FRR)
  - TI-LFA
  - SR
  - LDP
  - NetFlow
  - MoFRR
  - Quality of Service (QoS)
  - Traffic mirroring
  - Unnumbered interface for BVI
  - Video monitoring (Vidmon)
  - IRB with 802.1ah (BVI and Provider Backbone Bridge (PBB) should not be configured in the same bridge domain).
  - PIM snooping. (Need to use selective flood.)
  - VRF-aware DHCP relay
- The following areas are *not* supported on the Layer2 bridging (with BVI):
  - Static mac entry configuration in Bridge.
  - Mac ageing configuration at global config mode.
  - MAC Learning Disable.
  - Vlan rewrite.
- QOS configuration on BVI interface is not supported for egress.
- Label allocation mode per-CE with BVI is not supported in an access network along with PE-CE protocols enabled.



# How to Configure IRB

This section includes the following configuration tasks:

## Configuring the Bridge Group Virtual Interface

To configure a BVI, complete the following steps.

### Configuration Guidelines

Consider the following guidelines when configuring the BVI:

- The BVI must be assigned an IPv4 or IPv6 address that is in the same subnet as the hosts in the bridged segments.
- If the bridged network has multiple IP networks, then the BVI must be assigned secondary IP addresses for each network.

### SUMMARY STEPS

1. **configure**
2. **interface bvi** *identifier*
3. **ipv4 address** *ipv4-address mask* [**secondary**] **ipv6 address** *ipv6-prefix/prefix-length* [**eui-64**] [**route-tag** *route-tag value*]
4. **arp purge-delay** *seconds*
5. **arp timeout** *seconds*
6. **bandwidth** *rate*
7. **end** or **commit**

### DETAILED STEPS

---

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

```
Router# configure
```

Enters the global configuration mode.

**Step 2** **interface bvi** *identifier***Example:**

```
Router(config)# interface bvi 1
```

Specifies or creates a BVI, where *identifier* is a number from 1 to 65535.

**Step 3** **ipv4 address** *ipv4-address mask* [**secondary**] **ipv6 address** *ipv6-prefix/prefix-length* [**eui-64**] [**route-tag** *route-tag value*]**Example:**

```
Router(config-if)# ipv4 address 10.10.0.4 255.255.255.0
```

Specifies a primary or secondary IPv4 address or an IPv6 address for an interface.

#### Step 4 **arp purge-delay** *seconds*

##### **Example:**

```
Router(config-if)#arp purge-delay 120
```

(Optional) Specifies the amount of time (in *seconds*) to delay purging of Address Resolution Protocol (ARP) table entries when the interface goes down.

The range is 1 to 65535. By default purge delay is not configured.

#### Step 5 **arp timeout** *seconds*

##### **Example:**

```
Router(config-if)# arp timeout 12200
```

(Optional) Specifies how long dynamic entries learned on the interface remain in the ARP cache.

The range is 30 to 2144448000 seconds. The default is 14,400 seconds (4 hours).

#### Step 6 **bandwidth** *rate*

##### **Example:**

```
Router(config-if)# bandwidth 1000000
```

(Optional) Specifies the amount of bandwidth (in kilobits per second) to be allocated on the interface. This number is used as the cost metric in routing protocols for the BVI.

The range is 0 to 4294967295. The default is 10000000 (10 Gbps).

#### Step 7 **end** or **commit**

##### **Example:**

```
Router(config-if)# end
```

or

```
Router(config-if)# commit
```

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

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

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 Layer 2 AC Interfaces

To configure the Layer 2 AC interfaces for routing by a BVI, complete the following steps.

### SUMMARY STEPS

1. **configure**
2. **interface [HundredGigE | TenGigE] l2transport**
3. **end** or **commit**

### DETAILED STEPS

---

#### Step 1 **configure**

**Example:**

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

Enters global configuration mode.

#### Step 2 **interface [HundredGigE | TenGigE] l2transport**

**Example:**

```
RP/0/RP0/CPU0:router(config)# interface TenGigE 0/1/0/0.1 l2transport
```

Enables Layer 2 transport mode on a Gigabit Ethernet or 10-Gigabit Ethernet interface or subinterface and enters interface or subinterface configuration mode.

#### Step 3 **end** or **commit**

**Example:**

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

or

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

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

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

- 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 a Bridge Group and Assigning Interfaces to a Bridge Domain

To configure a bridge group and assign interfaces to a bridge domain, complete the following steps.

### SUMMARY STEPS

1. **configure**
2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **interface** [HundredGigE | TenGigE]
6. **end** or **commit**

### DETAILED STEPS

---

#### Step 1 **configure**

**Example:**

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

#### Step 2 **l2vpn**

**Example:**

```
RP/0/RP0/CPU0:router(config)# l2vpn
Enters L2VPN configuration mode.
```

#### Step 3 **bridge group** *bridge-group-name*

**Example:**

```
RP/0/RP0/CPU0:router(config-l2vpn)# bridge group 10
Creates a bridge group and enters L2VPN bridge group configuration mode.
```

**Step 4** `bridge-domain` *bridge-domain-name***Example:**

```
RP/0/RP0/CPU0:router(config-l2vpn-bg) # bridge-domain BD_1
```

Creates a bridge domain and enters L2VPN bridge group bridge domain configuration mode.

**Step 5** `interface` [**HundredGigE** | **TenGigE**]**Example:**

```
RP/0/RP0/CPU0:router(config-l2vpn-bg-bd) # interface HundredGigE 0/1/0/0.1
```

Associates the 100-Gigabit Ethernet or 10-Gigabit Ethernet interface with the specified bridge domain and enters L2VPN bridge group bridge domain attachment circuit configuration mode.

Repeat this step for as many interfaces as you want to associate with the bridge domain.

**Step 6** `end` or `commit`**Example:**

```
RP/0/RP0/CPU0:router(config-l2vpn-bg-bd-ac) # end
```

or

```
RP/0/RP0/CPU0:router(config-l2vpn-bg-bd-ac) # commit
```

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

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

- 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.

---

## Associating the BVI as the Routed Interface on a Bridge Domain

To associate the BVI as the routed interface on a bridge domain, complete the following steps.

### SUMMARY STEPS

1. **configure**

2. **l2vpn**
3. **bridge group** *bridge-group-name*
4. **bridge-domain** *bridge-domain-name*
5. **routed interface bvi** *identifier*
6. **end** or **commit**

## DETAILED STEPS

---

### Step 1 **configure**

#### Example:

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

Enters global configuration mode.

### Step 2 **l2vpn**

#### Example:

```
RP/0/RP0/CPU0:router(config)# l2vpn
```

Enters L2VPN configuration mode.

### Step 3 **bridge group** *bridge-group-name*

#### Example:

```
RP/0/RP0/CPU0:router(config-l2vpn)# bridge group BG_test
```

Creates a bridge group and enters L2VPN bridge group configuration mode.

### Step 4 **bridge-domain** *bridge-domain-name*

#### Example:

```
RP/0/RP0/CPU0:router(config-l2vpn-bg)# bridge-domain 1
```

Creates a bridge domain and enters L2VPN bridge group bridge domain configuration mode.

### Step 5 **routed interface bvi** *identifier*

#### Example:

```
RP/0/RP0/CPU0:router(config-l2vpn-bg-bd)# routed interface bvi 1
```

Associates the specified BVI as the routed interface for the interfaces assigned to the bridge domain.

### Step 6 **end** or **commit**

#### Example:

```
RP/0/RP0/CPU0:router(config-l2vpn-bg-bd)# end
```

OR

```
RP/0/RP0/CPU0:router(config-l2vpn-bg-bd)# commit
```

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

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

- 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.

## Displaying Information About a BVI

To display information about BVI status and packet counters, use the following commands:

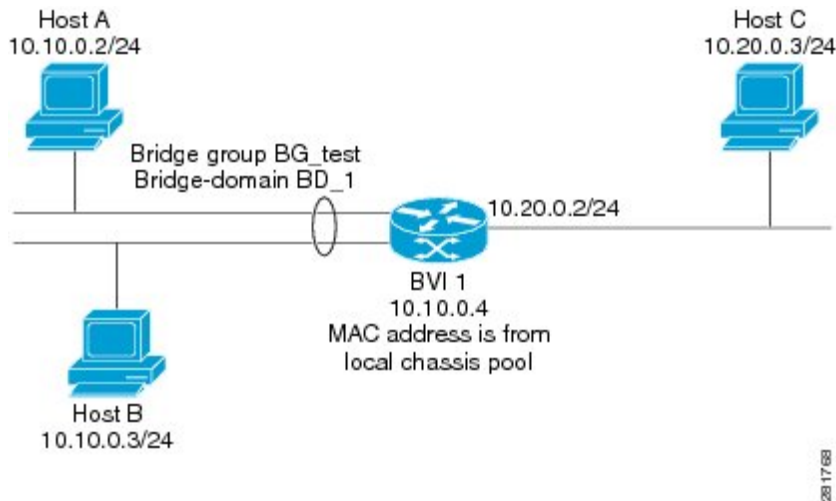
|                                                                                                                        |                                                                                            |
|------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------|
| <b>show interfaces bvi</b> <i>identifier</i> [ <b>accounting</b>   <b>brief</b>   <b>description</b>   <b>detail</b> ] | Displays interface status, line protocol state, and packet counters for the specified BVI. |
| <b>show adjacency bvi</b> <i>identifier</i> [ <b>detail</b>   <b>remote</b> ]                                          | Displays packet and byte transmit counters per adjacency to the specified BVI.             |
| <b>show l2vpn bridge-domain detail</b>                                                                                 | Displays the reason that a BVI is down.                                                    |

## Additional Information on IRB

### Packet Flows Using IRB

This figure shows a simplified functional diagram of an IRB implementation to describe different packet flows between Host A, B, and C. In this example, Host C is on a network with a connection to the same router. In reality, another router could be between Host C and the router shown.

Figure 13: IRB Packet Flows Between Hosts



When IRB is configured on a router, the following processing happens:

- ARP requests are resolved between the hosts and BVI that are part of the bridge domain.
- All packets from a host on a bridged interface go to the BVI if the destination MAC address matches the BVI MAC address. Otherwise, the packets are bridged.
- For packets destined for a host on a routed network, the BVI forwards the packets to the routing engine before sending them out a routed interface.
- All packets either from or destined to a host on a bridged interface go to the BVI first (unless the packet is destined for a host on the bridge domain).
- For packets that are destined for a host on a segment in the bridge domain that come in to the router on a routed interface, the BVI forwards the packet to the bridging engine, which forwards it through the appropriate bridged interface.

## Packet Flows When Host A Sends to Host B on the Bridge Domain

When Host A sends data to Host B in the bridge domain on the 10.10.0.0 network, no routing occurs. The hosts are on the same subnet and the packets are bridged between their segment interfaces on the router.

## Packet Flows When Host A Sends to Host C From the Bridge Domain to a Routed Interface

Using host information from this figure, the following occurs when Host A sends data to Host C from the IRB bridging domain to the routing domain:

- Host A sends the packet to the BVI (as long as any ARP request is resolved between the host and the BVI). The packet has the following information:
  - Source MAC address of host A.
  - Destination MAC address of the BVI.



- Since Host C is on another network and needs to be routed, the BVI forwards the packet to the routed interface with the following information:
  - IP source MAC address of Host A (10.10.0.2) is changed to the MAC address of the BVI (10.10.0.4).
  - IP destination address is the IP address of Host C (10.20.0.3).
- Interface 10.20.0.2 sees receipt of a packet from the routed BVI 10.10.0.4. The packet is then routed through interface 10.20.0.2 to Host C.

## Packet Flows When Host C Sends to Host B From a Routed Interface to the Bridge Domain

Using host information from this figure, the following occurs when Host C sends data to Host B from the IRB routing domain to the bridging domain:

- The packet comes into the routing domain with the following information:
  - MAC source address—MAC of Host C.
  - MAC destination address—MAC of the 10.20.0.2 ingress interface.
  - IP source address—IP address of Host C (10.20.0.3).
  - IP destination address—IP address of Host B (10.10.0.3).
- When interface 10.20.0.2 receives the packet, it looks in the routing table and determines that the packet needs to be forwarded to the BVI at 10.10.0.4.
- The routing engine captures the packet that is destined for the BVI and forwards it to the BVI's corresponding bridge domain. The packet is then bridged through the appropriate interface if the destination MAC address for Host B appears in the bridging table, or is flooded on all interfaces in the bridge group if the address is not in the bridging table.

## Configuration Examples for IRB

This section provides the following configuration examples:

### Basic IRB Configuration: Example

The following example shows how to perform the most basic IRB configuration:

```
! Configure the BVI and its IPv4 address
!
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)#interface bvi 1
RP/0/RP0/CPU0:router(config-if)#ipv4 address 10.10.0.4 255.255.255.0
RP/0/RP0/CPU0:router(config-if)# exit
!
! Configure the Layer 2 AC interface
!
```

```

RP/0/RP0/CPU0:router(config)#interface HundredGigE 0/1/0/0 l2transport
RP/0/RP0/CPU0:router(config-if)# exit
!
! Configure the L2VPN bridge group and bridge domain and assign interfaces
!
RP/0/RP0/CPU0:router(config)#l2vpn
RP/0/RP0/CPU0:router(config-l2vpn)#bridge group 10
RP/0/RP0/CPU0:router(config-l2vpn-bg)#bridge-domain 1
RP/0/RP0/CPU0:router(config-l2vpn-bg-bd)#interface HundredGigE 0/1/0/0
RP/0/RP0/CPU0:router(config-l2vpn-bg-bd-if)# exit
!
! Associate a BVI to the bridge domain
!
RP/0/RP0/CPU0:router(config-l2vpn-bg-bd)# routed interface bvi 1
RP/0/RP0/CPU0:router(config-l2vpn-bg-bd)# commit

```

## IPv4 Addressing on a BVI Supporting Multiple IP Networks: Example

The following example shows how to configure secondary IPv4 addresses on a BVI that supports bridge domains for the 10.10.10.0/24, 10.20.20.0/24, and 10.30.30.0/24 networks. In this example, the BVI must have an address on each of the bridge domain networks:

```

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)#interface bvi 1
RP/0/RP0/CPU0:router(config-if)#ipv4 address 10.10.10.4 255.255.255.0
RP/0/RP0/CPU0:router(config-if)#ipv4 address 10.20.20.4 255.255.255.0 secondary
RP/0/RP0/CPU0:router(config-if)#ipv4 address 10.30.30.4 255.255.255.0 secondary
RP/0/RP0/CPU0:router(config-if)# commit

```

## IRB With BVI and VRRP Configuration: Example

This example shows a partial router configuration for the relevant configuration areas for IRB support of a BVI and VRRP:




---

**Note** VRRPv6 is also supported.

---

```

l2vpn
 bridge group IRB
 bridge-domain IRB-EDGE
 interface TenGigE0/0/0/8
 !
 routed interface BVI 100
 !
 interface TenGigE0/0/0/8
 l2transport
 !
 interface BVI 100
 ipv4 address 10.21.1.1 255.255.255.0
 !
 router vrrp
 interface BVI 100
 address-family ipv4
 vrrp 1
 address 10.21.1.100

```

```
priority 100
!
```





## CHAPTER 7

# Configuring Link Bundling

The Link Bundling feature allows you to group multiple point-to-point links together into one logical link and provide higher bidirectional bandwidth, redundancy, and load balancing between two routers. A virtual interface is assigned to the bundled link. The component links can be dynamically added and deleted from the virtual interface.

The virtual interface is treated as a single interface on which one can configure an IP address and other software features used by the link bundle. Packets sent to the link bundle are forwarded to one of the links in the bundle.

A link bundle is simply a group of ports that are bundled together and act as a single link. The advantages of link bundles are as follows:

- Multiple links can span several line cards to form a single interface. Thus, the failure of a single link does not cause a loss of connectivity.
- Bundled interfaces increase bandwidth availability, because traffic is forwarded over all available members of the bundle. Therefore, traffic can flow on the available links if one of the links within a bundle fails. Bandwidth can be added without interrupting packet flow.

Cisco IOS XR software supports the following method of forming bundles of Ethernet interfaces:

- IEEE 802.3ad—Standard technology that employs a Link Aggregation Control Protocol (LACP) to ensure that all the member links in a bundle are compatible. Links that are incompatible or have failed are automatically removed from a bundle.
- [Limitations and Compatible Characteristics of Ethernet Link Bundles, on page 166](#)
- [Configuring Ethernet Link Bundles, on page 167](#)
- [Configuring LACP Fallback, on page 172](#)
- [VLANs on an Ethernet Link Bundle, on page 173](#)
- [Configuring VLAN over Bundles, on page 174](#)
- [LACP Short Period Time Intervals, on page 178](#)
- [Configuring the Default LACP Short Period Time Interval, on page 178](#)
- [Configuring Custom LACP Short Period Time Intervals, on page 180](#)
- [Bundle Consistency Checker, on page 186](#)
- [Information About Configuring Link Bundling, on page 190](#)

# Limitations and Compatible Characteristics of Ethernet Link Bundles

This list describes the properties and limitations of ethernet link bundles:

- The router supports mixed speed bundles. Mixed speed bundles allow member links of different bandwidth to be configured as active members in a single bundle. The ratio of the bandwidth for bundle members must not exceed 10. Also, the total weight of the bundle must not exceed 64. For example, 100Gbps link and 10Gbps links can be active members in a bundle and load-balancing on member links is based on bandwidth weightage.
- The weight of each bundle member is the ratio of its bandwidth to the lowest bandwidth member. Total weight of the bundle is the sum of weights or relative bandwidth of each bundle member. Since the weight for a bundle member is greater than or equal to 1 and less than or equal to 10, the total member of links in a bundle is less than 64 in mixed bundle case.
- Any type of Ethernet interfaces can be bundled, with or without the use of LACP (Link Aggregation Control Protocol).
- A single router can support a maximum of 256 bundle interfaces. Link bundles of only physical interfaces are supported.
- When enabling HQoS profile, the maximum available trunks by default (bundle main + sub-interfaces) are 256. If you need more trunks, configure the **hw-module profile bundle-scale <256/512/1024>** command. With HQoS enabled on bundle interfaces, the maximum priority level supported is 4.
- The following limitations apply to the number of supported bundle members with HQoS profile on Layer2 and Layer3 interfaces:
  - Maximum of 1024 trunks (128 physical interfaces + 896 sub-interfaces) and 16 bundle members.
  - Maximum of 256 trunks (128 physical interfaces + 128 sub-interfaces) and 64 bundle members.
  - Maximum of 512 trunks (128 physical interfaces + 384 sub-interfaces) and 32 bundle members.
- The following limitations apply to bundle sub-interfaces and the number of members per bundle :
  - Maximum of 1024 bundle sub-interfaces, each containing up to 16 member-links.
  - Maximum of 256 bundle sub-interfaces, each containing up to 64 member-links
  - Maximum of 512 bundle sub-interfaces, each containing up to 32 member-links
- Physical layer and link layer configuration are performed on individual member links of a bundle.
- Configuration of network layer protocols and higher layer applications is performed on the bundle itself.
- IPv4 and IPv6 addressing is supported on ethernet link bundles.
- A bundle can be administratively enabled or disabled.
- Each individual link within a bundle can be administratively enabled or disabled.
- Ethernet link bundles are created in the same way as Ethernet channels, where the user enters the same configuration on both end systems.

- QoS is supported and is applied proportionally on each bundle member.
- In case static MAC address is configured on a bundle-ether interface, the following limitations are applied:
  - Locally generated packets, such as ICMP, BGP, and so on, going out from the interface have the source MAC address as the statically configured MAC address.
  - Transit (forwarded) packets going out of the interface do not have the configured static MAC as source MAC address. In such a scenario, the upper 36-bits come from the system MAC address (or the original/dynamic MAC address) and the lower 12-bits come from the MAC address configured on the bundle. To check the dynamic pool of MAC addresses included, use the `show ethernet mac-allocation detail` command.

For example, if the dynamic MAC address was 008A.9624.48D8 and the configured static MAC address is 0011.2222.ABCD. Then, the source MAC for transit (forwarded) traffic will be 008A.9624.4BCD.




---

**Note** This limitation can cause traffic blackholing for the transit traffic, in case there is L2 ACL applied for security purpose. In such case, it is necessary to add permit statement for both MAC addresses in the L2 ACL.

---

- Load balancing (the distribution of data between member links) is done by flow instead of by packet. Data is distributed to a link in proportion to the bandwidth of the link in relation to its bundle.
- All links within a single bundle must terminate on the same two systems.
- Bundled interfaces are point-to-point.
- A link must be in the up state before it can be in distributing state in a bundle.
- Only physical links can be bundle members.
- Multicast traffic is load balanced over the members of a bundle. For a given flow, the internal processes selects the member link, and the traffic for the flow is sent over that member.

## Configuring Ethernet Link Bundles

This section describes how to configure an Ethernet link bundle.




---

**Note** In order for an Ethernet bundle to be active, you must perform the same configuration on both connection endpoints of the bundle.

---




---

**Tip** You can programmatically perform the configuration using `openconfig-if-aggregate.yang` OpenConfig data model. To get started with using data models, see the *Programmability Configuration Guide for Cisco NCS 5500 Series Routers*.

---

## SUMMARY STEPS

1. **configure**
2. **interface Bundle-Ether** *bundle-id*
3. **ipv4 address** *ipv4-address mask*
4. **bundle minimum-active bandwidth** *kbps*
5. **bundle minimum-active links** *links*
6. **bundle maximum-active links** *links* [**hot-standby**]
7. **exit**
8. **interface HundredGigE** *interface-path-id*
9. **bundle id** *bundle-id* [**mode** {**active** | **on** | **passive**}]
10. **bundle port-priority** *priority*
11. **no shutdown**
12. **exit**
13. **bundle id** *bundle-id* [**mode** {**active** | **passive** | **on**}] **no shutdown exit**
14. **end** or **commit**
15. **exit**
16. **exit**
17. Perform Step 1 through Step 15 on the remote end of the connection.
18. **show bundle Bundle-Ether** *bundle-id*
19. **show lacp Bundle-Ether** *bundle-id*

## DETAILED STEPS

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

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

Enters global configuration mode.

**Step 2** **interface Bundle-Ether** *bundle-id***Example:**

```
RP/0/RP0/CPU0:router(config)# interface Bundle-Ether 3
```

Creates a new Ethernet link bundle with the specified bundle-id. The range is 1 to 65535.

This **interface Bundle-Ether** command enters you into the interface configuration submenu, where you can enter interface specific configuration commands are entered. Use the **exit** command to exit from the interface configuration submenu back to the normal global configuration mode.

**Step 3** **ipv4 address** *ipv4-address mask***Example:**

```
RP/0/RP0/CPU0:router(config-if)# ipv4 address 10.1.2.3 255.0.0.0
```

Assigns an IP address and subnet mask to the virtual interface using the **ipv4 address** configuration subcommand.



**Note** • Only a Layer 3 bundle interface requires an IP address.

**Step 4** **bundle minimum-active bandwidth** *kbps*

**Example:**

```
RP/0/RP0/CPU0:router(config-if)# bundle minimum-active bandwidth 580000
```

(Optional) Sets the minimum amount of bandwidth required before a user can bring up a bundle.

**Step 5** **bundle minimum-active links** *links*

**Example:**

```
RP/0/RP0/CPU0:router(config-if)# bundle minimum-active links 2
```

(Optional) Sets the number of active links required before you can bring up a specific bundle.

**Step 6** **bundle maximum-active links** *links* [**hot-standby**]

**Example:**

```
RP/0/RP0/CPU0:router(config-if)# bundle maximum-active links 1 hot-standby
```

(Optional) Implements 1:1 link protection for the bundle, which causes the highest-priority link in the bundle to become active and the second-highest-priority link to become the standby. Also, specifies that a switchover between active and standby LACP-enabled links is implemented per a proprietary optimization.

**Note** • The priority of the active and standby links is based on the value of the **bundle port-priority** command.

**Step 7** **exit**

**Example:**

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

Exits interface configuration submode for the Ethernet link bundle.

**Step 8** **interface HundredGigE** *interface-path-id*

**Example:**

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

Enters interface configuration mode for the specified interface.

Enter the **HundredGigE** keyword to specify the interface type. Replace the *interface-path-id* argument with the node-id in the *rack/slot/module* format.

**Step 9** **bundle id** *bundle-id* [**mode** {**active** | **on** | **passive**}]

**Example:**

```
RP/0/RP0/CPU0:router(config-if)# bundle id 3 mode on
```

Adds the link to the specified bundle.

To enable active or passive LACP on the bundle, include the optional **mode active** or **mode passive** keywords in the command string.

To add the link to the bundle without LACP support, include the optional **mode on** keywords with the command string.

**Note** • If you do not specify the **mode** keyword, the default mode is **on** (LACP is not run over the port).

#### Step 10 **bundle port-priority** *priority*

##### **Example:**

```
RP/0/RP0/CPU0:router(config-if)# bundle port-priority 1
```

(Optional) If you set the **bundle maximum-active links** command to 1, you must also set the priority of the active link to the highest priority (lowest value) and the standby link to the second-highest priority (next lowest value). For example, you can set the priority of the active link to 1 and the standby link to 2.

#### Step 11 **no shutdown**

##### **Example:**

```
RP/0/RP0/CPU0:router(config-if)# no shutdown
```

(Optional) If a link is in the down state, bring it up. The **no shutdown** command returns the link to an up or down state depending on the configuration and state of the link.

#### Step 12 **exit**

##### **Example:**

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

Exits interface configuration submode for the Ethernet interface.

#### Step 13 **bundle id** *bundle-id* [**mode** {**active** | **passive** | **on**}] **no shutdown exit**

##### **Example:**

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

```
RP/0/RP0/CPU0:router(config-if)# bundle id 3
```

```
RP/0/RP0/CPU0:router(config-if)# bundle port-priority 2
```

```
RP/0/RP0/CPU0:router(config-if)# no shutdown
```

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

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

```
RP/0/RP0/CPU0:router(config-if)# bundle id 3
```

```
RP/0/RP0/CPU0:router(config-if)# no shutdown
```

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

(Optional) Repeat Step 8 through Step 11 to add more links to the bundle.

#### Step 14 **end** or **commit**

##### **Example:**

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

or

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

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

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

- 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.

**Step 15**     **exit**

**Example:**

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

Exits interface configuration mode.

**Step 16**     **exit**

**Example:**

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

Exits global configuration mode.

**Step 17**     Perform Step 1 through Step 15 on the remote end of the connection.

Brings up the other end of the link bundle.

**Step 18**     **show bundle Bundle-Ether** *bundle-id*

**Example:**

```
RP/0/RP0/CPU0:router# show bundle Bundle-Ether 3
```

(Optional) Shows information about the specified Ethernet link bundle.

**Step 19**     **show lacp Bundle-Ether** *bundle-id*

**Example:**

```
RP/0/RP0/CPU0:router# show lacp Bundle-Ether 3
```

(Optional) Shows detailed information about LACP ports and their peers.

# Configuring LACP Fallback

This section describes how to configure the LACP Fallback feature.

## SUMMARY STEPS

1. **configure**
2. **interface Bundle-Ether** *bundle-id*
3. **bundle lacp-fallback timeout** *timeout value*
4. **end** or **commit**
5. **show bundle infrastructure database ma bdl-info Bundle-e1010 | inc text**
6. **show bundle infrastructure database ma bdl-info Bundle-e1015 | inc text**

## DETAILED STEPS

### Step 1 **configure**

**Example:**

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

Enters global configuration mode.

### Step 2 **interface Bundle-Ether** *bundle-id*

**Example:**

```
RP/0/RP0/CPU0:router(config)# interface Bundle-Ether 3
```

Creates and names a new Ethernet link bundle.

The **interface Bundle-Ether** command enters into the interface configuration submenu, where you can enter interface-specific configuration commands. Use the **exit** command to exit from the interface configuration submenu back to the normal return to global configuration mode.

### Step 3 **bundle lacp-fallback timeout** *timeout value*

**Example:**

```
RP/0/RP0/CPU0:router(config-if)# bundle lacp-fallback timeout 4
```

Enables the LACP Fallback feature.

### Step 4 **end** or **commit**

**Example:**

```
RP/0/RP0/CPU0:router(config-subif)# commit
```

Saves configuration changes.

### Step 5 **show bundle infrastructure database ma bdl-info Bundle-e1010 | inc text**

**Example:**

```
RP/0/RP0/CPU0:router# show bundle infrastructure database ma bdl-info Bundle-e1010 | inc "fallback"
```

(Optional) Shows the MA information of the bundle manager.

**Step 6** `show bundle infrastructure database ma bdl-info Bundle-e1015 | inc text`

**Example:**

```
RP/0/RP0/CPU0:router# show bundle infrastructure database ma bdl-info Bundle-e1015 | inc "fallback"
```

(Optional) Shows the MA information of the bundle manager.

## VLANs on an Ethernet Link Bundle

802.1Q VLAN subinterfaces can be configured on 802.3ad Ethernet link bundles. Keep the following information in mind when adding VLANs on an Ethernet link bundle:

- There is no separate limit defined for Layer 3 sub-interfaces on a bundle. However, an overall system limit of 4000 is applicable for NCS5001 and NCS5002, while a limit of 2000 is applicable for NCS5011.




---

**Note** The memory requirement for bundle VLANs is slightly higher than standard physical interfaces.

---

To create a VLAN subinterface on a bundle, include the VLAN subinterface instance with the **interface Bundle-Ether** command, as follows:

```
interface Bundle-Ether interface-bundle-id.subinterface
```

After you create a VLAN on an Ethernet link bundle, all VLAN subinterface configuration is supported on that link bundle.

VLAN subinterfaces can support multiple Layer 2 frame types and services, such as Ethernet Flow Points (EFPs) and Layer 3 services.

Layer 2 EFPs are configured as follows:

```
interface bundle-ether instance.subinterface l2transport. encapsulation dot1q xxxxx
```

Layer 3 VLAN subinterfaces are configured as follows:

```
interface bundle-ether instance.subinterface, encapsulation dot1q xxxxx
```




---

**Note** The difference between the Layer 2 and Layer 3 interfaces is the **l2transport** keyword. Both types of interfaces use **dot1q encapsulation**.

---

# Configuring VLAN over Bundles

This section describes how to configure a VLAN bundle. The creation of a VLAN bundle involves three main tasks:

## SUMMARY STEPS

1. Create an Ethernet bundle.
2. Create VLAN subinterfaces and assign them to the Ethernet bundle.
3. Assign Ethernet links to the Ethernet bundle.

## DETAILED STEPS

- 
- |               |                                                                   |
|---------------|-------------------------------------------------------------------|
| <b>Step 1</b> | Create an Ethernet bundle.                                        |
| <b>Step 2</b> | Create VLAN subinterfaces and assign them to the Ethernet bundle. |
| <b>Step 3</b> | Assign Ethernet links to the Ethernet bundle.                     |
- 

These tasks are describe in detail in the procedure that follows.




---

**Note** In order for a VLAN bundle to be active, you must perform the same configuration on both ends of the bundle connection.

---

## SUMMARY STEPS

1. **configure**
2. **interface Bundle-Ether** *bundle-id*
3. **ipv4 address** *ipv4-address mask*
4. **bundle minimum-active bandwidth** *kbps*
5. **bundle minimum-active links** *links*
6. **bundle maximum-active links** *links* [**hot-standby**]
7. **exit**
8. **interface Bundle-Ether** *bundle-id.vlan-id*
9. **encapsulation dot1q***vlan-id*
10. **ipv4 address** *ipv4-address mask*
11. **no shutdown**
12. **exit**
13. Repeat Step 9 through Step 12 to add more VLANs to the bundle you created in Step 2.
14. **end** or **commit**
15. **exit**
16. **exit**
17. **configure**
18. **interface** {**TenGigE** | **FortyGigE** | **HundredGigE**}*interface-path-id*

## DETAILED STEPS

---

### Step 1 **configure**

#### Example:

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

Enters global configuration mode.

### Step 2 **interface Bundle-Ether *bundle-id***

#### Example:

```
RP/0/RP0/CPU0:router#(config)# interface Bundle-Ether 3
```

Creates and names a new Ethernet link bundle.

This **interface Bundle-Ether** command enters you into the interface configuration submode, where you can enter interface-specific configuration commands. Use the **exit** command to exit from the interface configuration submode back to the normal global configuration mode.

### Step 3 **ipv4 address *ipv4-address mask***

#### Example:

```
RP/0/RP0/CPU0:router(config-if)# ipv4 address 10.1.2.3 255.0.0.0
```

Assigns an IP address and subnet mask to the virtual interface using the **ipv4 address** configuration subcommand.

### Step 4 **bundle minimum-active bandwidth *kbps***

#### Example:

```
RP/0/RP0/CPU0:router(config-if)# bundle minimum-active bandwidth 580000
```

(Optional) Sets the minimum amount of bandwidth required before a user can bring up a bundle.

### Step 5 **bundle minimum-active links *links***

#### Example:

```
RP/0/RP0/CPU0:router(config-if)# bundle minimum-active links 2
```

(Optional) Sets the number of active links required before you can bring up a specific bundle.

### Step 6 **bundle maximum-active links *links* [hot-standby]**

#### Example:

```
RP/0/RP0/CPU0:router(config-if)# bundle maximum-active links 1 hot-standby
```

(Optional) Implements 1:1 link protection for the bundle, which causes the highest-priority link in the bundle to become active and the second-highest-priority link to become the standby. Also, specifies that a switchover between active and standby LACP-enabled links is implemented per a proprietary optimization.

**Note** The priority of the active and standby links is based on the value of the **bundle port-priority** command.

### Step 7 **exit**

**Example:**

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

Exits the interface configuration submode.

**Step 8** **interface Bundle-Ether** *bundle-id.vlan-id***Example:**

```
RP/0/RP0/CPU0:router#(config)# interface Bundle-Ether 3.1
```

Creates a new VLAN, and assigns the VLAN to the Ethernet bundle you created in Step 2.

Replace the *bundle-id* argument with the *bundle-id* you created in Step 2.

Replace the *vlan-id* with a subinterface identifier.

Range is from 1 to 4093 inclusive (0, 4094, and 4095 are reserved).

**Note** When you include the *.vlan-id* argument with the **interface Bundle-Ether** *bundle-id* command, you enter subinterface configuration mode.

**Step 9** **encapsulation dot1q***vlan-id***Example:**

```
RP/0/RP0/CPU0:router(config-subif)# encapsulation dot1q 100
```

Sets the Layer 2 encapsulation of an interface.

**Step 10** **ipv4 address** *ipv4-address mask***Example:**

```
RP/0/RP0/CPU0:router#(config-subif)# ipv4 address 10.1.2.3/24
```

Assigns an IP address and subnet mask to the subinterface.

**Step 11** **no shutdown****Example:**

```
RP/0/RP0/CPU0:router#(config-subif)# no shutdown
```

(Optional) If a link is in the down state, bring it up. The **no shutdown** command returns the link to an up or down state depending on the configuration and state of the link.

**Step 12** **exit****Example:**

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

Exits subinterface configuration mode for the VLAN subinterface.

**Step 13** Repeat Step 9 through Step 12 to add more VLANs to the bundle you created in Step 2.

(Optional) Adds more subinterfaces to the bundle.

**Step 14** **end** or **commit**



**Example:**

```
RP/0/RP0/CPU0:router(config-subif)# end
```

or

```
RP/0/RP0/CPU0:router(config-subif)# commit
```

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

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

- 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.

**Step 15**    **exit****Example:**

```
RP/0/RP0/CPU0:router(config-subif)# end
```

Exits interface configuration mode.

**Step 16**    **exit****Example:**

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

Exits global configuration mode.

**Step 17**    **configure****Example:**

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

Enters global configuration mode.

**Step 18**    **interface {TenGigE | FortyGigE | HundredGigE} interface-path-id****Example:**

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

Enters interface configuration mode for the Ethernet interface you want to add to the Bundle.

Enter the **GigabitEthernet** or **TenGigE** keyword to specify the interface type. Replace the *interface-path-id* argument with the node-id in the rack/slot/module format.

**Note** A VLAN bundle is not active until you add an Ethernet interface on both ends of the link bundle.

---

## LACP Short Period Time Intervals

As packets are exchanged across member links of a bundled interface, some member links may slow down or time-out and fail. LACP packets are exchanged periodically across these links to verify the stability and reliability of the links over which they pass. The configuration of short period time intervals, in which LACP packets are sent, enables faster detection and recovery from link failures.

Short period time intervals are configured as follows:

- In milliseconds
- In increments of 100 milliseconds
- In the range 100 to 1000 milliseconds
- The default is 1000 milliseconds (1 second)
- Up to 64 member links
- Up to 1280 packets per second (pps)

After 6 missed packets, the link is detached from the bundle.

When the short period time interval is *not* configured, LACP packets are transmitted over a member link every 30 seconds by default.

When the short period time interval is configured, LACP packets are transmitted over a member link once every 1000 milliseconds (1 second) by default. Optionally, both the transmit and receive intervals can be configured to less than 1000 milliseconds, independently or together, in increments of 100 milliseconds (100, 200, 300, and so on).

When you configure a custom LACP short period *transmit* interval at one end of a link, you must configure the same time period for the *receive* interval at the other end of the link.



---

**Note** You must always configure the *transmit* interval at both ends of the connection before you configure the *receive* interval at either end of the connection. Failure to configure the *transmit* interval at both ends first results in route flapping (a route going up and down continuously). When you remove a custom LACP short period, you must do it in reverse order. You must remove the *receive* intervals first and then the *transmit* intervals.

---

## Configuring the Default LACP Short Period Time Interval

This section describes how to configure the default short period time interval for sending and receiving LACP packets on a Gigabit Ethernet interface. This procedure also enables the LACP short period.

## SUMMARY STEPS

1. **configure**
2. **interface HundredGigE***interface-path*
3. **bundle id** *number* **mode active**
4. **lacp period short**
5. **end** or **commit**

## DETAILED STEPS

### Step 1 **configure**

**Example:**

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

Enters global configuration mode.

### Step 2 **interface HundredGigE***interface-path*

**Example:**

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

Creates a Gigabit Ethernet interface and enters interface configuration mode.

### Step 3 **bundle id** *number* **mode active**

**Example:**

```
RP/0/RP0/CPU0:router(config-if)# bundle id 1 mode active
```

Specifies the bundle interface and puts the member interface in active mode.

### Step 4 **lacp period short**

**Example:**

```
RP/0/RP0/CPU0:router(config-if)# lacp period short
```

Configures a short period time interval for the sending and receiving of LACP packets, using the default time period of 1000 milliseconds or 1 second.

### Step 5 **end** or **commit**

**Example:**

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

or

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

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

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

- 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.

### Example

This example shows how to configure the LACP short period time interval to the default time of 1000 milliseconds (1 second):

```
config
interface HundredGigE 0/1/0/1
 bundle id 1 mode active
 lacp period short
 commit
```

The following example shows how to configure custom LACP short period transmit and receive intervals to *less than* the default of 1000 milliseconds (1 second):

```
config
interface HundredGigE 0/1/0/1
 bundle id 1 mode active
 lacp period short
 commit

config
interface HundredGigE 0/1/0/1
 lacp period short transmit 100
 commit

config
interface HundredGigE 0/1/0/1
 lacp period short receive 100
 commit
```

## Configuring Custom LACP Short Period Time Intervals

This section describes how to configure custom short period time intervals (less than 1000 milliseconds) for sending and receiving LACP packets on a Gigabit Ethernet interface.



**Note** You must always configure the *transmit* interval at both ends of the connection before you configure the *receive* interval at either end of the connection. Failure to configure the *transmit* interval at both ends first results in route flapping (a route going up and down continuously). When you remove a custom LACP short period, you must do it in reverse order. You must remove the *receive* intervals first and then the *transmit* intervals.

## SUMMARY STEPS

1. **configure**
2. **interface Bundle-Ether** *bundle-id*
3. **ipv4 address** *ipv4-address mask*
4. **bundle minimum-active bandwidth** *kbps*
5. **bundle minimum-active links** *links*
6. **bundle maximum-active links** *links*
7. **exit**
8. **interface Bundle-Ether** *bundle-id.vlan-id*
9. **dot1q vlan** *vlan-id*
10. **ipv4 address** *ipv4-address mask*
11. **no shutdown**
12. **exit**
13. Repeat Step 7 through Step 12 to add more VLANs to the bundle you created in Step 2.
14. **end** or **commit**
15. **exit**
16. **exit**
17. **show ethernet trunk bundle-ether** *instance*
18. **configure**
19. **interface {HundredGigE }** *interface-path-id*
20. **bundle id** *bundle-id* [**mode** {**active** | **on** | **passive**}]
21. **no shutdown**
22. Repeat Step 19 through Step 21 to add more Ethernet interfaces to the VLAN bundle.
23. **end** or **commit**
24. Perform Step 1 through Step 23 on the remote end of the VLAN bundle connection.
25. **show bundle Bundle-Ether** *bundle-id* [**reasons**]
26. **show ethernet trunk bundle-ether** *instance*

## DETAILED STEPS

**Step 1** **configure**

**Example:**

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

Enters global configuration mode.

**Step 2** **interface Bundle-Ether** *bundle-id***Example:**

```
RP/0/RP0/CPU0:router(config)# interface Bundle-Ether 3
```

Creates and names a new Ethernet link bundle.

This **interface Bundle-Ether** command enters you into the interface configuration submenu, where you can enter interface-specific configuration commands. Use the **exit** command to exit from the interface configuration submenu back to the normal global configuration mode.

**Step 3** **ipv4 address** *ipv4-address mask***Example:**

```
RP/0/RP0/CPU0:router(config-if)# ipv4 address 10.1.2.3 255.0.0.0
```

Assigns an IP address and subnet mask to the virtual interface using the **ipv4 address** configuration subcommand.

**Step 4** **bundle minimum-active bandwidth** *kbps***Example:**

```
RP/0/RP0/CPU0:router(config-if)# bundle minimum-active bandwidth 580000
```

(Optional) Sets the minimum amount of bandwidth required before a user can bring up a bundle.

**Step 5** **bundle minimum-active links** *links***Example:**

```
RP/0/RP0/CPU0:router(config-if)# bundle minimum-active links 2
```

(Optional) Sets the number of active links required before you can bring up a specific bundle.

**Step 6** **bundle maximum-active links** *links***Example:**

```
RP/0/RP0/CPU0:router(config-if)# bundle maximum-active links 1
```

(Optional) Designates one active link and one link in standby mode that can take over immediately for a bundle if the active link fails (1:1 protection).

- Note**
- The default number of active links allowed in a single bundle is 8.
  - If the **bundle maximum-active** command is issued, then only the highest-priority link within the bundle is active. The priority is based on the value from the **bundle port-priority** command, where a lower value is a higher priority. Therefore, we recommend that you configure a higher priority on the link that you want to be the active link.

**Step 7** **exit****Example:**

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

Exits the interface configuration submenu.

**Step 8**      **interface Bundle-Ether** *bundle-id.vlan-id***Example:**

```
RP/0/RP0/CPU0:router#(config)# interface Bundle-Ether 3.1
```

Creates a new VLAN, and assigns the VLAN to the Ethernet bundle you created in Step 2.

Replace the *bundle-id* argument with the *bundle-id* you created in Step 2.

Replace the *vlan-id* with a subinterface identifier. Range is from 1 to 4093 inclusive (0, 4094, and 4095 are reserved).

**Note**      • When you include the *vlan-id* argument with the **interface Bundle-Ether** *bundle-id* command, you enter subinterface configuration mode.

**Step 9**      **dot1q vlan** *vlan-id***Example:**

```
RP/0/RP0/CPU0:router(config-subif)# dot1q vlan 10
```

Assigns a VLAN to the subinterface.

Replace the *vlan-id* argument with a subinterface identifier. Range is from 1 to 4093 inclusive (0, 4094, and 4095 are reserved).

**Step 10**     **ipv4 address** *ipv4-address mask***Example:**

```
RP/0/RP0/CPU0:router(config-subif)# ipv4 address 10.1.2.3/24
```

Assigns an IP address and subnet mask to the subinterface.

**Step 11**     **no shutdown****Example:**

```
RP/0/RP0/CPU0:router(config-subif)# no shutdown
```

(Optional) If a link is in the down state, bring it up. The **no shutdown** command returns the link to an up or down state depending on the configuration and state of the link.

**Step 12**     **exit****Example:**

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

Exits subinterface configuration mode for the VLAN subinterface.

**Step 13**     Repeat Step 7 through Step 12 to add more VLANs to the bundle you created in Step 2.

(Optional) Adds more subinterfaces to the bundle.

**Step 14**     **end** or **commit****Example:**

```
RP/0/RP0/CPU0:router(config-subif)# end
```

or

```
RP/0/RP0/CPU0:router(config-subif)# commit
```

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes: `Uncommitted changes found, commit them before exiting (yes/no/cancel)?`
- 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.

#### Step 15 **exit**

##### Example:

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

Exits interface configuration mode.

#### Step 16 **exit**

##### Example:

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

Exits global configuration mode.

#### Step 17 **show ethernet trunk bundle-ether** *instance*

##### Example:

```
RP/0/RP0/CPU0:router# show ethernet trunk bundle-ether 5
```

(Optional) Displays the interface configuration.

The Ethernet bundle instance range is from 1 through 65535.

#### Step 18 **configure**

##### Example:

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

Enters global configuration mode.

#### Step 19 **interface {HundredGigE } interface-path-id**

##### Example:

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



Enters the interface configuration mode for the Ethernet interface you want to add to the Bundle.

Enter the **HundredGigE** keyword to specify the interface type. Replace the *interface-path-id* argument with the node-id in the rack/slot/module format.

**Note** • A VLAN bundle is not active until you add an Ethernet interface on both ends of the link bundle.

**Step 20** **bundle id** *bundle-id* [**mode** {**active** | **on** | **passive**}]

**Example:**

```
RP/0/RP0/CPU0:router(config-if)# bundle-id 3
```

Adds an Ethernet interface to the bundle you configured in Step 2 through Step 13.

To enable active or passive LACP on the bundle, include the optional **mode active** or **mode passive** keywords in the command string.

To add the interface to the bundle without LACP support, include the optional **mode on** keywords with the command string.

**Note** • If you do not specify the **mode** keyword, the default mode is **on** (LACP is not run over the port).

**Step 21** **no shutdown**

**Example:**

```
RP/0/RP0/CPU0:router(config-if)# no shutdown
```

(Optional) If a link is in the down state, bring it up. The **no shutdown** command returns the link to an up or down state depending on the configuration and state of the link.

**Step 22** Repeat Step 19 through Step 21 to add more Ethernet interfaces to the VLAN bundle.

—

**Step 23** **end** or **commit**

**Example:**

```
RP/0/RP0/CPU0:router(config-subif)# end
```

OR

```
RP/0/RP0/CPU0:router(config-subif)# commit
```

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes: `Uncommitted changes found, commit them before exiting (yes/no/cancel)?`
- 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.

**Step 24** Perform Step 1 through Step 23 on the remote end of the VLAN bundle connection.

Brings up the other end of the link bundle.

**Step 25** **show bundle Bundle-Ether** *bundle-id* [reasons]

**Example:**

```
RP/0/RP0/CPU0:router# show bundle Bundle-Ether 3 reasons
```

(Optional) Shows information about the specified Ethernet link bundle.

The **show bundle Bundle-Ether** command displays information about the specified bundle. If your bundle has been configured properly and is carrying traffic, the State field in the **show bundle Bundle-Ether** command output will show the number “4,” which means the specified VLAN bundle port is “distributing.”

**Step 26** **show ethernet trunk bundle-ether** *instance*

**Example:**

```
RP/0/RP0/CPU0:router# show ethernet trunk bundle-ether 5
```

(Optional) Displays the interface configuration.

The Ethernet bundle instance range is from 1 through 65535.

## Bundle Consistency Checker

*Table 11: Feature History Table*

| Feature Name                     | Release Information | Feature Description                                                                                                                                                                                                                                                                                                                                                                              |
|----------------------------------|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Bundle Consistency Checker (BCC) | Release 7.3.1       | From the running configuration, Bundle Consistency Checker (BCC) fetches information about the ingress/egress traffic from the bundle, sub-bundle, and active member nodes and saves it in the database. BCC also collects data from all the running nodes and then compares it with the information saved in the database. Any inconsistencies, programming errors, stale entries are reported. |

In a scaled setup, a bundle programming check is difficult to perform and time consuming. Moreover, an issue is reported only when the user detects it, and not automatically. During multiple test executions, it isn't possible to detect the initial failure, which causes other subsequent failures. Bundle Consistency Checker (BCC) implements bundle programming and consistency check by using the following steps:

1. BCC uses the running configuration to detect discrepancies.

2. BCC forms a Bundle Consistency Checker Data Base (BCCDB) with the bundle, sub-bundle, or member information fetched from the running configuration.
3. BCC dumps the required data from all available nodes. It then uses BCCDB as a source to verify bundle programming and consistency in all other layer dumps.
4. BCC reports inconsistencies, programming errors, stale entries, and deletes any pending objects.

**Supporting Interfaces**

The following interfaces support BCC:

- Bundle
- Bundle sub-interface

The following table lists BCC behaviour during inconsistencies in bundle configuration or programming errors.

| Case                         | BCC Behaviour                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| When no bundle is configured | <pre> Router# <b>show bundle consistency</b> Building configuration... Dumping Data..... Done Parsing Data..... Not Done  BCC Stopped: Found 3 info/exceptions/errors  Logs Preview:  2020-07-13 10:34:22,774: INFO: Bundlemgr PD dont have any bundle data  2020-07-13 10:34:22,832: INFO: BMPI dont have any bundle data  2020-07-13 10:34:23,728: INFO: No Bundle is configured/No member is added to Bundle  Logs: /var/log/bcc_exception.log /var/log/bcc_debug.log                     </pre> |

| Case                                               | BCC Behaviour                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|----------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| When a bundle is configured but no member is added | <pre> Router# show bundle consistency Building configuration... Dumping Data..... Done Parsing Data..... Not Done  BCC Stopped: Found 4 info/exceptions/errors  Logs Preview:  2020-07-13 10:36:32,513: INFO: Bundlemgr PD dont have any bundle data  2020-07-13 10:36:32,566: INFO: No member is added to bundle BE1(0x3c00400c)  2020-07-13 10:36:32,566: INFO: BMPI dont have any bundle data  2020-07-13 10:36:33,453: INFO: No Bundle is configured/No member is added to Bundle  Logs: /var/log/bcc_exception.log /var/log/bcc_debug.log </pre>                                           |
| When a bundle is configured and members are added  | <pre> Router# show bundle consistency Building configuration... Dumping Data..... Done Parsing Data..... Done Bundle Consistency Check..... Done Bundle Programming Check..... Done Stale Entry Check..... Done Bundle Health Check..... Done  Overall Results:  Inconsistencies           : 0           Stale Entries       : 0 BCM Programming Error      : 0           Delete Pending DPA Objects      : 0 Info/Error/Python Exception : 0           Overall Bundle Health Status : WARNING  Execute 'show bundle status' to see detailed reason for 'WARNING' in bundle health check </pre> |

| Case                                                                                                   | BCC Behaviour                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|--------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>When there is no encapsulation configuration for L2 or L3 sub-bundle or no member for L2 bundle</p> | <pre> Router# show bundle consistency Building configuration... Dumping Data..... Done Parsing Data..... Done Bundle Consistency Check..... Done Bundle Programming Check..... Done Stale Entry Check..... Done Bundle Health Check..... Done  Overall Results:  Inconsistencies          : 0          Stale Entries       : 0 BCM Programming Error     : 0          Delete Pending DPA Objects      : 0 Info/Error/Python Exception : 3          Overall Bundle Health       Status : WARNING  Execute 'show bundle status' to see detailed reason for 'WARNING' in bundle health check  Logs Preview:  2020-07-12 17:38:26,568: INFO: No member is added to bundle BE2(0x80042bc) ==&gt; 12 bundle main  2020-07-12 17:38:32,573: interface Bundle-Ether1.1: Dont have any encapsulation config. ==&gt; 13 sub  2020-07-12 17:38:32,574: interface Bundle-Ether1.130: Dont have any encapsulation config. ==&gt; 12sub  Logs:  /var/log/bcc_inconsistencies.log /var/log/bcc_programming_error.log /var/log/bcc_stale_entries.log /var/log/bcc_delay_delete.log /var/log/bcc_bundle_health.log /var/log/bcc_exception.log /var/log/bcc_debug.log                     </pre> |

| Case                      | BCC Behaviour                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|---------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| During programming errors | <pre> Router# show bundle consistency  Building configuration... Dumping Data..... Done Parsing Data..... Done Bundle Consistency Check..... Done Bundle Programming Check..... Done Stale Entry Check..... Done Bundle Health Check..... Done  Overall Results:  Inconsistencies           : 0           Stale Entries       : 0 BCM Programming Error     : 1           Delete Pending DPA Objects      : 0 Info/Error/Python Exception : 0           Overall Bundle Health Status      : WARNING  Execute 'show bundle status' to see detailed reason for 'WARNING' in bundle health check  Logs Preview:  2020-07-12 18:48:22,658: Programming Error 1: BE1(0x80041ec) NPU 0,0/RP0/CPU0 Vlan Domain 0x33 != GigabitEthernet0_0_0_2 Vlan Domain 0xa  Logs:  /var/log/bcc_inconsistencies.log /var/log/bcc_programming_error.log /var/log/bcc_stale_entries.log /var/log/bcc_delay_delete.log /var/log/bcc_bundle_health.log /var/log/bcc_exception.log /var/log/bcc_debug.log </pre> |

## Information About Configuring Link Bundling

To configure link bundling, you must understand the following concepts:

### IEEE 802.3ad Standard

The IEEE 802.3ad standard typically defines a method of forming Ethernet link bundles.

For each link configured as bundle member, the following information is exchanged between the systems that host each end of the link bundle:

- A globally unique local system identifier
- An identifier (operational key) for the bundle of which the link is a member
- An identifier (port ID) for the link
- The current aggregation status of the link

This information is used to form the link aggregation group identifier (LAG ID). Links that share a common LAG ID can be aggregated. Individual links have unique LAG IDs.

The system identifier distinguishes one router from another, and its uniqueness is guaranteed through the use of a MAC address from the system. The bundle and link identifiers have significance only to the router assigning them, which must guarantee that no two links have the same identifier, and that no two bundles have the same identifier.

The information from the peer system is combined with the information from the local system to determine the compatibility of the links configured to be members of a bundle.

The MAC address of the first link attached to a bundle becomes the MAC address of the bundle itself. The bundle uses this MAC address until that link (the first link attached to the bundle) is detached from the bundle, or until the user configures a different MAC address. The bundle MAC address is used by all member links when passing bundle traffic. Any unicast or multicast addresses set on the bundle are also set on all the member links.



---

**Note** We recommend that you avoid modifying the MAC address, because changes in the MAC address can affect packet forwarding.

---

## Link Bundle Configuration Overview

The following steps provide a general overview of the link bundle configuration. Keep in mind that a link must be cleared of all previous network layer configuration before it can be added to a bundle:

1. In global configuration mode, create a link bundle. To create an Ethernet link bundle, enter the **interface Bundle-Ether** command.
2. Assign an IP address and subnet mask to the virtual interface using the **ipv4 address** command.
3. Add interfaces to the bundle you created in Step 1 with the **bundle id** command in the interface configuration submode.

You can add up to 32 links to a single bundle.

4. You can optionally implement 1:1 link protection for the bundle by setting the **bundle maximum-active links** command to 1. Performing this configuration causes the highest-priority link in the bundle to become active and the second-highest-priority link to become the standby. (The link priority is based on the value of the **bundle port-priority** command.) If the active link fails, the standby link immediately becomes the active link.



---

**Note** A link is configured as a member of a bundle from the interface configuration submode for that link.

---

## Link Switchover

By default, a maximum of 64 links in a bundle can actively carry traffic. If one member link in a bundle fails, traffic is redirected to the remaining operational member links.

You can optionally implement 1:1 link protection for a bundle by setting the **bundle maximum-active links** command to 1. By doing so, you designate one active link and one or more dedicated standby links. If the active link fails, a switchover occurs and a standby link immediately becomes active, thereby ensuring uninterrupted traffic.

If the active and standby links are running LACP, you can choose between an IEEE standard-based switchover (the default) or a faster proprietary optimized switchover. If the active and standby links are not running LACP, the proprietary optimized switchover option is used.

Regardless of the type of switchover you are using, you can disable the wait-while timer, which expedites the state negotiations of the standby link and causes a faster switchover from a failed active link to the standby link.

To do so, you can use the **lacp fast-switchover** command.

## LACP Fallback

The LACP Fallback feature allows an active LACP interface to establish a Link Aggregation Group (LAG) port-channel before the port-channel receives the Link Aggregation and Control Protocol (LACP) protocol data units (PDU) from its peer. With the LACP Fallback feature configured, the router allows the server to bring up the LAG, before receiving any LACP PDUs from the server, and keeps one port active. This allows the server to establish a connection to PXE server over one Ethernet port, download its boot image and then continue the booting process. When the server boot process is complete, the server fully forms an LACP port-channel.





## CHAPTER 8

# Configuring Traffic Mirroring

This module describes the configuration of the traffic mirroring feature. Traffic mirroring is sometimes called port mirroring, or switched port analyzer (SPAN). You can then pass this traffic to a destination port on the same router.

### Feature Release History

| Release       | Modification                                                                                                                                                              |
|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Release 7.0.2 | SPAN over Pseudo-Wire was introduced.                                                                                                                                     |
| Release 7.1.2 | SPAN to File was introduced.                                                                                                                                              |
| Release 7.2.1 | <ul style="list-style-type: none"><li>• File Mirroring was introduced.</li><li>• Traffic Mirroring was introduced on Cisco NC57 line cards in native mode only.</li></ul> |
| Release 7.3.1 | PCAPng file format was introduced.                                                                                                                                        |
| Release 7.4.1 | Port Mirroring Enhancements for Cisco NC57 line cards were introduced.                                                                                                    |
| Release 7.5.2 | Mirror first option in global configuration mode was introduced.                                                                                                          |
| Release 7.5.3 | ERSPAN Traffic to a Destination Tunnel in a Non-Default VRF was introduced.                                                                                               |

| Release       | Modification                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Release 7.5.4 | <ul style="list-style-type: none"> <li>• Multiple SPAN ACL Sessions in a Single Interface was introduced.</li> <li>• Monitor Multiple SPAN ACL and Security ACL Sessions was introduced.</li> <li>• SPAN Using 7-Tuples ACL was introduced.</li> <li>• DSCP Marking on Egress GRE Tunnel in ERSPAN was introduced.</li> <li>• DSCP Bitmask to filter Ingress SPAN was introduced.</li> <li>• Mirroring Forward-Drop Packets was introduced.</li> </ul> |

- [Introduction to Traffic Mirroring, on page 194](#)
- [SPAN Types, Supported Features, and Configurations, on page 200](#)
- [Troubleshoot Traffic Mirroring, on page 229](#)

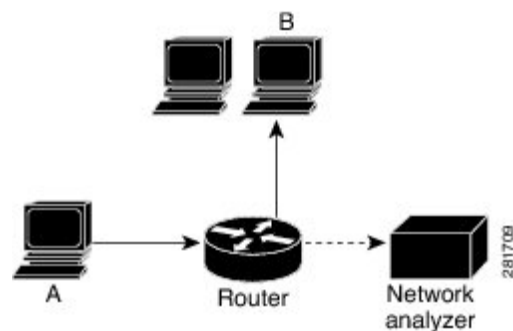
## Introduction to Traffic Mirroring

Traffic mirroring, also referred to as Port mirroring or Switched Port Analyzer (SPAN), is a Cisco proprietary feature that enables you to monitor network traffic passing in or out of a set of ports on a router. You can then mirror this traffic to a remote destination or a destination port on the same router.

Traffic mirroring copies traffic from one or more source ports and sends the copied traffic to one or more destinations for analysis by a network analyzer or other monitoring devices. Traffic mirroring does not affect the flow of traffic on the source interfaces or sub-interfaces. It allows the mirrored traffic to be sent to a destination interface or sub-interface.

For example, you can attach a traffic or network analyzer to the router and capture the ethernet traffic that is sent by host A to host B.

**Figure 14: Traffic Mirroring Operation**



## Traffic Mirroring Terminology

- Ingress Traffic — Traffic that comes into the router.

- Egress Traffic — Traffic that goes out of the router.
- Source port—A port that is monitored with the use of traffic mirroring. It is also called a monitored port.
- Destination port—A port that monitors source ports, usually where a network analyzer is connected. It is also called a monitoring port.
- Monitor session—A designation for a collection of SPAN configurations consisting of a single destination and, potentially, one or many source ports.

## Traffic Mirroring Types

The following types of traffic mirroring are supported:

- [Local SPAN](#)
- [SPAN on Subinterfaces](#)
- [ACL-based SPAN](#)
- [ERSPAN](#)
- [SPAN over Pseudowire](#)
- [SPAN-to-File, on page 223](#)
- [File Mirroring](#)

## Characteristics of Source Port

A source port, also called a monitored port, is a routed port that you monitor for network traffic analysis. In a single traffic mirroring session, you can monitor source port traffic. The Cisco NCS 5500 Series routers support a maximum of up to 800 source ports.

A source port has these characteristics:

- It can be any data port type, such as Bundle Interface, 100 Gigabit Ethernet physical port, or 10 Gigabit Ethernet physical port.
- Each source port can be monitored in only one traffic mirroring session.
- When a port is used as a source port, the same port cannot be used as a destination port.
- Each source port can be configured with a direction (ingress, egress, or both) to monitor local traffic mirroring. Remote traffic mirroring is supported both in the ingress and egress directions. For bundles, the monitored direction applies to all physical ports in the group.

## Characteristics of Destination Port

Each session must have a destination port or file that receives a copy of the traffic from the source ports.

A destination port has these characteristics:

- A destination port cannot be a source port.

- A destination port must reside on the same router as the source port for local traffic mirroring. For remote mirroring, the destination is always a GRE tunnel.
- For remote mirroring, the destination is a GRE tunnel.  
From Release 7.4.1, the destination can be an L2 sub-interface on NC57 line cards.
- A destination port for local mirroring can be any Ethernet physical port, EFP, GRE tunnel interface, or bundle interface. It can be a Layer 2 or Layer 3 transport interface.
- At any time, a destination port can participate in only one traffic mirroring session. A destination port in one traffic mirroring session cannot be a destination port for a second traffic mirroring session. In other words, no two monitor sessions can have the same destination port.

## Characteristics of Monitor Session

A monitor session is a collection of traffic mirroring configurations consisting of a single destination and, potentially, many source interfaces. For any given monitor session, the traffic from the source interfaces (called *source ports*) is sent to the monitoring port or destination port. If there are more than one source port in a monitoring session, the traffic from the several mirrored traffic streams is combined at the destination port. The result is that the traffic that comes out of the destination port is a combination of the traffic from one or more source ports.

Monitor sessions have these characteristics:

- A single monitor session can have only one destination port.
- A single destination port can belong to only one monitor session.
- A monitor session can have a maximum of 800 source ports. This maximum limit is applicable only when the maximum number of source ports from all monitoring sessions does not exceed 800.

## Supported Scale

- From Cisco IOS XR Software Release 7.2.1 to 7.3.1, Cisco NC57 line cards support only four Rx and three Tx monitor sessions in native mode. From 7.4.1 release, 24 sessions in total are supported in native mode. Sessions can be configured as Rx-only, Tx-only, or Rx/Tx.

## Restrictions

### Generic Restrictions

The following are the generic restrictions related to traffic mirroring:

- Partial mirroring and sampled mirroring are not supported.
- Sub-interface configured as source interface is not supported on SPAN.
- The destination bundle interfaces flap when:
  - both the mirror source and destination are bundle interfaces in the Link Aggregation Control Protocol (LACP) mode.

- mirror packets next-hop is a router or a switch instead of a traffic analyzer.

This behavior is observed due to a mismatch of LACP packets on the next-hop bundle interface due to the mirroring of LACP packets on the source bundle interface.

- Bridge group virtual interfaces (BVI) are not supported as source ports or destination ports.
- Bundle members cannot be used as source ports in NC57 line cards.
- Bundle members cannot be used as destination ports.
- Fragmentation of mirror copies is not handled by SPAN when SPAN destination MTU is less than the packet size. Existing behaviour if the MTU of destination interface is less than the packet size is as below:

| Platforms | Rx SPAN                                                                    | Tx SPAN                       |
|-----------|----------------------------------------------------------------------------|-------------------------------|
| NCS 5500  | Mirror copies are not fragmented. Receives whole packets as mirror copies. | Mirror copies are fragmented. |
| NCS 5700  | Mirror copies are not fragmented. Do not receive mirror copies.            | Mirror copies are fragmented. |

You can configure the SPAN destination with an MTU which is greater than the packet size.

- SPAN only supports port-level source interfaces.

### ACL-based SPAN Restrictions

The following restrictions apply to SPAN-ACL:

- *Table 12: SPAN-ACL Support*

| Platforms | Rx Direction                                                                              | Tx Direction   |
|-----------|-------------------------------------------------------------------------------------------|----------------|
| NCS 5500  | Supported at the port level, that is, in the ingress direction for IPv4 or IPv6 ACLs.     | Not supported. |
| NCS 5700  | Supported on both the main interfaces and sub-interfaces from Cisco IOS XR Release 7.4.1. |                |

- Multi-SPAN ACL is supported in the Rx direction in Cisco IOS XR Release 7.5.4 .
- Multi-SPAN ACL sessions can be used only with [7-Tuples SPAN ACL](#).
- MPLS traffic cannot be captured with SPAN-ACL.
  - ACL for any MPLS traffic is not supported.
- Traffic mirroring counters are not supported.
- ACL-based traffic mirroring is not supported with Layer 2 (ethernet-services) ACLs.

- Main interface as span source interface and ACL with the **capture** keyword on same main interface's sub-interface are not supported.
- If a SPAN session with the **acl** keyword is applied on an interface with no ACL rule attached to that interface, SPAN happens without any filtering.
- Configure one or more ACLs on the source interface or any interface on the same network processing unit as the source interface, to avoid default mirroring of traffic. If a Bundle interface is a source interface, configure the ACL on any interface on the same network processing unit as all active bundle-members. Bundle members can be on multiple NPUs. Also, ensure that the ACLs configured are of the same protocol type and direction as the SPAN configuration. For example, if you configure SPAN with ACL for IPv4 or IPv6, configure an ingress IPv4 or IPv6 ACL on that network processing unit respectively.

### ACL-based SPAN for Outgoing Traffic (Tx) Restrictions

The following restrictions apply to traffic mirroring using ACLs for outgoing (Tx) traffic on Cisco NCS 5700 Series line cards and routers:

- SPAN configuration with **port mode** on the main interface and Tx SPAN ACL configuration on the sub-interface of the same port isn't supported.
- BVI interface as a SPAN source interface is not supported.
- Hybrid ACLs with only compress level 3 are supported.
- 24 SPAN sessions are supported for both Rx and Tx destinations.
- ACL-based traffic mirroring for the outgoing (Tx) traffic is supported on the following routers and line cards for L3 interfaces:
  - NCS-57B1-5DSE
  - NCS-57C3-MODS-SYS
  - NC57-18DD-SE
  - NC57-36H-SE

### ERSPAN Restrictions

This section provides the restrictions that apply to ERSPAN and multiple ERSPAN sessions.

The following restrictions apply to ERSPAN:

- The value of ERSPAN session-ID is always zero. IOS XR command for configuring ERSPAN is not available.
- ERSPAN next-hop must have ARP resolved. Any other traffic or protocol will trigger ARP.
- ERSPAN packets with outgoing interface having MPLS encapsulation are not supported.
  - Additional routers may encapsulate in MPLS.
- ERSPAN sessions can be created only on physical interfaces. The sessions cannot be created on sub-interfaces.
- ERSPAN decapsulation is not supported.

- ERSPAN does not work if the GRE next hop is reachable over sub-interface. For ERSPAN to work, the next hop must be reachable over the main interface.
- When you use the same ACEs defined in both the IPv4 and IPv6 ACLs, the router doesn't perform ERSPAN mirroring for the ACLs that have the priority set as 2 ms.
- ERSPAN decapsulation is not supported. Tunnel destination should be network analyzer.

### Multiple ERSPAN Restrictions

- All sessions under the source port should have SPAN access control list (ACL) enabled.
- A few sessions with SPAN ACL and a few without SPAN ACLs in the same source interface are not supported.
- No two sessions should have the same ACL in the same source interface. Each session should have a different ACL.
- Multiple sessions without ACL in the same interface are not supported.
- Multi-SPAN ACL does not support the **Deny** action.
- One SPAN session with the keyword ACL (use security acl as the keyword) and other SPAN sessions with the keyword SPAN ACL are not supported.
- At a time, you can make only one mirror copy of a packet.
- Capturing keywords is not required.
- Multiple sessions under the same interface cannot have a combination of directions. Only RX is supported.

### SPAN over Pseudowire Restrictions

SPAN over Pseudowire (PW-SPAN) does not support the listed functionalities:

- Monitor session statistics
- Partial packet SPAN
- Sampled SPAN

### SPAN-to-File Restrictions

SPAN to File has the following restrictions:

- A maximum of 1000 source ports are supported across the system. Individual platforms may support lower numbers. The SPAN session may be any of these currently supported classes: Ethernet, IPv4, IPv6, MPLS-IPv4, and MPLS-IPv6.
- Provides a buffer range of 1000-1000000 KB. The default buffer size is set to 1000 KB.
- Provides support for SPAN source.
  - Each source port can be monitored in only one traffic mirroring session.
  - Each source port can be configured with a direction (ingress, egress, or both) to monitor local traffic mirroring.

- Only supported on the Cisco NCS550x and Cisco NCS55Ax line cards.
- Only port-level is supported.
- VLAN interface as source port is not supported.
- Bundle members as source interfaces are not supported.
- Filtering based on Egress ACL is not supported.
- Source port statistics is not supported.
- Not supported on Cisco NC57 line cards.
- Span to file mirror packets are punted from NPU to CPU at a maximum shaper rate of 40 mbps.

### File Mirroring Restrictions

The following restrictions apply to file mirroring:

- Supported only on Dual RP systems.
- Supports syncing only from active to standby RP. If files are copied into standby `/harddisk:/mirror` location, it won't be synced to active RP.
- A slight delay is observed in `show mirror` command output when mirror checksum configuration is enabled.
- Not supported on multichassis systems.

### Forward-Drop Packets Mirroring Restrictions

These are some restrictions for Forward-Drop packets mirroring:

- Only one global forward-drop session can be configured on a router.
- When traffic-class is configured under monitor-session for forward-drop, the type of service (ToS) byte of the outgoing ERSPAN packet is overwritten with the configured traffic-class value.
- In-band traffic destined to router management interface cannot be captured using this functionality.

## SPAN Types, Supported Features, and Configurations

### Local SPAN

This is the most basic form of traffic mirroring. The network analyzer or sniffer is attached directly to the destination interface. In other words, all monitored ports are located on the same router as the destination port.



## Remote SPAN

**Table 13: Feature History Table**

| Feature Name                                   | Release Information | Feature Description                                                                        |
|------------------------------------------------|---------------------|--------------------------------------------------------------------------------------------|
| Port Mirroring Enhancements on NC57 Line Cards | Release 7.4.1       | You can configure a subinterface as a destination on Cisco NC57 line cards in native mode. |

From Release 7.4.1, the destination can be an L2 subinterface on NC57 line cards.

From Release 7.4.1, a restricted form of remote traffic mirroring or remote SPAN is implemented on NC57 line cards. In this form, the router sends traffic to a single destination port that pushes a VLAN tag. Destination interface is a subinterface with VLAN encapsulation.

### Configure Remote Traffic Mirroring

#### Step 1 **configure**

**Example:**

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

Enters global configuration mode.

#### Step 2 **monitor-session** *session-name*

**Example:**

```
RP/0/RP0/CPU0:router(config)# monitor-session mon1 ethernet
RP/0/RP0/CPU0:router(config-mon)#
```

Defines a monitor session and enters monitor session configuration mode.

#### Step 3 **destination interface** *subinterface*

**Example:**

```
RP/0/RP0/CPU0:router(config-mon)# destination interface TenGigE 0/2/0/4.1
```

Specifies the destination subinterface to which traffic is replicated.

#### Step 4 **exit**

**Example:**

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

Exits monitor session configuration mode and returns to global configuration mode.

#### Step 5 **interface** *type number*

**Example:**

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

Enters interface configuration mode for the specified source interface. The interface number is entered in *rack/slot/module/port* notation. For more information about the syntax for the router, use the question mark (?) online help function.

### Step 6 **monitor-session** *session-name* **ethernet direction rx-onlyport-only**

#### Example:

```
RP/0/RP0/CPU0:router(config-if)# monitor-session mon1 ethernet
direction rx-only port-only
```

Specifies the monitor session to be used on this interface. Use the **direction** keyword to specify that only ingress or egress traffic is mirrored.

### Step 7 **end** or **commit**

#### Example:

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

or

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

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

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

- 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.

Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

### Step 8 **show monitor-session** [*session-name*] **status** [*detail*] [*error*]

#### Example:

```
RP/0/RP0/CPU0:router# show monitor-session
```

Displays information about the traffic mirroring session.

## Example

This example shows the basic configuration for traffic mirroring with physical interfaces.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# monitor-session ms1
RP/0/RP0/CPU0:router(config-mon)# destination interface HundredGigE0/2/0/15
RP/0/RP0/CPU0:router(config-mon)# commit

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface TenGigE0/2/0/19
RP/0/RP0/CPU0:router(config-if)# monitor-session ms1 port-level
RP/0/RP0/CPU0:router(config-if)# commit

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface TenGigE0/2/0/19
RP/0/RP0/CPU0:router(config-if)# monitor-session ms1 direction rx-only port-level
RP/0/RP0/CPU0:router(config-if)# commit

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface TenGigE0/2/0/19
RP/0/RP0/CPU0:router(config-if)# monitor-session ms1 direction tx-only port-level
RP/0/RP0/CPU0:router(config-if)# commit
```

This example shows sample output of the show monitor-session command with the status keyword:

```
RP/0/RSP0/CPU0:router# show monitor-session status
Monitor-session cisco-rtpl
Destination interface HundredGigE 0/5/0/38
=====
Source Interface Dir Status

TenGigE0/5/0/4 Both Operational
TenGigE0/5/0/17 Both Operational
RP/0/RSP0/CPU0:router# show monitor-session status detail
Monitor-session sess1
Destination interface is not configured
Source Interfaces

TenGigE0/1/0/0
Direction: Both
ACL match: Disabled
Portion: Full packet
Status: Not operational (destination interface not known).
TenGigE0/1/0/1
Direction: Both
ACL match: Disabled
Portion: First 100 bytes

RP/0/RSP0/CPU0:router# show monitor-session status error
Monitor-session ms1
Destination interface TenGigE0/2/0/15 is not configured
=====
Source Interface Dir Status

Monitor-session ms2
Destination interface is not configured
=====
Source Interface Dir Status

RP/0/RP0/CPU0:router# show monitor-session test status
Monitor-session test (ipv4)
Destination Nexthop 255.254.254.4
=====
```

```

Source Interface Dir Status

Gi0/0/0/2.2 Rx Not operational (source same as destination)
Gi0/0/0/2.3 Rx Not operational (Destination not active)
Gi0/0/0/2.4 Rx Operational
Gi0/0/0/4 Rx Error: see detailed output for explanation
RP/0/RP0/CPU0:router# show monitor-session test status error
Monitor-session test
Destination Nexthop ipv4 address 255.254.254.4
=====
Source Interface Status

Gi0/0/0/4 < Error: FULL Error Details >

```

## SPAN on Subinterfaces

Layer 2 source ports can be mirrored on Cisco NCS 5500 routers and Cisco NC57 line cards.

### Monitoring Traffic Mirroring on a Layer 2 Interface

This section describes the configuration for monitoring traffic on a Layer 2 interface.

#### Configuration

To monitor traffic mirroring on a Layer 2 interface, configure the monitor under `l2transport` sub-config of the interface:

```

RP/0/RP0/CPU0:router(config)# interface TenGigE0/0/0/42
RP/0/RP0/CPU0:router(config-if)# l2transport
RP/0/RP0/CPU0:router(config-if-l2)# monitor-session EASTON ethernet port-level

```

#### Verification

```

RP/0/RP0/CPU0:router# show monitor-session status
Thu Aug 29 21:42:22.829 UTC
Monitor-session EASTON
Destination interface TenGigE0/0/0/20
=====
Source Interface Dir Status

Te0/0/0/42 (port) Both Operational

```

## ACL-based SPAN

Traffic is mirrored based on the configuration of the interface ACL.

You can mirror traffic based on the definition of an interface access control list. When you mirror Layer 3 traffic, the ACL is configured using the **ipv4 access-list** or the **ipv6 access-list** command with the **capture** option. The **permit** and **deny** commands determine if the packets in the traffic are permitted or denied. The **capture** option designates the packet is to be mirrored to the destination port, and it is supported only on permit type of Access Control Entries (ACEs).




---

**Note** Prior to Release 6.5.1, ACL-based traffic mirroring required the use of UDK (User-Defined TCAM Key) with the **enable-capture** option so that the **capture** option can be configured in the ACL.

---

## Configuring Security ACLs for Traffic Mirroring

This section describes the configuration for creating security ACLs for traffic mirroring.

In ACL-based traffic mirroring, traffic is mirrored based on the configuration of the interface ACL. You can mirror traffic based on the definition of an interface access control list. When you're mirroring Layer 3 or Layer 2 traffic, the ACL is configured using the **ipv4 access-list** or the **ipv6 access-list** command with the **capture** option. The **permit** and **deny** commands determine the behavior of the regular traffic.

### Configure an IPv4 ACL for Traffic Mirroring

Use the following steps to configure ACLs for traffic mirroring.

```
/* Create an IPv4 ACL (TM-ACL) for traffic mirroring */
Router(config)# ipv4 access-list TM-ACL
Router(config-ipv4-acl)# 10 permit udp 10.1.1.0 0.0.0.255 eq 10 any capture
Router(config-ipv4-acl)# 20 permit udp 10.1.1.0 0.0.0.255 eq 20 any
Router(config-ipv4-acl)# exit
Router(config)# commit

/* Validate the configuration */
Router(config)# show run
Thu May 17 11:17:49.968 IST
Building configuration...
!! IOS XR Configuration 0.0.0
!! Last configuration change at Thu May 17 11:17:47 2018 by user
...
ipv4 access-list TM-ACL
 10 permit udp 10.1.1.0 0.0.0.255 eq 10 any capture
 20 permit udp 10.1.1.0 0.0.0.255 eq 20 any
!
```

You have successfully configured an IPv4 ACL for traffic mirroring.

## Configuring UDF-Based Security ACL for Traffic Mirroring

### Before you begin

This section describes the configuration steps for UDF-based security ACLs for traffic mirroring.

#### Step 1 configure

##### Example:

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

#### Step 2 udf *udf-name* header {inner | outer} {12 | 13 | 14} offset *offset-in-bytes* length *length-in-bytes*

##### Example:

```
RP/0/RP0/CPU0:router(config)# udf udf3 header outer 14 offset 0 length 1
(config-mon)#
```

##### Example:

```
RP/0/RP0/CPU0:router(config)# udf udf3 header inner 14 offset 10 length 2
(config-mon)#
```

**Example:**

```
RP/0/RP0/CPU0:router(config)# udf udf3 header outer 14 offset 50 length 1
(config-mon)#
```

Configures individual UDF definitions. You can specify the name of the UDF, the networking header from which offset, and the length of data to be extracted.

The **inner** or **outer** keywords indicate the start of the offset from the unencapsulated Layer 3 or Layer 4 headers, or if there is an encapsulated packet, they indicate the start of offset from the inner L3/L4.

**Note** The maximum offset allowed, from the start of any header, is 63 bytes

The **length** keyword specifies, in bytes, the length from the offset. The range is from 1 to 4.

**Step 3** **ipv4 access-list** *acl-name***Example:**

```
RP/0/RP0/CPU0:router(config)# ipv4 access-list acl1
```

Creates ACL and enters IP ACL configuration mode. The length of the *acl-name* argument can be up to 64 characters.

**Step 4** **permit** *regular-ace-match-criteria* **udf** *udf-name1 value1 ... udf-name8 value8***Example:**

```
RP/0/RP0/CPU0:router(config-ipv4-acl)# 10 permit ipv4 any any udf udf1 0x1234 0xffff udf3 0x56 0xff
capture
RP/0/RP0/CPU0:router(config-ipv4-acl)# 30 permit ipv4 any any dscp af11 udf udf5 0x22 0x22 capture
```

Configures ACL with UDF match.

**Step 5** **exit****Example:**

```
RP/0/RP0/CPU0:router(config-ipv4-acl)# exit
```

Exits IP ACL configuration mode and returns to global configuration mode.

**Step 6** **interface** *number***Example:**

```
RP/0/RP0/CPU0:router(config)# interface HundredGigE 0/2/0/2
```

Configures interface and enters interface configuration mode.

**Step 7** **ipv4 access-group** *acl-name* **ingress****Example:**

```
RP/0/RP0/CPU0:router(config-if)# ipv4 access-group acl1 ingress
```

Applies access list to an interface.

**Step 8**      **commit****Example:**

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

Applies access list to an interface.

**Verifying UDF-based Security ACL**

Use the **show monitor-session status detail** command to verify the configuration of UDF on security ACL.

```
RP/0/RP0/CPU0:leaf1# show monitor-session 1 status detail
```

```
Fri May 12 19:40:39.429 UTC
Monitor-session 1
 Destination interface tunnel-ip3
 Source Interfaces

 TenGigE0/0/0/15
 Direction: Rx-only
 Port level: True
 ACL match: Enabled
 Portion: Full packet
 Interval: Mirror all packets
 Status: Not operational (destination not active)
```

**DSCP Bitmask to Filter Ingress SPAN Traffic**

*Table 14: Feature History Table*

| Feature Name                                | Release Information | Feature Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|---------------------------------------------|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DSCP Bitmask to Filter Ingress SPAN Traffic | Release 7.5.4       | <p>You can now mirror multiple traffic flows for matched Differentiated Service Code Point (DSCP) value of IP header on the SPAN. The matched DSCP value is based on the DSCP value and the bitmask configured in Access Control List (ACL) rule.</p> <p>Earlier, you could monitor single traffic flow by setting the <a href="#">RFC 4594</a> defined DSCP values in the IP header.</p> <p>This feature introduces the following changes:</p> <ul style="list-style-type: none"> <li>• <b>CLI:</b> <a href="#">permit (IPv4)</a>, and <a href="#">permit (IPv6)</a> are modified to include new keyword <b>bitmask</b>.</li> <li>• <b>YANG DATA Model:</b> New XPathS for <a href="#">Cisco-IOS-XR-um-ipv4-access-list-cfg</a> and <a href="#">Cisco-IOS-XR-um-ipv6-access-list-cfg</a> (see <a href="#">Github</a>, <a href="#">YANG Data Models Navigator</a>).</li> </ul> |

Starting Release 7.5.4, You can configure an ACL rule with DSCP bitmask on the SPAN to mirror specific traffic flows.

Without ACL rule, SPAN mirrors all the traffic on the incoming port. When ACL is configured with DSCP and DSCP mask on the SPAN, SPAN mirrors the traffic whose DSCP value lies within the combination of DSCP value and the specified mask.

A DSCP value is mapped to a single traffic class as per the defined value in [RFC2474](#). Masking the DSCP value in ACL rule allows to mirror multiple traffic flows. DSCP value and mask operate similar to IPv4 address and mask.

## Configure DSCP Bitmask to Filter Ingress SPAN Traffic

To configure DSCP bitmask, use the bitmask option along with the dscp option while configuring the ACL.

### Configuration Example for IPv4

This example shows how you can configure DSCP bitmask on ingress SPAN for IPv4 traffic.

```
/*configure the ACL*/
Router# config
Router(config)# ipv4 access-list acl1
Router(config-ipv4-acl)# 10 permit ipv4 host 192.0.2.1 any dscp af22 bitmask 0x3f
Router(config-ipv4-acl)# commit
Router(config-ipv4-acl)# exit

/* Perform the following configurations to attach the created ACL to an interface*/
Router(config)# interface HundredGigE0/0/0/6
Router(config-if)# ipv4 address 192.0.2.51 255.255.255.0

/* Monitor the ingress ACL applied and DSCP masked IPv4 traffic on SPAN*/
Router(config-if)# monitor-session TEST ethernet direction rx-only port-level acl ipv4 acl1
Router(config-if)# commit
```

### Running Configuration

```
Router(config)# show running-config ipv4 access-list
ipv4 access-list acl1
 10 permit ipv4 host 192.0.2.1 any dscp af22 bitmask 0x3f
!

interface HundredGigE0/0/0/6
 ipv4 address 192.0.2.51 255.255.255.0
 monitor-session TEST ethernet direction rx-only port-level acl ipv4 acl1
!
!
```

### Configuration Example for IPv6

This example shows how you can configure DSCP bitmask on ingress SPAN for IPv6 traffic.

```
/*configure the ACL*/
Router# config
Router(config)# ipv6 access-list acl1
Router(config-ipv6-acl)# 10 permit ipv6 host 2001:DB8::2/32 any dscp 33 bitmask 0x3f
Router(config-ipv6-acl)# commit
Router(config-ipv6-acl)# exit

/* Perform the following configurations to attach the created ACL to an interface*/
Router(config)# interface HundredGigE 0/0/10/3
Router(config-if)# ipv6 address 2001:DB8::1/32

/* Monitor the ingress ACL applied and DSCP masked IPv4 traffic on ERSPAN*/
```



```
Router(config-if)# monitor-session TEST ethernet direction rx-only port-level acl ipv6 acl1
Router(config-if)# commit
```

### Running Configuration

```
Router(config)# show running-config ipv6 access-list
ipv6 access-list acl1
 10 permit ipv6 acl1 host 2001:DB8::2/32 any dscp 33 bitmask 0x3f
!
interface HundredGigE0/0/10/3
 ipv6 address 2001:db8::1/32
 monitor-session TEST ethernet direction rx-only port-level acl ipv6 acl1
!
!
```

## Multiple SPAN ACL Sessions in Single Interface

Table 15: Feature History Table

| Feature Name                                     | Release Information | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|--------------------------------------------------|---------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Multiple SPAN ACL Sessions in a Single Interface | Release 7.5.4       | <p>With this release, you can configure multiple SPAN ACL sessions under a single interface. A maximum of three sessions can be configured simultaneously.</p> <p>This feature, which is supported on layer 3 interfaces, helps you in monitoring traffic from different parts of your network simultaneously to see the network's overall performance.</p> <p>In addition, using this feature, you can get a better network visibility, more efficient use of network resources, and flexibility.</p> <p>You should specify the monitor sessions to be used on the interface. Use the <b>monitor-session session name ethernet direction rx-only port-level</b> command to specify that only the ingress traffic is mirrored. This feature is not supported on subinterfaces.</p> |

This feature allows you to configure multiple SPAN ACL sessions in the same source interface. The maximum number of sessions that are supported under an interface is three. The ACL is applicable only in the ingress direction (direction Rx). This configuration is supported only on Layer 3 interfaces.

To differentiate multiple SPAN sessions under the same source interface, span session ID is used. When a packet matches multiple entries at the router, priority attribute is used to choose the correct destination for the packet. When a single packet tries to match multiple SPAN sessions, you should configure correct priority fields to identify the correct destination. The ACL with the lowest priority is chosen.

For Cisco NCS 5500 routers, the merge group value is always 1, and the priority value can be of any value within the supported range of 1 to 1000.

Multiple SPAN ACL sessions in a single interface help the administrators in the following ways:

- Monitor traffic from different parts of your network simultaneously to see the overall network performance.
- Isolate traffic from specific networks for troubleshooting network issues.
- Segment traffic for different purposes, such as security, compliance, or performance analysis.

## Configure Multiple SPAN ACL Sessions

### Configure Multiple SPAN ACL Sessions

Specify the monitor sessions to be used on the interface. Use the direction keyword to specify that only ingress traffic is mirrored. See the following example:

```
Router(config)#interface TenGigE0/0/0/26
Router(config-if)#monitor-session ses1 ethernet direction rx-only port-level
Router(config-if)#acl ipv4 acl1
!
Router(config-if)#monitor-session ses2 ethernet direction rx-only port-level
Router(config-if)#acl ipv4 acl2
!
Router(config-if)#monitor-session ses3 ethernet direction rx-only port-level
Router(config-if)#acl ipv4 acl3
!
Router(config-if)#monitor-session ses4 ethernet direction rx-only port-level
Router(config-if)#acl ipv4 acl4
!
!
```

### Verify the Sessions

The following example shows the details of the monitor sessions.

```
Router##sh monitor-session status
Tue Mar 21 16:14:15.879 UTC
Monitor-session ses1
Destination interface TenGigE0/0/0/9
=====
Source Interface Dir Status

Te0/0/0/0 (port) Rx Operational

Monitor-session ses2
Destination interface TenGigE0/0/0/1
=====
Source Interface Dir Status

Te0/0/0/0 (port) Rx Operational

Monitor-session ses3
Destination interface TenGigE0/0/0/2
=====
Source Interface Dir Status

Te0/0/0/0 (port) Rx Operational

RP/0/RP0/CPU0:ios#
```

### Configuring the Correct Priority

When one packet tries to match more than one SPAN session, the priority field helps in identifying the correct destination.



**Note** Merge group and priority fields are not mandatory. But if used, configure both fields.

```
Router(config)#interface tenGigE 0/0/0/24
Router(config-if)#monitor-session ses1 ethernet port-level
Router(config-if)#acl ipv4 acl1 merge-group 1 priority 30
```

To verify the traffic, use the following sample **show monitor-session** command:

```
Router#show monitor-session status detail
Tue Mar 21 16:15:02.741 UTC
Monitor-session ses1
 Destination interface TenGigE0/0/0/9
 Source Interfaces

 TenGigE0/0/0/0
 Direction: Rx-only
 Port level: True
 ACL match: Disabled
 IPv4 ACL: Enabled (acl1, merge-group: 1,priority: 1)
 IPv6 ACL: Disabled
 Portion: Full packet
 Interval: Mirror all packets
 Mirror drops: Disabled
 Status: Operational

Monitor-session ses2
 Destination interface TenGigE0/0/0/1
 Source Interfaces

 TenGigE0/0/0/0
 Direction: Rx-only
 Port level: True
 ACL match: Disabled
 IPv4 ACL: Enabled (acl2)
 IPv6 ACL: Disabled
 Portion: Full packet
 Interval: Mirror all packets
 Mirror drops: Disabled
 Status: Operational

Monitor-session ses3
 Destination interface TenGigE0/0/0/2
 Source Interfaces

 TenGigE0/0/0/0
 Direction: Rx-only
 Port level: True
 ACL match: Disabled
 IPv4 ACL: Enabled (acl3)
 IPv6 ACL: Disabled
 Portion: Full packet
 Interval: Mirror all packets
 Mirror drops: Disabled
 Status: Operational

Monitor-session ses4
 Destination interface TenGigE0/0/0/6
 Source Interfaces

 TenGigE0/0/0/0
 Direction: Rx-only
 Port level: True
 ACL match: Disabled
 IPv4 ACL: Enabled (acl4)
 IPv6 ACL: Disabled
 Portion: Full packet
 Interval: Mirror all packets
 Mirror drops: Disabled
 Status: Operational
Router#
```

## Monitor Multiple SPAN ACL and Security ACL Sessions

Table 16: Feature History Table

| Feature Name                                        | Release Information | Feature Description                                                                                                                                                                                                                                                                |
|-----------------------------------------------------|---------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Monitor Multiple SPAN ACL and Security ACL Sessions | Release 7.5.4       | With this feature, you can use SPAN and security ACLs together to monitor multiple SPAN ACL sessions under the same source interface. SPAN ACL helps you to distribute the mirrored traffic over different destination interfaces. Security ACL allows selective incoming traffic. |

Starting Cisco IOS XR Software Release 7.5.4 you can monitor multiple ERSPAN sessions using GREv4 under the same source interface. Multiple SPAN ACL monitor sessions configured on an interface allow you to choose the destination interface for the mirrored traffic. For the configuration of monitor sessions, you can use SPAN and security ACLs together.

The SPAN and security ACLs apply only in the ingress traffic.

### Configure Multiple SPAN ACL and Security ACL Monitor Sessions

This example shows how to attach the SPAN and security ACLs to configure multiple monitoring sessions.

#### Configuration example

Use the following configuration to attach SPAN and security ACLs for traffic mirroring.

```
Router# config
/*Perform the following configurations to attach the SPAN ACL to an interface*/
Router(config-if)#monitor-session always-on-v4 ethernet direction rx-only port-level
Router(config-if-mon)#acl ipv4 v4-monitor-acl1
Router(config-if-mon)#acl ipv6 v6-monitor-acl1
Router(config-if-mon)#exit
Router(config-if)#monitor-session on-demand-v4 ethernet direction rx-only port-level
Router(config-if-mon)#acl ipv4 v4-monitor-acl2
Router(config-if-mon)#acl ipv6 v6-monitor-acl2
Router(config-if-mon)#exit
/*Perform the following configurations to attach the security ACL to an interface*/
Router(config-if)#ipv4 access-group sec_aclv4 ingress
Router(config-if)#ipv6 access-group sec_aclv6 ingress
Router(config-if)#commit
```

#### Running configuration

```
Router(config)#show running-config interface
monitor-session always-on-v4 ethernet direction rx-only port-level
 acl ipv4 v4-monitor-acl2
 acl ipv6 v6-monitor-acl2
!
monitor-session on-demand-v4 ethernet direction rx-only port-level
 acl ipv4 v4-monitor-acl2
 acl ipv6 v6-monitor-acl2
!
```

```
ipv4 access-group sec_aclv4 ingress
ipv6 access-group sec_aclv6 ingress
!
```

## Attaching the Configurable Source Interface

### Step 1 **configure**

#### Example:

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

Enters global configuration mode.

### Step 2 **interface** *type number*

#### Example:

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

Enters interface configuration mode for the specified source interface. The interface number is entered in *rack/slot/module/port* notation. For more information about the syntax for the router, use the question mark (?) online help function.

### Step 3 **ipv4 access-group** *acl-name* {**ingress** | **egress**}

#### Example:

```
RP/0/RP0/CPU0:router(config-if)# ipv4 access-group acl1 ingress
```

Controls access to an interface.

### Step 4 **monitor-session** *session-name* **ethernet direction rx-only** **port-level** **acl**

#### Example:

```
RP/0/RP0/CPU0:router(config-if)# monitor-session mon1 ethernet direction rx-only port-level acl
RP/0/RP0/CPU0:router(config-if-mon)#
```

Attaches a monitor session to the source interface and enters monitor session configuration mode.

**Note** **rx-only** specifies that only ingress traffic is replicated.

### Step 5 **acl**

#### Example:

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

Specifies that the traffic mirrored is according to the defined ACL.

**Note** If an ACL is configured by name, then this step overrides any ACL that may be configured on the interface.

### Step 6 **exit**

#### Example:

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

Exits monitor session configuration mode and returns to interface configuration mode.

### Step 7 **end** or **commit**

#### Example:

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

or

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

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

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

- 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.

### Step 8 **show monitor-session [session-name] status [detail] [error]**

#### Example:

```
RP/0/RP0/CPU0:router# show monitor-session status
```

Displays information about the monitor session.

## ERSPAN

Encapsulated Remote Switched Port Analyzer (ERSPAN) transports mirrored traffic over an IP network. The traffic is encapsulated at the source router and is transferred across the network. The packet is decapsulated at the destination router and then sent to the destination interface.

Encapsulated Remote SPAN (ERSPAN) enables generic routing encapsulation (GRE) for all captured traffic and allows it to be extended across Layer 3 domains.

ERSPAN involves mirroring traffic through a GRE tunnel to a remote site. For more information on configuring the GRE tunnel that is used as the destination for the monitor sessions, see the chapter *Configuring GRE Tunnels*.



**Note** A copy of every packet includes the Layer 2 header if the ethernet keyword is configured. As this renders the mirrored packets unroutable, the end point of the GRE tunnel must be the network analyzer.

## Introduction to ERSPAN Egress Rate Limit

With ERSPAN egress rate limit feature, you can monitor traffic flow through any IP network. This includes third-party switches and routers.

ERSAPN operates in the following modes:

- ERSPAN Source Session – box where the traffic originates (is SPANned).
- ERSPAN Termination Session or Destination Session – box where the traffic is analyzed.

This feature provides rate limiting of the mirroring traffic or the egress traffic. With rate limiting, you can limit the amount of egress traffic to a specific rate, which prevents the network and remote ERSPAN destination traffic overloading. Be informed, if the egress rate-limit exceeds then the system may cap or drop the monitored traffic.

You can configure the QoS parameters on the traffic monitor session.

- Traffic Class (0 through 7)
  - Traffic class 0 has the lowest priority and 7 the highest.
  - The default traffic class is the same as that of the original traffic class.
- The Discard Class (0 through 2):
  - The default is 0.
  - The discard class configuration is used in WRED.

### Benefits

With ERSPAN Egress rate limit feature, you can limit the egress traffic or the mirrored and use the mirrored traffic for data analysis.

### Topology

*Figure 15: Topology for ERSPAN Egress Rate Limit*



The encapsulated packet for ERSPAN is in ARPA/IP format with GRE encapsulation. The system sends the GRE tunneled packet to the destination box identified by an IP address. At the destination box, SPAN-ASIC decodes this packet and sends out the packets through a port. ERSPAN egress rate limit feature is applied on the router egress interface to rate limit the monitored traffic.

The intermediate switches carrying ERSPAN traffic from source session to termination session can belong to any L3 network.

### Configure ERSPAN Egress Rate Limit

Use the following steps to configure ERSPAN egress rate limit:

```
monitor-session ERSPAN ethernet
destination interface tunnel-ip1
!

RP/0/RP0/CPU0:pyke-008#sh run int tunnel-ip 1

interface tunnel-ip1
ipv4 address 4.4.4.1 255.255.255.0
tunnel mode gre ipv4
tunnel source 20.1.1.1
tunnel destination 20.1.1.2
!

RP/0/RP0/CPU0:pyke-008#sh run int hundredGigE 0/0/0/16

interface HundredGigE0/0/0/16
ipv4 address 215.1.1.1 255.255.255.0
ipv6 address 3001::2/64
monitor-session ERSPAN ethernet direction rx-only port-level
 acl
!
ipv4 access-group ACL6 ingress
```

### Running Configuration

```
!! Policy-map to be used with the ERSPAN Destination (egress interface)
!! Traffic class is set to 5. For packets in this class, apply shaping
!! as well as WRED.
class-map match-any TC5
 match traffic-class 5
end-class-map
!
policy-map shape-foo
 class TC5
 random-detect discard-class 0 10000 bytes 40000 bytes
 random-detect discard-class 1 40000 bytes 80000 bytes
 random-detect discard-class 2 80000 bytes 200000 bytes
 shape average percent 15
 !
 class class-default
 !
end-policy-map
!
!!GRE Tunnel Interface
interface Loopback49
 ipv4 address 49.49.49.49 255.255.255.255
!
interface tunnel-ip100
 ipv4 address 130.100.1.1 255.255.255.0
 tunnel mode gre ipv4
 tunnel source 49.49.49.49
 tunnel destination 10.8.1.2
!
!!ERSPAN Monitor Session with GRE tunnel as the Destination Interface, and with QoS
configuration
monitor-session FOO ethernet
 destination interface tunnel-ip100
```



```

traffic-class 5
discard-class 1
!
!!ERSPAN Source Interface
interface TenGigE0/6/0/4/0
description connected to TGEN 9/5
ipv4 address 10.4.90.1 255.255.255.0
monitor-session FOO ethernet port-level
!
!
!!ERSPAN Destination ip-tunnel00's underlying interface, with egress policy-map shape-foo
attached
interface TenGigE0/6/0/9/0
service-policy output shape-foo
ipv4 address 10.8.1.1 255.255.255.0

```

### Verification

```
RP/0/RP0/CPU0:ios#show monitor-session FOO status detail
```

```
Wed May 2 15:14:05.762 UTC
```

```
Monitor-session FOO
```

```
Destination interface tunnel-ip100
```

```
Source Interfaces
```

```

```

```
TenGigE0/6/0/4/0
```

```
Direction: Both
```

```
Port level: True
```

```
ACL match: Disabled
```

```
Portion: Full packet
```

```
Interval: Mirror all packets
```

```
Status: Operational
```

```
RP/0/RP0/CPU0:ios#
```

```
show monitor-session <sess-id> status internal
```

```
RP/0/RP0/CPU0:ios#show monitor-session FOO status internal
```

```
Wed May 2 15:13:06.063 UTC
```

```
Information from SPAN Manager and MA on all nodes:
```

```
Monitor-session FOO (ID 0x00000001) (Ethernet)
```

```
SPAN Mgr: Destination interface tunnel-ip100 (0x0800001c)
```

```
Last error: Success
```

```
Tunnel data:
```

```
Mode: GREoIPv4
```

```
Source IP: 49.49.49.49
```

```
Dest IP: 10.8.1.2
```

```
VRF:
```

```
ToS: 0 (copied)
```

```
TTL: 255
```

```
DFbit: Not set
```

```
0/6/CPU0: Destination interface tunnel-ip100 (0x0800001c)
```

```
Tunnel data:
```

```
Mode: GREoIPv4
```

```
Source IP: 49.49.49.49
```

```
Dest IP: 10.8.1.2
```

```
VRF:
```

```
ToS: 0 (copied)
```

```
TTL: 255
```

```
DFbit: Not set
```

```
Information from SPAN EA on all nodes:
```

```
Monitor-session 0x00000001 (Ethernet)
```

```
0/6/CPU0: Name 'FOO', destination interface tunnel-ip100 (0x0800001c)
```

```
Platform, 0/6/CPU0:
```

```
Dest Port: 0xe7d
```

```

ERSPAN Encap:
 Tunnel ID: 0x4001380b
 ERSPAN Tunnel ID: 0x4001380c
 IP-NH Grp key: 0x3140000cc5
 IP-NH hdl: 0x308a5fa5e0
 IP-NH IFH: 0x30002a0
 IP-NH IPAddr: 10.4.91.2

NPU MirrorRx MirrorTx
00 0x00000003 0x00000004
01 0x00000003 0x00000004
02 0x00000003 0x00000004
03 0x00000003 0x00000004
04 0x00000003 0x00000004
05 0x00000003 0x00000004
RP/0/RP0/CPU0:ios#

```

## ERSPAN Traffic to a Destination Tunnel in a Non-Default VRF

**Table 17: Feature History Table**

| Feature Name                                                | Release Information | Description                                                                                                                                                                                                                                                                                                                                                                                                                                |
|-------------------------------------------------------------|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ERSPAN Traffic to a Destination Tunnel in a Non-Default VRF | Release 7.5.3       | <p>The tunnels are grouped under the VRFs and you can segregate the traffic towards a specific VRF domain.</p> <p>Encapsulated Remote Switched Port Analyzer (ERSPAN) now transports mirrored traffic through GRE tunnels with multiple VRFs, helping you design your network with multiple Layer 3 partitions.</p> <p>In earlier releases, ERSPAN transported mirrored traffic through GRE tunnels that belonged to only default VRF.</p> |

Here, the tunnel interface, where the traffic mirroring is destined, is now in a VRF.

The traffic coming out of the interfaces of a router do not have any grouping. By configuring a specific VRF, you can now identify the incoming traffic group.

### Configuration

Use the following command to configure a specific VRF:

```

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface tunnel-ip 2
RP/0/RP0/CPU0:router(config)# tunnel vrf red

```

For more information on enabling the tunnel mode in GRE, see [Configuring GRE Tunnels](#).

### Configuration example

The following example shows a tunnel interface configured with endpoints in a non-default VRF (**vrf: red**):

```
Router#show run int tunnel-ip 2
Thu Feb 3 06:18:28.075 UTC
interface tunnel-ip2
 ipv4 address 102.1.1.100 255.255.255.0
 tunnel tos 32
 tunnel mode gre ipv4
 tunnel source 120.1.1.100
 tunnel vrf red
 tunnel destination 120.1.1.1

Router#show monitor-session status
Thu Feb 3 06:18:11.061 UTC
Monitor-session ERSPAN-2
Destination interface tunnel-ip2
=====
Source Interface Dir Status

Te0/0/0/5 (port) Rx Operational
```

### Verification

The following CLI output shows how to verify, if the configured tunnel VRF is programmed in the session:

```
Router#show monitor-session ERSPAN-2 status internal
Thu Feb 3 06:19:50.014 UTC

Information from SPAN Manager and MA on all nodes:
Monitor-session ERSPAN-2 (ID 0x00000003) (Ethernet)
SPAN Mgr: Destination interface tunnel-ip2 (0x20008024)
Last error: Success
Tunnel data:
 Mode: GREoIPv4
 Source IP: 120.1.1.100
 Dest IP: 120.1.1.1
 VRF: red
 VRF TBL ID: 0
 ToS: 32
 TTL: 255
 DFbit: Not set
```

## DSCP Marking on Egress GRE Tunnel in ERSPAN

Table 18: Feature History Table

| Feature Name                                | Release Information | Feature Description                                                                                                                                                                                                                                                                  |
|---------------------------------------------|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DSCP Marking on Egress GRE Tunnel in ERSPAN | Release 7.5.4       | You can now set or modify Differentiated Service Code Point (DSCP) value on the ERSPAN GRE tunnel header. This feature allows you to control the QoS for your network's ERSPAN GRE tunnel traffic and eases the effort to control your customers' bandwidth across next-hop routers. |

Starting Cisco IOS XR Release 7.5.4, you can set or modify DSCP marking on ERSPAN GRE tunnels. ERSPAN uses GRE encapsulation to route SPAN capture traffic.

## Configure DSCP Marking on Egress GRE Tunnel in ERSPAN

### Configuration Example

This example shows how you can configure DSCP Marking on Egress GRE tunnel in ERSPAN.

```
Router#configure terminal
Router(config)#interface tunnel-ip1
Router(config-if)#tunnel tos 96
Router(config-if)#tunnel mode gre ipv4
Router(config-if)#tunnel source 192.0.2.1
Router(config-if)#tunnel destination 192.0.2.254
```




---

**Note** You can configure DSCP value on both IPv4 and IPv6 headers.

---

### Running Configuration

```
interface tunnel-ip1
 tunnel tos 96
 tunnel mode gre ipv4
 tunnel source 192.0.2.1
 tunnel destination 192.0.2.254
!
```

### Verification

You can use the following commands to verify that tos value is configured:

```
Router#show run interface tunnel-ip 1
interface tunnel-ip1
 ipv4 address 192.0.2.0/24
 tunnel tos 96
 tunnel mode gre ipv4
 tunnel source 192.0.2.1
 tunnel vrf red
 tunnel destination 192.0.2.254

Router#show monitor-session ERSPAN-2 status internal

Information from SPAN Manager and MA on all nodes:
Monitor-session ERSPAN-2 (ID 0x00000003) (Ethernet)
SPAN Mgr: Destination interface tunnel-ip1 (0x20008024)
 Last error: Success
 Tunnel data:
 Mode: GREoIPv4
 Source IP: 192.0.2.1
 Dest IP: 192.0.2.254
 VRF: red
 VRF TBL ID: 0
 ToS: 96
 TTL: 255
 DFbit: Not set
```

## SPAN over Pseudowire

Pseudo-wire traffic mirroring (known as PW-SPAN) is an extra functionality on the existing SPAN solutions. The existing SPAN solutions are monitored on a destination interface or through a GRE tunnel or RSPAN. In PW-SPAN, the traffic mirroring destination port is configured to be a pseudo-wire rather than a physical port. Here, the designated traffic on the source port is mirrored over the pseudo-wire to a central location. This allows the centralization of expensive network traffic analysis tools.

Because the pseudo-wire carries only mirrored traffic, this traffic is unidirectional. Incoming traffic from the remote provider edge is not allowed. Typically, a monitor session should be created with a destination pseudo-wire. This monitor session is one of the L2VPN xconnect segments. The other segment of the L2VPN VPWS is a pseudowire.



**Note** Only port-level source interfaces are supported.

### Configure SPAN over Pseudowire

Use the following steps to configure SPAN over Pseudowire:

#### Configure SPAN monitor session

```
RP/0/RP0/CPU0:router#config
RP/0/RP0/CPU0:router(config)#monitor-session M1
RP/0/RP0/CPU0:router(config-mon)#destination pseudowire
RP/0/RP0/CPU0:router(config-mon)#commit
```

#### Configure SPAN source

```
RP/0/RP0/CPU0:router#config
Fri Sep 6 03:49:59.312 UTC
RP/0/RP0/CPU0:router(config)#interface Bundle-Ether100
RP/0/RP0/CPU0:router(config-if)#monitor-session M1 ethernet port-level
RP/0/RP0/CPU0:router(config-if-mon)#commit
```

#### Configure l2vpn xconnect

```
RP/0/RP0/CPU0:router(config)#l2vpn
RP/0/RP0/CPU0:router(config-l2vpn)#pw-class span
RP/0/RP0/CPU0:router(config-l2vpn-pwc)#encapsulation mpls
RP/0/RP0/CPU0:router(config-l2vpn-pwc-mpls)#transport-mode ethernet
RP/0/RP0/CPU0:router(config-l2vpn)#xconnect group 1
RP/0/RP0/CPU0:router(config-l2vpn-xc)#p2p 2
RP/0/RP0/CPU0:router(config-l2vpn-xc-p2p)#monitor-session M1
RP/0/RP0/CPU0:router(config-l2vpn-xc-p2p)#neighbor ipv4 10.10.10.1 pw-id 2
RP/0/RP0/CPU0:router(config-l2vpn-xc-p2p)#pw-class span
RP/0/RP0/CPU0:router(config-l2vpn-xc-p2p)#commit
```

### Verify SPAN over Pseudowire

The following examples show how to verify SPAN over Pseudowire configuration.

To check monitor session status:

```
RP/0/RP0/CPU0:router#show run monitor-session M1
monitor-session M1 ethernet
 destination pseudowire

RP/0/RP0/CPU0:router#show monitor-session M1 status
```

```

Monitor-session M1
Destination pseudowire
Source Interface Dir Status
BE100 (port) Both Operational
BE400 (port) Both Operational

RP/0/RP0/CPU0:router#show monitor-session M1 status detail
Monitor-session M1
Destination pseudowire
Source Interfaces

Bundle-Ether100
 Direction: Both
 Port level: True
 ACL match: Disabled
 Portion: Full packet
 Interval: Mirror all packets
 Status: Operational
Bundle-Ether400
 Direction: Both
 Port level: True
 ACL match: Disabled
 Portion: Full packet
 Interval: Mirror all packets
 Status: Operational

```

To check underlying l2vpn xconnect:

```

RP/0/RP0/CPU0:router#show run l2vpn
l2vpn
pw-class span
 encapsulation mpls
 transport-mode ethernet
 !
 !
p2p 2
 monitor-session M1
 neighbor ipv4 10.10.10.1 pw-id 2
 pw-class span
 !
 !
p2p 10
 monitor-session M2
 neighbor ipv4 10.10.10.1 pw-id 10
 pw-class span
 !
 !
 !
RP/0/RP0/CPU0:router#show l2vpn xconnect
Fri Sep 6 03:41:15.691 UTC
Legend: ST = State, UP = Up, DN = Down, AD = Admin Down, UR = Unresolved,
 SB = Standby, SR = Standby Ready, (PP) = Partially Programmed

XConnect
Group Name ST Segment 1

1 2 UP M1
 ST
 Segment 2
 Description ST

1 10 UP M2
 ST
 Description ST

```

## SPAN-to-File

SPAN-to-File is an extension of the pre-existing SPAN feature that allows network packets to be mirrored to a file instead of an interface. This simplifies the analysis of the packets at a later stage. The file format is PCAP, which helps that data to be used by tools, such as tcpdump or Wireshark.



**Warning** Be cautious when you apply this feature to files located on interfaces with high traffic.

When a file is configured as a destination for a SPAN session, a buffer is created on each node to which the network packets are logged. The buffer is for all packets on the node regardless of which interface they are from, that is, multiple interfaces may be providing packets for the same buffer. The buffers are deleted when the session configuration is removed. The file is written by each node to a location on the active RP which contains the node ID of the node on which the buffer was located.

If multiple interfaces are attached to a session, then interfaces on the same node are expected to have their packets sent to the same file. Bundle interfaces can be attached to a session with a file destination, which is similar to attaching individual interfaces.

## SPAN-to-File Enhancements

*Table 19: Feature History Table*

| Feature Name                      | Release Information | Feature Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|-----------------------------------|---------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SPAN Mirror First                 | Release 7.5.2       | <p>With your knowledge of expected packet header size, you can now mirror only the first N bytes of a packet where N can have possible values from 1 through 10000. This allows only the packet headers to be mirrored and not the user payload, ensuring the privacy and security of user data. It also reduces the load on network resources by processing only a few bytes to identify issues in the network.</p> <p>With the introduction of this feature, you can use the <b>mirror first</b> option in the global configuration mode of the <b>monitor-session</b> command.</p>                                                                                                                                                                                                                                              |
| SPAN-to-File - PCAPng File Format | Release 7.3.1       | <p>PCAPng is the next generation of packet capturing format that contains data packets captured over a network and stored in a standard format.</p> <p>The PCAPng file contains different types of information blocks, such as the section header, interface description, enhanced packet, simple packet, name resolution, and interface statistics. These blocks can be used to rebuild the captured packets into recognizable data.</p> <p>The PCAPng file format:</p> <ul style="list-style-type: none"> <li>• Provides the capability to enhance and extend the existing capabilities of data storage over time</li> <li>• Allows you to merge or append data to an existing file.</li> <li>• Enables to read data independently from network, hardware, and operating system of the machine that made the capture.</li> </ul> |

## Configure SPAN-to-File

Use the following command to configure SPAN to File:

```
monitor-session <name> [ethernet|ipv4|ipv6|mpls-ipv4|mpls-ipv6]
 destination file [size <kbytes>] [buffer-type linear]
```

The `monitor-session <name> [ethernet|ipv4|ipv6|mpls-ipv4|mpls-ipv6]` part of the command creates a monitor-session with the specified name and class and is a pre-existing chain point from the current SPAN feature. The `destination file [size <kbytes>] [buffer-type linear]` part of the command adds a new “file” option to the existing “destination”.

`destination file` has the following configuration options:

- Buffer size.
- Two types of buffer:
  - Circular: Once the buffer is full, the start is overwritten.
  - Linear: Once the buffer is full, no further packets are logged.




---

**Note** The default buffer-type is circular. Only linear buffer is explicitly configurable. Changing any of the parameters (buffer size or type) recreates the session, and clears any buffers of packets.

---

All configuration options which are applied to an attachment currently supported for other SPAN types should also be supported by SPAN to file. This may include:

- ACLs
- Write only first X bytes of packet.
- Mirror interval from 512 to 16k.




---

**Note** These options are implemented by the platform when punting the packet.

---

Once a session has been created, then interfaces may be attached to it using the following configuration:

```
interface GigabitEthernet 0/0/0/0
 monitor-session <name> [ethernet|ipv4|ipv6|mpls-ipv4|mpls-ipv6]
```

The attachment configuration is unchanged by SPAN-to-File feature.




---

**Note** Once the SPAN-to-File session is attached to source interface, mirroring starts and packets are punted from NPU to CPU and dropped at CPU until the **packet-collection start action** command is executed.

---

### Configuration Examples

To configure a `mon1` monitor session, use these commands:

```
monitor-session mon1 ethernet
 destination file size 230000
!
```



In the above example, omitting the `buffer-type` option results in default circular buffer.

To configure a `mon2` monitor session with the `linear` buffer type, use these commands:

```
monitor-session mon2 ethernet
 destination file size 1000 buffer-type linear
!
```

To attach monitor session to a physical or bundle interface, use these commands:

```
interface Bundle-Ether1
monitor-session ms7 ethernet
!
```

To configure a `mon3` monitor session with the `mirror first` option, use these commands:

```
monitor-session mon3 ethernet
mirror first 101
!
```

### Running Configuration

```
!! IOS XR Configuration 7.1.1.124I
!! Last configuration change at Tue Nov 26 19:29:05 2019 by root
!
hostname OC
logging console informational
!
monitor-session mon2 ethernet
 destination file size 1000 buffer-type linear

!
interface Bundle-Ether1
monitor-session ms7 ethernet
end
```

### Verification

To verify packet collection status:

```
RP/0/RP0/CPU0:router#show monitor-session status
Monitor-session mon1
Destination File - Packet collecting
=====
Source Interface Dir Status

Hu0/9/0/2 Rx Operational

Monitor-session mon2
Destination File - Packet collecting
=====
Source Interface Dir Status

BE2.1 Rx Operational
```

If packet collection is not active, the following line is displayed:

```
Monitor-session mon2
Destination File - Not collecting
```

Here, `Status-Operational` and `Destination File - Not collecting` indicates that mirroring has started and packets are being punted from NPU to CPU but getting dropped at CPU until the **packet-collection start action** command is executed.

## Action Commands for SPAN-to-File

Action commands are added to start and stop network packet collection. The commands may only be run on sessions where the destination is a file. The action command auto-completes names of globally configured SPAN to File sessions. See the table below for more information on action commands.

**Table 20: Action Commands for SPAN-to-File**

| Action | Command                                                                                                                                 | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|--------|-----------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Start  | <pre>monitor-session &lt;name&gt; packet-collection start</pre>                                                                         | <p>Issue this command to start writing packets for the specified session to the configured buffer.</p> <p>Once the SPAN is configured and operational, the packets are punted to CPU and dropped by CPU until the <code>monitor-session &lt;name&gt; packet-collection start</code> command is executed.</p>                                                                                                                                                                                                                                                                               |
| Stop   | <pre>monitor-session &lt;name&gt; packet-collection stop [ discard-data   write directory &lt;dir&gt; filename &lt;filename&gt; ]</pre> | <p>Issue this command to stop writing packets to the configured buffer.</p> <ul style="list-style-type: none"> <li>• <code>discard-data</code>: Specify this option to clear the buffer.</li> <li>• <code>discard-data</code>: Specify this option to write the buffer to the disk before it is cleared.</li> </ul> <p>The buffer is written in .pcap format in this location:<br/> <code>/&lt;directory&gt;/&lt;node_id&gt;/&lt;filename&gt;.pcap</code>.</p> <p>The .pcap extension that the user adds to the filename is removed automatically to avoid a duplicate file extension.</p> |

## File Mirroring

Prior to Cisco IOS XR Software Release 7.2.1, the router did not support file mirroring from active RP to standby RP. Administrators had to manually perform the task or use EEM scripts to sync files across active RP and standby RP. Starting with Cisco IOS XR Software Release 7.2.1, the file mirroring feature enables the router to copy files or directories automatically from `/harddisk:/mirror` location in active RP to `/harddisk:/mirror` location in standby RP or RSP without user intervention or EEM scripts.

Two new CLIs have been introduced for the file mirroring feature:

- **mirror enable**

The `/harddisk:/mirror` directory is created by default, but file mirroring functionality is only enabled by executing the `mirror enable` command from configuration terminal. Status of the mirrored files can be viewed with `show mirror status` command.

- **mirror enable checksum**

The `mirror enable checksum` command enables MD5 checksum across active to standby RP to check integrity of the files. This command is optional.

## Configure File Mirroring

File mirroring has to be enabled explicitly on the router. It is not enabled by default.

```
RP/0/RSP0/CPU0:router#show run mirror
```

```
Thu Jun 25 10:12:17.303 UTC
mirror enable
mirror checksum
```

Following is an example of copying running configuration to `harddisk:/mirror` location:

```
RP/0/RSP0/CPU0:router#copy running-config harddisk:/mirror/run_config
Wed Jul 8 10:25:51.064 PDT
Destination file name (control-c to abort): [/mirror/run_config]?
Building configuration..
32691 lines built in 2 seconds (16345)lines/sec
[OK]
```

### Verification

To verify the syncing of file copied to mirror directory, use the `show mirror` command.

```
RP/0/RSP0/CPU0:router#show mirror
Wed Jul 8 10:31:21.644 PDT
% Mirror rsync is using checksum, this show command may take several minutes if you have
many files. Use Ctrl+C to abort
MIRROR DIR: /harddisk:/mirror/
% Last sync of this dir ended at Wed Jul 8 10:31:11 2020
Location	Mirrored	MD5 Checksum	Modification Time
run_config |yes |76fclb906bec4fe08ecda0c93f6c7815 |Wed Jul 8 10:25:56 2020
```

If checksum is disabled, `show mirror` command displays the following output:

```
RP/0/RSP0/CPU0:router#show mirror
Wed Jul 8 10:39:09.646 PDT
MIRROR DIR: /harddisk:/mirror/
% Last sync of this dir ended at Wed Jul 8 10:31:11 2020
Location	Mirrored	Modification Time
run_config |yes |Wed Jul 8 10:25:56 2020
```

If there is a mismatch during the syncing process, use `show mirror mismatch` command to verify.

```
RP/0/RP0/CPU0:router# show mirror mismatch
Wed Jul 8 10:31:21.644 PDT
MIRROR DIR: /harddisk:/mirror/
% Last sync of this dir ended at Wed Jul 8 10:31:11 2020
Location	Mismatch Reason	Action Needed
test.txt |newly created item. |send to standby
```

## Forward-Drop Packets Mirroring

In a network, packets are forwarded from one device to another until they reach their destination. However, in some cases, routers may drop packets during this forwarding process. These packets are known as forward-drop packets.

Packets can be dropped for several reasons such as congestion on the network, errors in the packet header or payload, blocking by firewall or access control lists (ACL), and so on. These forward-drop packets are typically discarded before they can reach their intended destination, and may have to be re-transmitted by the source device. This feature supports mirroring of these forward-drop packets at the ingress (Rx direction) to another destination. When a global forward-drop session is configured for the router, the forward-drop packets at the ingress are mirrored or copied to the configured destination. You can configure the mirror destination as a file (for SPAN-to-file sessions) or an IPv4 GRE tunnel ID (for ERSPAN).

### Mirror Forward-Drop Packets

Table 21: Feature History Table

| Feature Name                | Release Information | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|-----------------------------|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Mirror Forward-Drop Packets | Release 7.5.4       | <p>Mirroring forward-drop packets feature copies or mirrors the packets that are dropped during the forwarding process at the router ingress to a configured destination. These mirrored packets can be captured and analyzed using network monitoring tools. The analysis of dropped packets helps you understand the types of traffic that are blocked, analyze potential security threats, troubleshoot, and optimize network performance.</p> <p>This feature introduces the following changes:</p> <ul style="list-style-type: none"> <li>• <b>CLI:</b> <a href="#">forward-drop rx</a></li> <li>• <b>YANG Data Model:</b> New XPath for Cisco-IOS-XR-um-monitor-session-cfg.yang (see <a href="#">GitHub</a>, <a href="#">YANG Data Models Navigator</a>)</li> </ul> |

Mirroring forward-drop packets to a suitable destination for analysis can help in the following:

- **Network visibility:** By mirroring and analyzing forward-drop packets, network administrators gain better visibility into the types of traffic that are blocked by the firewalls and access control lists (ACL).
- **Threat detection:** As the original dropped packet is forwarded without any change, it helps in identifying the source of potential security threats.
- **Troubleshooting:** Analyzing forward-drop packets helps in troubleshooting network issues that may be causing the packet drop. This helps in taking proactive measures to avoid escalation of the issue.

### Configure Forward-Drop Mirroring

Perform the following tasks on the router to configure a global session for mirroring forward-drop packets:

1. Configure the tunnel mode.
2. Configure the tunnel source.

3. Configure the tunnel destination.
4. Configure a traffic mirroring session.
5. Associate a destination interface with the traffic mirroring session.
6. Run **forward-drop rx** command to start mirroring forward-drop packets.

This example shows how to configure a global traffic mirroring session for forward-drop packets.

```
Router(config)# interface tunnel-ip 2
Router(config-if)# tunnel mode gre ipv4
Router(config-if)# tunnel source 20.20.20.20
Router(config-if)# tunnel destination 192.1.1.3
Router(config-if)!
Router(config)# monitor-session mon2 ethernet
Router(config)#destination interface tunnel-ip2
Router(config)#forward-drop rx
Router(config)#!
```

### Running Configuration

This section shows forward-drop running configuration.

```
RP/0/RSP0/CPU0:router#sh running-config
interface tunnel-ip 2
tunnel mode gre ipv4
tunnel source 20.20.20.20
tunnel destination 192.1.1.3
!
monitor-session mon2 ethernet
destination interface tunnel-ip2
forward-drop rx
!
```

### Verification

Verify the forward-drop packets are mirrored using the **show monitor-session** command.

```
Router#show monitor-session mon2 status detail
Mon Aug 15 19:14:31.975 UTC
Monitor-session mon2
 Destination interface tunnel-ip2
 All forwarding drops:
 Direction: Rx
 Source Interfaces

```

## Troubleshoot Traffic Mirroring

When you encounter any issue with traffic mirroring, begin troubleshooting by checking the output of the **show monitor-session status** command. This command displays the recorded state of all sessions and source interfaces:

```
show monitor-session status
Monitor-session 5
rx destination interface tunnel-ip5
tx destination is not specified
=====
```

```

Source Interface Dir Status

Te0/0/0/23 (port) Rx Operational

```

In the preceding example, the line marked as `<Session status>` can indicate one of these configuration errors:

| Session Status                                                                      | Explanation                                                                                                                                                                                                                                                                                                             |
|-------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Session is not configured globally                                                  | The session does not exist in global configuration. Review the <b>show</b> command output and ensure that a session with a correct name has been configured.                                                                                                                                                            |
| Destination interface <code>&lt;intf&gt;</code> ( <code>&lt;down-state&gt;</code> ) | The destination interface is not in Up state in the Interface Manager. You can verify the state using the <b>show interfaces</b> command. Check the configuration to determine what might be keeping the interface from coming up (for example, a sub-interface needs to have an appropriate encapsulation configured). |

The `<Source interface status>` can report these messages:

| Source Interface Status                                         | Explanation                                                                                                                                                                                                                                                                             |
|-----------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Operational                                                     | Everything appears to be working correctly in traffic mirroring. If you are still having issues, follow up with the platform teams in the first instance, if mirroring is not operating as expected.                                                                                    |
| Not operational (Session is not configured globally)            | The session does not exist in global configuration. Check the <b>show</b> command output to ensure that a session with the right name has been configured.                                                                                                                              |
| Not operational (destination not known)                         | The session exists, but it either does not have a destination interface specified or the destination interface named for the session does not exist. For example, if the destination is a sub-interface that has not been created.                                                      |
| Not operational (source same as destination)                    | The session exists, but the destination and source are the same interface. In this case, traffic mirroring does not work.                                                                                                                                                               |
| Not operational (destination not active)                        | The destination interface or pseudowire is not in the Up state. See the corresponding <i>Session status</i> error messages for suggested resolutions.                                                                                                                                   |
| Not operational (source state <code>&lt;down-state&gt;</code> ) | The source interface is not in the Up state. You can verify the state using the <b>show interfaces</b> command. Check the configuration to see what might be keeping the interface from coming up (for example, a sub-interface needs to have an appropriate encapsulation configured). |
| Error: see detailed output for explanation                      | Traffic mirroring has encountered an error. Run the <b>show monitor-session status detail</b> command to display more information.                                                                                                                                                      |

The **show monitor-session status detail** command displays full details of the configuration parameters and any errors encountered. For example:

```
RP/0/RP0/CPU0:router show monitor-session status detail
```

```
Monitor-session sess1
```

```

Destination interface is not configured
Source Interfaces

TenGigE0/0/0/1
 Direction: Both
 ACL match: Disabled
 Portion: Full packet
 Status: Not operational (destination interface not known)
TenGigE0/0/0/2
 Direction: Both
 ACL match: Disabled
 Portion: First 100 bytes
 Status: Not operational (destination interface not known). Error: 'Viking SPAN PD' detected
the 'warning' condition 'PRM connection
 creation failure'.
Monitor-session foo
Destination next-hop TenGigE 0/0/0/0
Source Interfaces

TenGigE 0/1/0/0.100:
 Direction: Both
 Status: Operating
TenGigE 0/2/0/0.200:
 Direction: Tx
 Status: Error: <blah>

Monitor session bar
No destination configured
Source Interfaces

TenGigE 0/3/0/0.100:
 Direction: Rx
 Status: Not operational(no destination)

```

Here are additional trace and debug commands:

```

RP/0/RP0/CPU0:router# show monitor-session trace ?

platform Enable platform trace
process Filter debug by process(cisco-support)

RP/0/RP0/CPU0:router# show monitor-session trace platform ?

errors Display error traces(cisco-support)
events Display event traces(cisco-support)

RP/0/RP0/CPU0:router#show monitor-session trace platform events location all ?

usrtdir Specify directory to collect unsorted traces(cisco-support)
| Output Modifiers
<cr>

RP/0/RP0/CPU0:router#show monitor-session trace platform errors location all ?

usrtdir Specify directory to collect unsorted traces(cisco-support)
| Output Modifiers
<cr>

RP/0/RP0/CPU0:router# debug monitor-session process ?

all All SPAN processes(cisco-support)
ea SPAN EA(cisco-support)
ma SPAN MA(cisco-support)

```

```
mgr SPAN Manager(cisco-support)

RP/0/RP0/CPU0:router# debug monitor-session process all
RP/0/RP0/CPU0:router# debug monitor-session process ea
RP/0/RP0/CPU0:router# debug monitor-session process ma
RP/0/RP0/CPU0:router# show monitor-session process mgr

detail Display detailed output
errors Display only attachments which have errors
internal Display internal monitor-session information
| Output Modifiers

RP/0/RP0/CPU0:router# show monitor-session status
RP/0/RP0/CPU0:router# show monitor-session status errors
RP/0/RP0/CPU0:router# show monitor-session status internal
RP/0/RP0/CPU0:router# show tech-support span ?

file Specify a valid file name (e.g. disk0:tmp.log)
list-CLIs list the commands that would be run (don't execute) (cisco-support)
location Specify a location(cisco-support)
rack Specify a rack(cisco-support)
time-out per show command timeout configuration(cisco-support)
<cr>
```





## CHAPTER 9

# Configuring Virtual Loopback and Null Interfaces

This module describes the configuration of loopback and null interfaces. Loopback and null interfaces are considered virtual interfaces.

A virtual interface represents a logical packet switching entity within the router. Virtual interfaces have a global scope and do not have an associated location. Virtual interfaces have instead a globally unique numerical ID after their names. Examples are Loopback 0, Loopback 1, and Loopback 99999. The ID is unique per virtual interface type to make the entire name string unique such that you can have both Loopback 0 and Null 0.

Loopback and null interfaces have their control plane presence on the active route switch processor (RSP). The configuration and control plane are mirrored onto the standby RSP and, in the event of a failover, the virtual interfaces move to the ex-standby, which then becomes the newly active RSP.

- [Information About Configuring Virtual Interfaces, on page 233](#)

## Information About Configuring Virtual Interfaces

To configure virtual interfaces, you must understand the following concepts:

### Virtual Loopback Interface Overview

A virtual loopback interface is a virtual interface with a single endpoint that is always up. Any packet transmitted over a virtual loopback interface is immediately received by the same interface. Loopback interfaces emulate a physical interface.

In Cisco IOS XR Software, virtual loopback interfaces perform these functions:

- Loopback interfaces can act as a termination address for routing protocol sessions. This allows routing protocol sessions to stay up even if the outbound interface is down.
- You can ping the loopback interface to verify that the router IP stack is working properly.

In applications where other routers or access servers attempt to reach a virtual loopback interface, you must configure a routing protocol to distribute the subnet assigned to the loopback address.

Packets routed to the loopback interface are rerouted back to the router or access server and processed locally. IP packets routed out to the loopback interface but not destined to the loopback interface are dropped. Under these two conditions, the loopback interface can behave like a null interface.

## Prerequisites for Configuring Virtual Interfaces

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.

## Configuring Virtual Loopback Interfaces

This task explains how to configure a basic loopback interface.

### Restrictions

The IP address of a loopback interface must be unique across all routers on the network. It must not be used by another interface on the router, and it must not be used by an interface on any other router on the network.

### SUMMARY STEPS

1. **configure**
2. **interface loopback** *instance*
3. **ipv4 address** *ip-address*
4. **end** or **commit**
5. **show interface***type instance*

### DETAILED STEPS

---

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

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

Enters global configuration mode.

**Step 2** **interface loopback** *instance***Example:**

```
RP/0/RP0/CPU0:router#(config)# interface Loopback 3
```

Enters interface configuration mode and names the new loopback interface.

**Step 3** **ipv4 address** *ip-address***Example:**

```
RP/0/RP0/CPU0:router(config-if)# ipv4 address 100.100.100.69 255.255.255.255
```

Assigns an IP address and subnet mask to the virtual loopback interface using the **ipv4 address** configuration command.

**Step 4** **end** or **commit****Example:**

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

or

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

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

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

- 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.

#### Step 5 `show interface` *type instance*

##### Example:

```
RP/0/RP0/CPU0:router# show interfaces Loopback0
```

(Optional) Displays the configuration of the loopback interface.

### Example

This example shows how to configure a loopback interface:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface Loopback0
RP/0/RP0/CPU0:router(config-if)# ipv4 address 100.100.100.69 255.255.255.255
RP/0/RP0/CPU0:router(config-if)# ipv6 address 100::69/128
RP/0/RP0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes
RP/0/RP0/CPU0:router# show interfaces Loopback0
```

```
Loopback0 is up, line protocol is up
Interface state transitions: 1
Hardware is Loopback interface(s)
Internet address is 100.100.100.69/32
MTU 1500 bytes, BW 0 Kbit
 reliability Unknown, txload Unknown, rxload Unknown
Encapsulation Loopback, loopback not set,
Last link flapped 01:57:47
Last input Unknown, output Unknown
Last clearing of "show interface" counters Unknown
Input/output data rate is disabled.
```

## Null Interface Overview

A null interface functions similarly to the null devices available on most operating systems. This interface is always up and can never forward or receive traffic; encapsulation always fails. The null interface provides an alternative method of filtering traffic. You can avoid the overhead involved with using access lists by directing undesired network traffic to the null interface.

The only interface configuration command that you can specify for the null interface is the **ipv4 unreachable** command. With the **ipv4 unreachable** command, if the software receives a non-broadcast packet destined for itself that uses a protocol it does not recognize, it sends an Internet Control Message Protocol (ICMP) protocol unreachable message to the source. If the software receives a datagram that it cannot deliver to its ultimate destination because it knows of no route to the destination address, it replies to the originator of that datagram with an ICMP host unreachable message. By default **ipv4 unreachable** command is enabled. If we do not want ICMP to send protocol unreachable, then we need to configure using the **ipv4 icmp unreachable disable** command.

The Null 0 interface is created by default during boot process and cannot be removed. The **ipv4 unreachable** command can be configured for this interface, but most configuration is unnecessary because this interface just discards all the packets sent to it.

The Null 0 interface can be displayed with the **show interfaces null0** command.

## Configuring Null Interfaces

This task explains how to configure a basic null interface.

### SUMMARY STEPS

1. **configure**
2. **interface null 0**
3. **end** or **commit**
4. **show interfaces null 0**

### DETAILED STEPS

---

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

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

Enters global configuration mode.

**Step 2**    **interface null 0****Example:**

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

Enters the null 0 interface configuration mode.

**Step 3**    **end** or **commit****Example:**

```
RP/0/RP0/CPU0:router(config-null0)# end
```

or

```
RP/0/RP0/CPU0:router(config-null0)# commit
```

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

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

- 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.

#### Step 4 show interfaces null 0

##### Example:

```
RP/0/RP0/CPU0:router# show interfaces null 0
```

Verifies the configuration of the null interface.

#### Example

This example shows how to configure a null interface:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface Null 0
RP/0/RP0/CPU0:router(config-null0)# ipv4 icmp unreachable disable
RP/0/RP0/CPU0:router(config-null0)# end
Uncommitted changes found, commit them? [yes]: yes
RP/0/RP0/CPU0:router# show interfaces Null 0
```

```
Null0 is up, line protocol is up
Interface state transitions: 1
Hardware is Null interface
Internet address is Unknown
MTU 1500 bytes, BW 0 Kbit
reliability 255/255, txload Unknown, rxload Unknown
Encapsulation Null, loopback not set,
Last link flapped 4d20h
Last input never, output never
Last clearing of "show interface" counters 05:42:04
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

```

## Configuring Virtual IPv4 Interfaces

This task explains how to configure an IPv4 virtual interface.

### SUMMARY STEPS

1. **configure**
2. **ipv4 virtual address** *ipv4-*
3. **end** or **commit**

### DETAILED STEPS

#### Step 1 **configure**

##### Example:

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

Enters global configuration mode.

#### Step 2 **ipv4 virtual address** *ipv4-*

##### Example:

```
RP/0/RP0/CPU0:router(config)# ipv4 virtual address 10.3.32.154/8
```

Defines an IPv4 virtual address for the management Ethernet interface.

#### Step 3 **end** or **commit**

##### Example:

```
RP/0/RP0/CPU0:router(config-null0)# end
```

or

```
RP/0/RP0/CPU0:router(config-null0)# commit
```

Saves configuration changes.

- When you issue the **end** command, the system prompts you to commit changes:

```

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

```

- 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.
- 

### Example

This is an example for configuring a virtual IPv4 interface:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ipv4 virtual address 10.3.32.154/8
RP/0/RP0/CPU0:router(config-null0)# commit
```







## CHAPTER 10

# Configuring GRE Tunnels

Generic Routing Encapsulation (GRE) is a tunneling protocol that provides a simple generic approach to transport packets of one protocol over another protocol by means of encapsulation. This module provides information about how to configure a GRE tunnel.

- [Configuring GRE Tunnels, on page 241](#)
- [Single Pass GRE Encapsulation Allowing Line Rate Encapsulation, on page 243](#)

## Configuring GRE Tunnels

Tunneling provides a mechanism to transport packets of one protocol within another protocol. Generic Routing Encapsulation (GRE) is a tunneling protocol that provides a simple generic approach to transport packets of one protocol over another protocol with encapsulation. GRE encapsulates a payload, that is, an inner packet that needs to be delivered to a destination network inside an outer IP packet. The GRE tunnel behaves as a virtual point-to-point link that has two endpoints identified by the tunnel source and tunnel destination address. The tunnel endpoints send payloads through GRE tunnels by routing encapsulated packets through intervening IP networks. Other IP routers along the way do not parse the payload (the inner packet); they only parse the outer IP packet as they forward it towards the GRE tunnel endpoint. Upon reaching the tunnel endpoint, GRE encapsulation is removed and the payload is forwarded to the packet's ultimate destination.

Encapsulation by the outer packet takes place at the tunnel source whereas decapsulation of the outer packet takes place at the tunnel destination. Encapsulation and decapsulation data is collected periodically or on demand. Encapsulation statistics provide us the number of packets encapsulated at the tunnel source. Decapsulation statistics provide us the number of packets that are decapsulated at the tunnel destination. This data is stored as statistics in logical tables that are based on statistics type in the route processor. The different statistics types include L2 Interface TX Stats, L3 Interface TX Stats, TRAP stats, and so on. Encapsulation statistics can help you to infer the source of the traffic, and decapsulation statistics provide you the destination of the traffic. Decapsulation statistics also help you to detect the type of traffic as well.

### Guidelines and Restrictions for Configuring GRE Tunnels

The following restrictions apply while configuring GRE tunnels:

- The router supports up to 500 GRE tunnels.
- Only up to 16 unique source IP addresses are supported for the tunnel source.
- 2-pass to Single-pass migration, which means converting the same GRE tunnel, is not possible in a single configuration step. You must first delete the 2-pass tunnel and then add the Single-pass tunnel.

- Configurable MTU is not supported on Single-pass GRE interface, but supported on 2-pass GRE interface.
- L3VPN over GRE is not supported.
- To use the outer IPv4 GRE header for IP tunnel decapsulation in the hashing algorithm for ECMP and bundle member selection, use the `hw-module profile load-balance algorithm` command.

**Table 22: GRE Tunnels with Supported MTU and TOS Hardware Profiles**

| Supported Hardware                                                                                      | Profile Type | Maximum Supported Profile |
|---------------------------------------------------------------------------------------------------------|--------------|---------------------------|
| NC55-36x100G<br>NC55-18H18F<br>NC55-24x100G-SE<br>NC55-24H12F-SE<br>NC55-36x100G-S<br>NC55-6x200-DWDM-S | MTU          | 3                         |
| NC55-36x100G-A-SE<br>NC55-MOD-A-S<br>NC55-MOD-A-SE-S<br>NC55-32T16Q4H                                   | MTU          | 3                         |
| NC57-24DD<br>NC57-18DD-SE<br>NC57-36H-SE<br>NC57-36H6D<br>NC57-MOD-S                                    | MTU          | 7                         |
| NC55-36x100G<br>NC55-18H18F<br>NC55-24x100G-SE<br>NC55-24H12F-SE<br>NC55-36x100G-S<br>NC55-6x200-DWDM-S | TOS          | 8                         |
| NC55-36x100G-A-SE<br>NC55-MOD-A-S<br>NC55-MOD-A-SE-S<br>NC55-32T16Q4H                                   | TOS          | 8                         |



**Note** If the configured MTU and Tunnel TOS profile exceeds the supported hardware limit, the system displays SDK-Out of Memory error.

### Configuration Example

Configuring a GRE tunnel involves creating a tunnel interface and defining the tunnel source and destination. This example shows how to configure a GRE tunnel between Router1 and Router2. You need to configure tunnel interfaces on both the routers. Tunnel source IP address on Router1 will be configured as the tunnel destination IP address on Router2. Tunnel destination IP address on Router1 will be configured as the tunnel source IP address on Router2. In this example, OSPF is used as the routing protocol between the two routers. You can also configure BGP or IS-IS as the routing protocol.

```
RP/0/RP0/CPU0:Router1# configure
RP/0/RP0/CPU0:Router1(config)# interface tunnel-ip 30
RP/0/RP0/CPU0:Router1(config-if)# tunnel mode gre ipv4
RP/0/RP0/CPU0:Router1(config-if)# ipv4 address 10.1.1.1 255.255.255.0
RP/0/RP0/CPU0:Router1(config-if)# tunnel source 192.168.1.1
RP/0/RP0/CPU0:Router1(config-if)# tunnel destination 192.168.2.1
RP/0/RP0/CPU0:Router1(config-if)# exit
RP/0/RP0/CPU0:Router1(config)# interface Loopback 0
RP/0/RP0/CPU0:Router1(config-if)# ipv4 address 10.10.10.1
RP/0/RP0/CPU0:Router1(config-if)# exit
RP/0/RP0/CPU0:Router1(config)# router ospf 1
RP/0/RP0/CPU0:Router1(config-ospf)# router-id 192.168.4.1
RP/0/RP0/CPU0:Router1(config-ospf)# area 0
RP/0/RP0/CPU0:Router1(config-ospf-ar)# interface tunnel-ip 30
RP/0/RP0/CPU0:Router1(config-ospf-ar)# interface Loopback 0
RP/0/RP0/CPU0:Router1(config-ospf-ar)# commit

RP/0/RP0/CPU0:Router2# configure
RP/0/RP0/CPU0:Router2(config)# interface tunnel-ip 30
RP/0/RP0/CPU0:Router2(config-if)# tunnel mode gre ipv4
RP/0/RP0/CPU0:Router2(config-if)# ipv4 address 10.1.1.2 255.255.255.0
RP/0/RP0/CPU0:Router2(config-if)# tunnel source 192.168.2.1
RP/0/RP0/CPU0:Router2(config-if)# tunnel destination 192.168.1.1
RP/0/RP0/CPU0:Router2(config-if)# exit
RP/0/RP0/CPU0:Router2(config)# interface Loopback 0
RP/0/RP0/CPU0:Router2(config-if)# ipv4 address 2.2.2.2
RP/0/RP0/CPU0:Router2(config)# router ospf 1
RP/0/RP0/CPU0:Router2(config-ospf)# router-id 192.168.3.1
RP/0/RP0/CPU0:Router2(config-ospf)# area 0
RP/0/RP0/CPU0:Router2(config-ospf-ar)# interface tunnel-ip 30
RP/0/RP0/CPU0:Router2(config-ospf-ar)# interface Loopback 0
RP/0/RP0/CPU0:Router2(config-ospf-ar)# commit
```

## Single Pass GRE Encapsulation Allowing Line Rate Encapsulation

Single Pass GRE Encapsulation Allowing Line Rate Encapsulation feature, also known as Prefix-based GRE Tunnel Destination for Load Balancing feature, enables line rate GRE encapsulation traffic and enables flow entropy. Data-plane forwarding performance supports full line rate, which is adjusted to consider added encapsulation. GRE tunnel goes down if the destination is not available in RIB. Routing over GRE Single-pass tunnel is not supported in Release 6.3.2, so the traffic that is eligible for GRE encapsulation is identified using

an ACL filter that is based on GRE encapsulation. GRE tunnel destination address is an anycast address. All of the GRE encapsulation must be assigned based upon either an ACL or a policy-map, or both. Destinations may be individual addresses or /28 prefixes.

## Configure GRE Single-Pass Entropy

Perform the following tasks to configure the GRE Single-Pass Entropy feature:

- GRE Single-pass
- GRE Entropy(ECMP/UCMP)

```

/* GRE Single-Pass */

Router# configure
Router(config)# interface tunnel-ip30016
Router(config-if)# ipv4 address 216.1.1.1 255.255.255.0
Router(config-if)# ipv6 address 216:1:1::1/64
Router(config-if)# ipv6 enable
Router(config-if)# tunnel mode gre ipv4 encap
Router(config-if)# tunnel source Loopback22
Router(config-if)# tunnel destination 170.170.170.22
Router(config-if)# commit
Router(config-if)# exit

/* GRE Entropy (ECMP/UCMP) */

ECMP (ISIS)

Router# configure
Router(config)# router isis core
Router(config)# apply-group ISIS-INTERFACE
Router(config-isis)# is-type level-2-only
Router(config-isis)# net 49.1111.0000.0000.002.00
Router(config-isis)# nsr
Router(config-isis)# log adjacency changes
Router(config-isis)# address-family ipv4 unicast
Router(config-isis-af)# metric-style wide
Router(config-isis-af)# metric 2
Router(config-isis-af)# mpls traffic-eng level-2-only
Router(config-isis-af)# mpls traffic-eng router-id Loopback0
Router(config-isis-af)# maximum-paths 5
Router(config-isis-af)# commit
!

/* UCMP (ISIS) */

Router# configure
Router(config)# router isis core
Router(config)# apply-group ISIS-INTERFACE
Router(config-isis)# is-type level-2-only
Router(config-isis)# net 49.1111.0000.0000.002.00
Router(config-isis)# nsr
Router(config-isis)# log adjacency changes
Router(config-isis)# address-family ipv4 unicast
Router(config-isis-af)# metric-style wide
Router(config-isis-af)# ucmp
Router(config-isis-af)# metric 2
Router(config-isis-af)# mpls traffic-eng level-2-only

```

```

Router(config-isis-af)# mpls traffic-eng router-id Loopback0
Router(config-isis-af)# maximum-paths 5
Router(config-isis-af)# redistribute connected
Router(config-isis-af)# commit
Router(config-isis-af)# exit
!

Router# configure
Router(config)# interface Bundle-Ether3
Router(config-if)# apply-group ISIS-INTERFACE
Router(config-if)# address-family ipv4 unicast
Router(config-af)# metric 20
Router(config-af)# commit
Router(config-af)# exit
!

Router# configure
Router(config)# interface Bundle-Ether111
Router(config-if)# apply-group ISIS-INTERFACE
Router(config-if)# address-family ipv4 unicast
Router(config-af)# metric 15
Router(config-af)# commit
Router(config-af)# exit
!

/* ECMP (OSPF) */

Router# configure
Router(config)# router ospf 3
Router(config-ospf)# nsr
Router(config-ospf)# maximum paths 5
Router(config-ospf)# address-family ipv4 unicast
Router(config-ospf-af)# area 0
Router(config-ospf-af-ar)# interface Bundle-Ether3
Router(config-ospf-af-ar-if)# exit
!
Router(config-ospf-af-ar)# interface Bundle-Ether4
Router(config-ospf-af-ar-if)# exit
!
Router(config-ospf-af-ar)# interface Bundle-Ether111
Router(config-ospf-af-ar-if)# exit
!
Router(config-ospf-af-ar)# interface Bundle-Ether112
Router(config-ospf-af-ar-if)# exit
!
Router(config-ospf-af-ar)# interface Loopback23
Router(config-ospf-af-ar-if)# exit
!
Router(config-ospf-af-ar)# interface HundredGigE0/7/0/23
Router(config-ospf-af-ar-if)# commit
Router(config-ospf-af-ar-if)# exit

/* UCMP (OSPF) */

Router# configure
Router(config)# router ospf 3
Router(config-ospf)# nsr
Router(config-ospf)# maximum paths 5
Router(config-ospf)# ucmp
Router(config-ospf)# address-family ipv4 unicast
Router(config-ospf-af)# area 0
Router(config-ospf-af-ar)# interface Bundle-Ether3 cost 2

```

```

Router(config-ospf-af-ar-if) # exit
!
Router(config-ospf-af-ar) # interface Bundle-Ether4
Router(config-ospf-af-ar-if) # exit
!
Router(config-ospf-af-ar) # interface Bundle-Ether111
Router(config-ospf-af-ar-if) # exit
!
Router(config-ospf-af-ar) # interface Bundle-Ether112 cost 2
Router(config-ospf-af-ar-if) # exit
!
Router(config-ospf-af-ar) # interface Loopback23
Router(config-ospf-af-ar-if) # exit
!
Router(config-ospf-af-ar) # interface HundredGigE0/7/0/23
Router(config-ospf-af-ar-if) # commit
Router(config-ospf-af-ar-if) # exit

/* ECMP (BGP) */
Router# configure
Router(config) # router bgp 800
Router(config-bgp) # bgp bestpath as-path multipath-relax
Router(config-bgp) # address-family ipv4 unicast
Router(config-bgp-af) # network 170.170.170.3/32
Router(config-bgp-af) # network 170.170.170.10/32
Router(config-bgp-af) # network 170.170.170.11/32
Router(config-bgp-af) # network 170.170.172.3/32
Router(config-bgp-af) # network 180.180.180.9/32
Router(config-bgp-af) # network 180.180.180.20/32
Router(config-bgp-af) # network 180.180.180.21/32
Router(config-bgp-af) # network 180.180.180.24/32
Router(config-bgp-af) # network 180.180.180.25/32
Router(config-bgp-af) # commit
!
Router# configure
Router(config) # router bgp 800
Router(config-bgp) # neighbor 4.1.1.2
Router(config-bgp-nbr) # remote-as 300
Router(config-bgp-nbr) # address-family ipv4 unicast
Router(config-bgp-nbr-af) # address-family ipv4 unicast
Router(config-bgp-nbr-af) # route-policy pass-all in
Router(config-bgp-nbr-af) # route-policy pass-all out
Router(config-bgp-nbr-af) # commit
!

/* UCMP (BGP) */

Router# configure
Router(config) # router bgp 800
Router(config-bgp) # bgp bestpath as-path multipath-relax
Router(config-bgp) # address-family ipv4 unicast
Router(config-bgp-af) # maximum-paths ebgp 5
Router(config-bgp-af) # network 180.180.180.9/32
Router(config-bgp-af) # network 180.180.180.20/32
Router(config-bgp-af) # network 180.180.180.21/32
Router(config-bgp-af) # network 180.180.180.24/32
Router(config-bgp-af) # network 180.180.180.25/32
Router(config-bgp-af) # commit
!
Router# configure
Router(config) # router bgp 800
Router(config-bgp) # neighbor 7.1.5.2

```

```

Router(config-bgp-nbr)# remote-as 4000
Router(config-bgp-nbr)# address-family ipv4 unicast
Router(config-bgp-nbr-af)# address-family ipv4 unicast
Router(config-bgp-nbr-af)# route-policy TRANSITO_IN in
Router(config-bgp-nbr-af)# route-policy pass-all out
Router(config-bgp-nbr-af)# next-hop-self
Router(config-bgp-nbr-af)# commit
!
Router# configure
Router(config)# router bgp 800
Router(config-bgp)# 4.1.111.2
Router(config-bgp-nbr)# remote-as 4000
Router(config-bgp-nbr)# address-family ipv4 unicast
Router(config-bgp-nbr-af)# address-family ipv4 unicast
Router(config-bgp-nbr-af)# route-policy TRANSITO_IN in
Router(config-bgp-nbr-af)# route-policy pass-all out
Router(config-bgp-nbr-af)# next-hop-self
Router(config-bgp-nbr-af)# commit
!

/* Configure roupte policy */

Router# configure
Router(config)# route-policy TRANSITO_IN
Router(config-rpl)# if destination in (170.170.170.24/32) then
Router(config-rpl-if)# set extcommunity bandwidth (2906:1250000)
Router(config-rpl-if)# else
Router(config-rpl-else)# pass
Router(config-rpl-else)# endif
Router(config-rpl)# end-policy
!

Router# configure
Router(config)# route-policy TRANSIT1_IN
Router(config-rpl)# if destination in (170.170.170.24/32) then
Router(config-rpl-if)# set extcommunity bandwidth (2906:37500000)
Router(config-rpl-if)# else
Router(config-rpl-else)# pass
Router(config-rpl-else)# endif
Router(config-rpl)# end-policy

```

## Running Configuration

```

/* GRE Single-Pass configuration */

interface tunnel-ip30016
ipv4 address 216.1.1.1 255.255.255.0
ipv6 address 216:1:1::1/64
ipv6 enable
tunnel mode gre ipv4 encap
tunnel source Loopback22
tunnel destination 170.170.170.22
!

/* GRE Entropy(ECMP/UCMP) */

ECMP (ISIS)

router isis core
apply-group ISIS-INTERFACE

```

```

is-type level-2-only
net 49.1111.0000.0000.002.00
nsr
log adjacency changes
address-family ipv4 unicast
metric-style wide
metric 2
mpls traffic-eng level-2-only
mpls traffic-eng router-id Loopback0
maximum-paths 5
!

/* UCMP (ISIS) */

router isis core
apply-group ISIS-INTERFACE
is-type level-2-only
net 49.1111.0000.0000.002.00
nsr
log adjacency changes
address-family ipv4 unicast
metric-style wide
ucmp
metric 2
mpls traffic-eng level-2-only
mpls traffic-eng router-id Loopback0
maximum-paths 5
redistribute connected
!
interface Bundle-Ether3
apply-group ISIS-INTERFACE
address-family ipv4 unicast
metric 20
!

interface Bundle-Ether111
apply-group ISIS-INTERFACE
address-family ipv4 unicast
metric 15
!

!

/* ECMP (OSPF) */

router ospf 3
nsr
maximum paths 5
address-family ipv4 unicast
area 0
interface Bundle-Ether3
!
interface Bundle-Ether4
!
interface Bundle-Ether111
!
interface Bundle-Ether112
!
interface Loopback23
!
interface HundredGigE0/7/0/23
!
!
!
```



```
/* UCMP (OSPF) */

router ospf 3
nsr
maximum paths 5
ucmp
address-family ipv4 unicast
area 0
interface Bundle-Ether3
cost 2
!
interface Bundle-Ether4
!
interface Bundle-Ether111
!
interface Bundle-Ether112
cost 2
!
interface Loopback23
!
interface HundredGigE0/7/0/23
!
!
!

/* ECMP (BGP) */

router bgp 800
bgp bestpath as-path multipath-relax
address-family ipv4 unicast
maximum-paths ebgp 5
network 170.170.170.3/32
network 170.170.170.10/32
network 170.170.170.11/32
network 170.170.172.3/32
network 180.180.180.9/32
network 180.180.180.20/32
network 180.180.180.21/32
network 180.180.180.24/32
network 180.180.180.25/32
!
neighbor 4.1.1.2
remote-as 300
address-family ipv4 unicast
route-policy PASS-ALL in
route-policy PASS-ALL out
next-hop-self
!
!

/* UCMP (BGP) */

router bgp 800
bgp bestpath as-path multipath-relax
address-family ipv4 unicast
maximum-paths ebgp 5
network 180.180.180.9/32
network 180.180.180.20/32
network 180.180.180.21/32
network 180.180.180.24/32
network 180.180.180.25/32
!

neighbor 7.1.5.2
```

```

remote-as 4000
address-family ipv4 unicast
route-policy TRANSIT0_IN in
route-policy PASS-ALL out
next-hop-self
!
!
neighbor 4.1.111.2
remote-as 4000
address-family ipv4 unicast
route-policy TRANSIT1_IN in
route-policy PASS-ALL out
next-hop-self
!
!

/* Configure rouverte policy */

route-policy TRANSIT0_IN
if destination in (170.170.170.24/32) then
set extcommunity bandwidth (2906:1250000)
else
pass
endif
end-policy
!
route-policy TRANSIT1_IN
if destination in (170.170.170.24/32) then
set extcommunity bandwidth (2906:37500000)
else
pass
endif
end-policy
!

```

## Verification

Verify if the tunnel mode GRE encapsulation is enabled.

```
Router# show int tunnel-ip2
```

```

interface tunnel-ip2
ipv4 address 80.80.82.1 255.255.255.0
ipv6 address 2000:80:80:82::1/64
load-interval 30
tunnel mode gre ipv4 encap
tunnel source Loopback4
tunnel destination 11.4.2.2
!

```

```

RP/0/RP0/CPU0:PE1_5516#show int tunnel-ip2
tunnel-ip2 is up, line protocol is up
 Interface state transitions: 1
 Hardware is Tunnel
 Internet address is 80.80.82.1/24
 MTU 1500 bytes, BW 100 Kbit (Max: 100 Kbit)
 reliability 255/255, txload 0/255, rxload 0/255
 Encapsulation TUNNEL_IP, loopback not set,
 Last link flapped 1d18h
 Tunnel TOS 0
 Tunnel mode GRE IPV4, encap
 Keepalive is disabled.
 Tunnel source 11.11.12.1 (Loopback4), destination 11.4.2.2/32
 Tunnel TTL 255

```

```

Last input never, output never
Last clearing of "show interface" counters 14:53:37
30 second input rate 0 bits/sec, 0 packets/sec
30 second 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

```

Verify if the tunnel mode GRE encapsulation and decapsulation are enabled.

```

Router# sh interfaces tunnel-ip 5 accounting
Wed May 16 01:50:57.258 UTC
tunnel-ip5
 Protocol Pkts In Chars In Pkts Out Chars Out
 IPV4_UNICAST 489 55746 0 0
 IPV6_UNICAST 489 55746 0 0
 MPLS 587 69266 0 0

```

Verify if the recycle of the packets are not done under Recycle VoQ: 48:

```

Router# show tunnel ip ea summary location 0/7/CPU0

Number of tunnel updates to retry: 0
Number of tunnel updates retried: 0
Number of tunnel retries failed: 0
Platform:
Recycle VoQ: 48

```

|         | ReceivedBytes | ReceivedPackets | ReceivedKbps |
|---------|---------------|-----------------|--------------|
|         | DroppedBytes  | DroppedPackets  | DroppedKbps  |
| NPU 0:0 | 0             | 0               | 0            |
|         | 0             | 0               | 0            |
| 1       | 0             | 0               | 0            |
|         | 0             | 0               | 0            |
| 2       | 0             | 0               | 0            |
|         | 0             | 0               | 0            |
| 3       | 0             | 0               | 0            |
|         | 0             | 0               | 0            |
| ...     |               |                 |              |
| NPU 1:0 | 0             | 0               | 0            |
|         | 0             | 0               | 0            |
| 1       | 0             | 0               | 0            |
|         | 0             | 0               | 0            |
| 2       | 0             | 0               | 0            |
|         | 0             | 0               | 0            |
| 3       | 0             | 0               | 0            |
|         | 0             | 0               | 0            |
| NPU 2:0 | 0             | 0               | 0            |
|         | 0             | 0               | 0            |
| 1       | 0             | 0               | 0            |
|         | 0             | 0               | 0            |
| 2       | 0             | 0               | 0            |
|         | 0             | 0               | 0            |
| 3       | 0             | 0               | 0            |
|         | 0             | 0               | 0            |

Verify if the tunnel mode GRE encapsulation is enabled.

```

Router# show interfaces tunnel-ip * brief

Thu Sep 7 00:04:39.125 PDT
Intf Intf LineP Encap MTU BW

```

| Name    | State | State | Type      | (byte) | (Kbps) |
|---------|-------|-------|-----------|--------|--------|
| ti30001 | down  | down  | TUNNEL_IP | 1500   | 100    |
| ti30002 | up    | up    | TUNNEL_IP | 1500   | 100    |

Verify the tunnel endpoint route in RIB.

```
Router# show route 10.1.1.1
```

```
Routing entry for 10.0.0.0/8
Known via "static", distance 1, metric 0 (connected)
Installed Oct 2 15:50:56.755 for 00:39:24
Routing Descriptor Blocks
directly connected, via tunnel-ip109
Route metric is 0, Wt is 1
No advertising protos.
```

Verify if the tunnel mode GRE encapsulation is enabled.

```
Router# show tunnel ip ea database tunnel-ip 109 location 0/7/CPU0
```

```
----- node0_0_CPU0 -----
tunnel ifhandle 0x80022cc
tunnel source 161.115.1.2
tunnel destination 162.1.1.1/32
tunnel transport vrf table id 0xe0000000
tunnel mode gre ipv4, encap
tunnel bandwidth 100 kbps
tunnel platform id 0x0
tunnel flags 0x40003400
IntfStateUp
BcStateUp
Ipv4Caps
Encap
tunnel mtu 1500
tunnel tos 0
tunnel ttl 255
tunnel adjacency flags 0x1
tunnel o/p interface handle 0x0
tunnel key 0x0, entropy length 0 (mask 0xffffffff)
tunnel QT next 0x0
tunnel platform data (nil)
Platform:
Handle: (nil)
Decap ID: 0
Decap RIF: 0
Decap Recycle Encap ID: 0x00000000
Encap RIF: 0
Encap Recycle Encap ID: 0x00000000
Encap IPv4 Encap ID: 0x4001381b
Encap IPv6 Encap ID: 0x00000000
Encap MPLS Encap ID: 0x00000000
DecFEC DecRcyLIF DecStatsId EncRcyLIF
```

Verify if the QoS table is updated properly.

```
Router# show controllers npu stats voq base 48 instance all location
```

```
0/0/CPU0
```

```
Asic Instance = 0
```

```
VOQ Base = 48
```

|        | ReceivedPkts | ReceivedBytes | DroppedPkts | DroppedBytes |
|--------|--------------|---------------|-------------|--------------|
| COS0 = | 0            | 0             | 0           | 0            |
| COS1 = | 0            | 0             | 0           | 0            |
| COS2 = | 0            | 0             | 0           | 0            |
| COS3 = | 0            | 0             | 0           | 0            |

```
Asic Instance = 1
VOQ Base = 48
 ReceivedPkts ReceivedBytes DroppedPkts DroppedBytes

COS0 = 0 0 0 0
COS1 = 0 0 0 0
COS2 = 0 0 0 0
COS3 = 0 0 0 0

Asic Instance = 2
VOQ Base = 48
 ReceivedPkts ReceivedBytes DroppedPkts DroppedBytes

COS0 = 0 0 0 0
COS1 = 0 0 0 0
COS2 = 0 0 0 0
COS3 = 0 0 0 0
```





# CHAPTER 11

## Configuring IP-in-IP Tunnels

This chapter provides conceptual and configuration information for IP-in-IP tunnels.

### IP-in-IP Tunnels

*Table 23: Feature History Table*

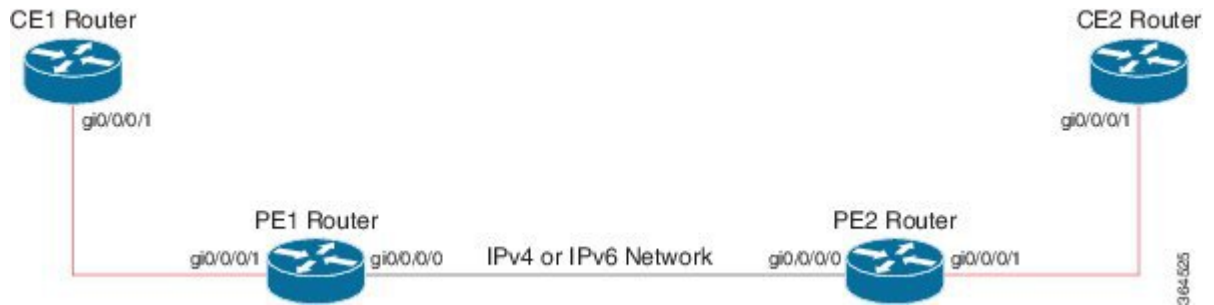
| Feature Name                                      | Release Information | Feature Description                                                                                                                                                                                                                                                                                                                                                                                                              |
|---------------------------------------------------|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Decapsulating IPv4 packets with IPv6 Outer Header | Release 7.5.4       | <p>With this release, decapsulation of IPv4 and IPv6 packets with IPv6 outer headers are supported. This decapsulation is supported only with <b>tunnel source direct</b> option and not with <b>tunnel source</b> with IPv6 address.</p> <p>This feature helps the administrators to take advantage of the benefits of IPv6, such as improved routing and security, without having to upgrade their entire network to IPv6.</p> |

Tunneling provides a mechanism to transport packets of one protocol within another protocol. IP-in-IP tunneling refers to the encapsulation and decapsulation of an IP packet as a payload in another IP packet. Cisco NCS 5500 Routers support IP-in-IP decapsulation with all possible combinations of IPv4 and IPv6; that is, IPv4 over IPv4, IPv6 over IPv4, IPv4 over IPv6, and IPv6 over IPv6. For example, an IPv4 over IPv6 refers to an IPv4 packet as a payload encapsulated within an IPv6 packet and routed across an IPv6 network to reach the destination IPv4 network, where it is decapsulated.

IP-in-IP tunneling can be used to connect remote networks securely or provide virtual private network (VPN) services.

The following example provides configurations for an IPv4 or IPv6 tunnel, with the transport VRF as the default VRF for the following simplified network topology.

Figure 16: IP-in-IP Tunnel Network Topology



## Configuration Example for IPv4 Tunnel

| PE1 Router Configuration                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | PE2 Router Configuration                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <pre>interface GigabitEthernet0/0/0/0 !! Link between PE1-PE2 ipv4 address 100.1.1.1/64 ! interface GigabitEthernet0/0/0/1 !! Link between CE1-PE1 ipv4 address 20.1.1.1/24 ipv6 address 20::1/64 ! interface tunnel-ip 1 ipv4 address 10.1.1.1/24 ipv6 address 10::1/64 tunnel mode ipv4 tunnel source GigabitEthernet0/0/0/0 tunnel destination 100.1.1.2 !  router static address-family ipv4 unicast  30.1.1.0/24 tunnel-ip1 address-family ipv6 unicast  30::0/64 tunnel-ip1 ! !</pre> | <pre>interface GigabitEthernet0/0/0/0 !! Link between PE1-PE2 ipv4 address 100.1.1.2/64 ! interface GigabitEthernet0/0/0/1 !! Link between PE2-CE2 ipv4 address 30.1.1.1/24 ipv6 address 30::1/64 ! interface tunnel-ip 1 ipv4 address 10.1.1.2/24 ipv6 address 10::2/64 tunnel mode ipv4 tunnel source GigabitEthernet0/0/0/0 tunnel destination 100.1.1.1 !  router static address-family ipv4 unicast  20.1.1.0/24 tunnel-ip1 address-family ipv6 unicast  20::0/64 tunnel-ip1 ! !</pre> |
| CE1 Router Configuration                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | CE2 Router Configuration                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <pre>interface GigabitEthernet0/0/0/1 !! Link between CE1-PE1 ipv4 address 20.1.1.2 255.255.255.0 ipv6 address 20::2/64 ! router static address-family ipv4 unicast  30.1.1.0/24 20.1.1.1 address-family ipv6 unicast  30::0/64 20::1 ! !</pre>                                                                                                                                                                                                                                             | <pre>interface GigabitEthernet0/0/0/1 !! Link between CE2-PE2 ipv4 address 30.1.1.2 255.255.255.0 ipv6 address 30::2/64 ! router static address-family ipv4 unicast  20.1.1.0/24 30.1.1.1 address-family ipv6 unicast  20::0/64 30::1 ! !</pre>                                                                                                                                                                                                                                             |



**Configuration Example for IPv6 Tunnel**

| PE1 Router Configuration                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | PE2 Router Configuration                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <pre> interface GigabitEthernet0/0/0/0 !! Link between PE1-PE2 ipv6 address 100::1/64 ! interface GigabitEthernet0/0/0/1 !! Link between CE1-PE1 vrf RED ipv4 address 20.1.1.1/24 ipv6 address 20::1/64 ! interface tunnel-ip 1 vrf RED ipv4 address 10.1.1.1/24 ipv6 address 10::1/64 tunnel mode ipv6 tunnel source GigabitEthernet0/0/0/0 tunnel destination 100::2 ! vrf RED address-family ipv6 unicast import route-target 2:1 ! export route-target 2:1 ! address-family ipv4 unicast import route-target 2:1 ! export route-target 2:1 ! router static vrf RED address-family ipv4 unicast 30.1.1.0/24 tunnel-ip1 address-family ipv6 unicast 30::0/64 tunnel-ip1 ! ! ! </pre> | <pre> interface GigabitEthernet0/0/0/0 !! Link between PE1-PE2 ipv6 address 100::2/64 ! interface GigabitEthernet0/0/0/1 !! Link between PE2-CE2 vrf RED ipv4 address 30.1.1.1/24 ipv6 address 30::1/64 ! interface tunnel-ip 1 vrf RED ipv4 address 10.1.1.2/24 ipv6 address 10::2/64 tunnel mode ipv6 tunnel source GigabitEthernet0/0/0/0 tunnel destination 100::1 ! vrf RED address-family ipv6 unicast import route-target 2:1 ! export route-target 2:1 ! address-family ipv4 unicast import route-target 2:1 ! export route-target 2:1 ! router static vrf RED address-family ipv4 unicast 20.1.1.0/24 tunnel-ip1 address-family ipv6 unicast 20::0/64 tunnel-ip1 ! ! ! </pre> |
| CE1 Router Configuration                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | CE2 Router Configuration                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <pre> interface GigabitEthernet0/0/0/1 !! Link between CE1-PE1 ipv4 address 20.1.1.2 255.255.255.0 ipv6 address 20::2/64 ! router static address-family ipv4 unicast 30.1.1.0/24 20.1.1.1 address-family ipv6 unicast 30::0/64 20::1 ! ! </pre>                                                                                                                                                                                                                                                                                                                                                                                                                                        | <pre> interface GigabitEthernet0/0/0/1 !! Link between CE2-PE2 ipv4 address 30.1.1.2 255.255.255.0 ipv6 address 30::2/64 ! router static address-family ipv4 unicast 20.1.1.0/24 30.1.1.1 address-family ipv6 unicast 20::0/64 30::1 ! ! </pre>                                                                                                                                                                                                                                                                                                                                                                                                                                        |

- [IP-in-IP Decapsulation, on page 258](#)

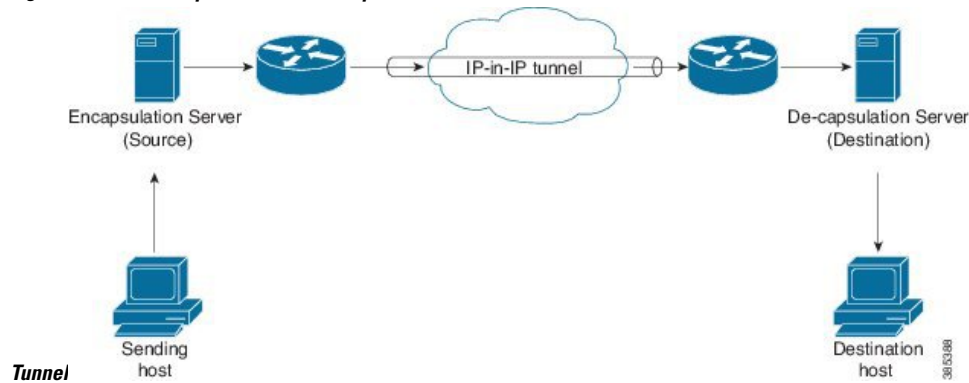
# IP-in-IP Decapsulation

Encapsulation of datagrams in a network is done for multiple reasons, such as when a source server wants to influence the route that a packet takes to reach the destination host. The source server is also known as the encapsulation server.

IP-in-IP encapsulation involves the insertion of an outer IP header over the existing IP header. The source and destination address in the outer IP header point to the endpoints of the IP-in-IP tunnel. The stack of IP headers is used to direct the packet over a predetermined path to the destination, provided the network administrator knows the loopback addresses of the routers transporting the packet. This tunneling mechanism can be used for determining availability and latency for most network architectures. It is to be noted that the entire path from source to the destination does not have to be included in the headers, but a segment of the network can be chosen for directing the packets.

The following illustration describes the basic IP-in-IP encapsulation and decapsulation model.

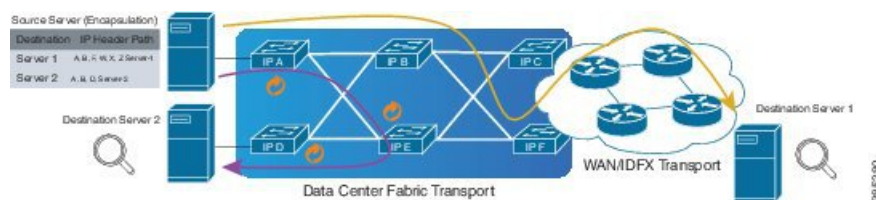
**Figure 17: Basic Encapsulation and Decapsulation with an IP-in-IP**



## Use Case: Configure IP-in-IP Decapsulation

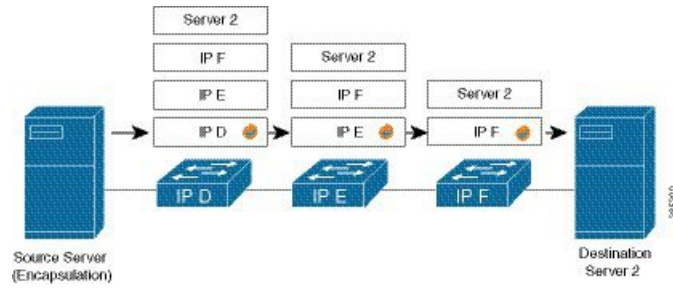
The following topology describes a use case where IP-in-IP encapsulation and decapsulation are used for different segments of the network from source to destination. The IP-in-IP tunnel consists of multiple routers that are used to decapsulate and direct the packet through the data center fabric network.

**Figure 18: IP-in-IP Decapsulation Through a Data Center Network**



The following illustration shows how the stacked IPv4 headers are de-capsulated as they traverse through the de-capsulating routers.

**Figure 19: IP Header Decapsulation**



**Stacked IP Header in an Encapsulated Packet**

The encapsulated packet has an outer IPv4 header that is stacked over the original IPv4 header, as shown in the following illustration.

## Encapsulated Packet

|                       |                    |
|-----------------------|--------------------|
| [-] Frame             |                    |
| [-] EthernetII        |                    |
| Preamble (hex)        | fb555555555555d5   |
| Destination MAC       | 62:19:88:64:E2:68  |
| Source MAC            | 00:10:94:00:00:02  |
| EtherType (hex)       | <auto> Internet IP |
| [-] IPv4 Header       |                    |
| Version (int)         | <auto> 4           |
| Header length (int)   | <auto> 5           |
| ToS/DiffServ          | tos (0x00)         |
| Total length (int)    | <auto> calculated  |
| Identification (int)  | 0                  |
| [-] Control Flags     |                    |
| Reserved (bit)        | 0                  |
| DF Bit (bit)          | 0                  |
| MF Bit (bit)          | 0                  |
| Fragment Offset (int) | 0                  |
| Time to live (int)    | 255                |
| Protocol (int)        | <auto> IP          |
| Checksum (int)        | <auto> 33492       |
| Source                | 192.xx.xx.xx       |
| Destination           | 127.0.0.1          |
| Header Options        |                    |
| Gateway               | 192.0.2.10         |
| [-] IPv4 Header       |                    |
| Version (int)         | <auto> 4           |
| Header length (int)   | <auto> 5           |
| ToS/DiffServ          | tos (0x00)         |
| Total length (int)    | <auto> calculated  |
| Identification (int)  | 0                  |
| [-] Control Flags     |                    |
| Reserved (bit)        | 0                  |

385413

## Configuration

You can use the following sample configuration on the routers to decapsulate the packet as it traverses the IP-in-IP tunnel:

```
RP/0/RP0/CPU0:router(config)# interface tunnel-ip 10
RP/0/RP0/CPU0:router(config-if)# tunnel mode ipv4 decap
RP/0/RP0/CPU0:router(config-if)# tunnel source loopback 0
RP/0/RP0/CPU0:router(config-if)# tunnel destination 10.10.1.2/32
```

- **tunnel-ip**: configures an IP-in-IP tunnel interface.

- **ipv4 unnumbered loopback address**: enables ipv4 packet processing without an explicit address, except for loopback address.
- **tunnel mode ipv4 decap**: enables IP-in-IP decapsulation.
- **tunnel source**: indicates the source address for the IP-in-IP decap tunnel w.r.t the router interface.
- **tunnel destination**: indicates the destination address for the IP-in-IP decap tunnel w.r.t the router interface.

### Running Configuration

```
RP/0/RP0/CPU0:router# show running-config interface tunnel-ip 10
...
interface tunnel-ip 10
 tunnel mode ipv4 decap
 tunnel source Loopback 0
 tunnel destination 10.10.1.2/32
```

This completes the configuration of IP-in-IP decapsulation.

## Decapsulation Using Tunnel Source Direct

Table 24: Feature History Table

| Feature Name                             | Release Information | Feature Description                                                                                                                                                                                                                                                                                           |
|------------------------------------------|---------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Decapsulating Using Tunnel Source Direct | Release 7.5.3       | Tunnel source direct allows you to decapsulate the tunnels on any L3 interface on the router.<br><br>You can use the <b>tunnel source direct</b> configuration command to choose the specific IP Equal-Cost Multipath (ECMP) links for troubleshooting, when there are multiple IP links between two devices. |

To debug faults in various large networks, you may have to capture and analyze the network traffic at a packet level. In datacenter networks, administrators face problems with the volume of traffic and diversity of faults. To troubleshoot faults in a timely manner, DCN administrators must identify affected packets inside large volumes of traffic. They must track them across multiple network components, analyze traffic traces for fault patterns, and test or confirm potential causes.

In some networks, IP-in-IP decapsulation is currently used in network management, to verify ECMP availability and to measure the latency of each path within a datacenter.

The Network Management System (NMS) sends IP-in-IP (IPv4 or IPv6) packets with a stack (multiple) of predefined IPv4 or IPv6 headers (device IP addresses). The destination device at each hop removes the outside header, performs a lookup on the next header, and forwards the packets if a route exists.

Using the **tunnel source direct** command, you can choose the specific IP Equal-Cost Multipath (ECMP) links for troubleshooting, when there are multiple IP links between two devices.



**Tip** You can programmatically configure and manage the Ethernet interfaces using `openconfig-ethernet-if.yang` and `openconfig-interfaces.yang` OpenConfig data models. To get started with using data models, see the *Programmability Configuration Guide for Cisco NCS 5500 Series Routers*.

## Guidelines and Limitations

The following guidelines are applicable to this feature.

- The **tunnel source direct** command is only compatible with 'tunnel mode decap' for IP-in-IP decapsulation.
- The source-direct tunnel is always operationally `up` unless it is administratively shut down. The directly connected interfaces are identified using the **show ip route direct** command.
- All Layer 3 interfaces that are configured on the device are supported.
- Platform can accept and program only certain number of IP addresses. The number of IP addresses depends on the make of the platform linecard (LC). Each LC can have different number of Network Processor (NP) slices and interfaces.
- Only one source-direct tunnel per address-family is supported for configuration.
- Regular decapsulation tunnels which have specific source address, are supported. However, the tunnel's specific source address must not be part of any interface.

The following functionalities are not supported for the **tunnel source direct** option.

- GRE tunneling mode.
- VRF (only default VRF is supported).
- ACL and QoS on the tunnels.
- Tunnel encapsulation.
- Tunnel NetIO DLL: Decapsulation is not supported if the packet is punted to slow path.

## Configure Decapsulation Using Tunnel Source Direct

### Configuration

The **tunnel source direct** configures IP-in-IP tunnel decapsulation on any directly connected IP addresses. This option is now supported only when the IP-in-IP decapsulation is used to source route the packets through the network.

This example shows how to configure IP-in-IP tunnel decapsulation on directly connected IP addresses:

```
Router# configure terminal
Router(config)#interface Tunnel4
 Router(config)#tunnel mode ipv4 decap
 Router(config)#tunnel source direct
 Router(config)#no shutdown
```

This example shows how to configure IP-in-IP tunnel decapsulation on IPv6 enabled networks:

```
Router# configure terminal
Router(config)#interface Tunnel6
```

```
Router(config)#tunnel mode ipv6 decap
Router(config)#tunnel source direct
Router(config)#no shutdown
```

### Verifying the Configuration

The following example shows how to verify IP-in-IP tunnel decapsulation with **tunnel source direct** option:

```
Router#show running-config interface tunnel 1
interface Tunnell
 tunnel mode ipv6ipv6 decapsulate-any
 tunnel source direct
 no shutdown

Router#show interface tunnel 1
Tunnell is up Admin State: up
MTU 1460 bytes, BW 9 Kbit
Tunnel protocol/transport IPv6/DECAPANY/IPv6
Tunnel source - direct
Tx 0 packets output, 0 bytes Rx 0 packets input, 0 bytes
```







# CHAPTER 12

## Understand Generic UDP Encapsulation

UDP encapsulation is a technique of adding network headers to the packets and then encapsulating the packets within the User Datagram Protocol (UDP).

Encapsulating packets using UDP facilitates efficient transport across networks. By leveraging Receive Side Scaling (RSS) and Equal Cost Multipath (ECMP) routing, UDP provides significant performance benefits for load-balancing. The use of the UDP source port provides entropy to ECMP hashing and provides the ability to use the IP source or destination, and the L4 Port for load-balancing entropy.

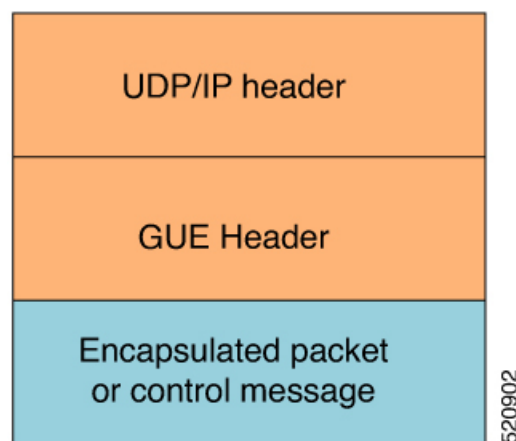
Traditional mechanisms like Generic Routing Encapsulation (GRE) can only handle the outer Source IP address and parts of the destination address and may not provide sufficient load balance entropy.

Generic UDP Encapsulation (GUE) is a UDP-based network encapsulation protocol that encapsulates IPv4 and IPv6 packets. GUE provides native UDP encapsulation and defines an additional header, that helps to determine the payload carried by the IP packet. The additional header can include items such as a virtual networking identifier, security data for validating or authenticating the GUE header, congestion control data, and so on.

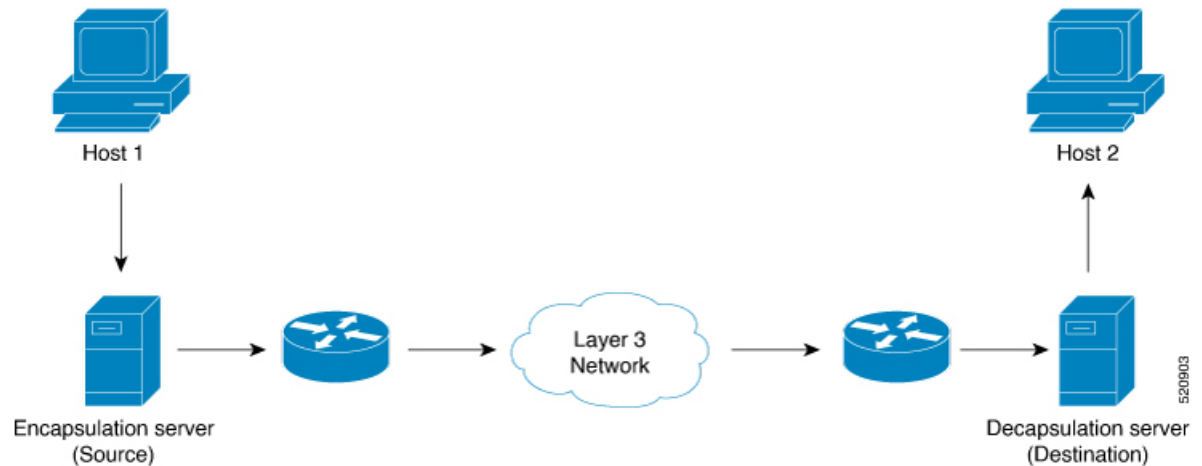
In GUE, the payload is encapsulated in an IP packet that can be IPv4 or IPv6 Carrier. The UDP header is added to provide extra hashing parameters, and optional payload demultiplexing. At the decapsulation node, the Carrier IP and UDP headers are removed, and the packet is forwarded based on the inner payload.

A GUE packet has the general format:

**Figure 20: GUE Packet Format**



For example, if the data stream is sent from Host 1 to Host 2. The server acts as a GUE encapsulator that is sending the packets from Host 1. The server, on the other end receiving the data, validates the data for the valid carrier IP and UDP header and decapsulates the data.



GUE has various variants, but variant 1 of GUE allows direct encapsulation of IPv4 and IPv6 in UDP. This technique saves encapsulation overhead on links for the use of IP encapsulation, and also need not allocate a separate UDP port number for IP-over-UDP encapsulation.

Variant 1 has no GUE header, but a UDP packet carries an IP packet. The first two bits of the UDP payload is the GUE variant field and match with the first 2 bits of the version number in the IP header.

### Benefits of using GUE

- Allows direct encapsulation of payloads like IPv4 and IPv6 in the UDP packet.
  - You can use UDP port for demultiplexing payloads.
  - You can use a single UDP port allowing systems to employ parsing models to identify payloads.
- Leverages the UDP header for entropy labels by encoding a tuple-based source port.
- Leverages source IP addresses for load-balance encoding. Destination also could be terminated based on a subnet providing additional bits for entropy.
- Avoids special handling for transit nodes because they only see an IP-UDP packet with some payload..
- Eases implementation of UDP tunneling with GUE. This is because of the direct encapsulation method of the payloads into UDP.
- [Restrictions, on page 266](#)
- [Configure GUE, on page 267](#)
- [Flexible Assignment of UDP Port Numbers for Decapsulation, on page 269](#)

## Restrictions

- Supports Generic UDP Decapsulation for variant 1 only.
- Receives IPv4 packets with the defined GUE port of 6080.

- Decapsulates IPv6 packets with the defined GUE port of 6615.
- Receives MPLS packets with the UDPoMPLS port of 6635
- Range of source or destination ports is not supported.
- Range, Source, or Destination addresses are not supported, but subnet mask entries are allowed.
- Destination Port is mandatory to perform decapsulation.
- Terminating GRE after GUE or GUE after GRE is not supported.
- Terminating a label such as a VPN Deaggregation after GUE termination is not supported.
- Slow path support is not supported. To resolve the inner IP Adjacency, use the **cef proactive-arp-nd enable** command.
- Running the **clear all** command doesn't clear the interface of all its existing configurations.

## Configure GUE

Use the following configuration work flow to configure GUE, which is required to decode an incoming GUE packet on router:

1. Configure a traffic class: Create a traffic class and specify various criteria for classifying packets using the match commands, and an instruction on how to evaluate these match commands.
2. Configure a policy map: Define a policy map and associate the traffic class with the traffic policy.
3. Apply the policy per VRF basis, and apply this policy on all the interfaces that are part of the VRF.

### Configuration Example

1. Configure a traffic class:

```
Router# configure
Router(config)# class-map type traffic match-all gre-1
Router(config-cmap)# match destination-address ipv4 225.100.20.0 255.255.255.0
Router(config-cmap)# match protocol gre
Router(config-cmap)# end-class-map
Router(config)# commit

Router(config)# class-map type traffic match-all udp-v4
Router(config-cmap)# match destination-address ipv4 220.100.20.0 255.255.255.0
Router(config-cmap)# match source-address ipv4 210.100.20.0 255.255.255.0
Router(config-cmap)# match protocol udp
Router(config-cmap)# match destination-port 6080
Router(config-cmap)# end-class-map
Router(config)# commit

Router(config)# class-map type traffic match-all udp-mpls1
Router(config-cmap)# match destination-address ipv4 220.100.20.0 255.255.255.0
Router(config-cmap)# match source-address ipv4 210.100.20.0 255.255.255.0
Router(config-cmap)# match destination-port 6635
Router(config-cmap)# end-class-map
Router(config)# commit
```

```

Router(config)# class-map type traffic match-all udp-v6
Router(config-cmap)# match destination-address ipv4 220.100.20.0 255.255.255.0
Router(config-cmap)# match source-address ipv4 210.100.20.0 255.255.255.0
Router(config-cmap)# match protocol udp
Router(config-cmap)# match destination-port 6615
Router(config-cmap)# end-class-map
Router(config)# commit

```

2. Define a policy map and associate the traffic class with the traffic policy:

```

Router(config)# policy-map type pbr magic-decap
Router(config-pmap)# class type traffic gre-1
Router(config-pmap-c)# decapsulate gre
Router(config-pmap-c)# exit

Router(config-pmap)# class type traffic udp-v4
Router(config-pmap-c)# decapsulate gue variant 1
Router(config-pmap-c)# exit

Router(config-pmap)# class type traffic udp-v6
Router(config-pmap-c)# decapsulate gue variant 1
Router(config-pmap-c)# exit
!
Router(config-pmap)# class type traffic udp-mpls1
Router(config-pmap-c)# decapsulate gue variant 1
Router(config-pmap-c)# exit

Router(config-pmap)# class type traffic class-default
Router(config-pmap-c)# exit

Router(config-pmap)# end-policy-map
Router(config)# commit
Router(config)# exit

```

3. Apply the policy per VRF basis:

```

Router# configure
Router(config)# vrf-policy
Router(config-vrf-policy)# vrf default address-family ipv4 policy type pbr input magic-decap
Router(config-vrf-policy)# commit

```

### Configure Generic UDP Decapsulation for Load Balancing

On transit routers, the outer IP for hashing is used to encode the entropy parameters. But at the terminating or decapsulating router, the payload is used for hashing. However, you can use the outer IP at the decapsulating router as well, as payloads may have limited entropy. To enable the outer IP based hashing on the decapsulation router, use this command:

```

Router(config)# hw-module profile load-balance algorithm ip-tunnel
Router(config)# commit

```




---

**Note** Unlike other **hw-module** commands, the **hw-module profile load-balance algorithm ip-tunnel** command requires a reload of the system.

---

# Flexible Assignment of UDP Port Numbers for Decapsulation

Table 25: Feature History Table

| Feature Name                                              | Release Information | Feature Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|-----------------------------------------------------------|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Flexible Assignment of UDP Port Numbers for Decapsulation | Release 7.3.3       | <p>This feature gives you the flexibility to assign UDP port numbers from 1000 through 6400, through which IPv4, IPv6, and MPLS packets can be decapsulated. Such flexibility allows you to segregate the ingress traffic based on a QoS policy.</p> <p>In earlier releases, you could assign only default ports for decapsulation.</p> <p>The following command is introduced for this feature:</p> <pre>hw-module profile gue udp-dest-port ipv4 &lt;port number&gt; ipv6 &lt;port number&gt; mpls &lt;port number&gt;</pre> |

This feature provides decapsulation support for GUE packets. In GUE, the payload is encapsulated in an IP packet—IPv4 or IPv6 carrier. The UDP header is added to provide extra hashing parameters and optional payload demultiplexing. At the decapsulation node, the carrier IP and UDP headers are removed, and the packet is forwarded based on the inner payload. Prior to Release 7.3.3, packets were decapsulated using UDP port numbers 6080, 6615, and 6635 for IPv4, IPv6, and MPLS payloads respectively. Starting from Release 7.3.3, you can assign UDP port numbers from 1000 through 64000 to decapsulate IPv4, IPv6, and MPLS packets.

## Guidelines for Setting up Decapsulation Using Flexible Port Numbers

Apply these guidelines while assigning flexible port numbers for decapsulation:

| Packet                     | IPv4                                                                                 | IPv6                                                             | MPLS                                                             |
|----------------------------|--------------------------------------------------------------------------------------|------------------------------------------------------------------|------------------------------------------------------------------|
| UDP Outer Header           | Configure IPv4 port on the hardware module.                                          | Configure IPv6 port on the hardware module.                      | Configure MPLS port on the hardware module.                      |
| Encapsulation Outer Header | Configure an IPv4 encapsulation outer header that matches with the class map source. |                                                                  |                                                                  |
| Inner Payload              | Note that packets are forwarded based on the inner IPv4 payload.                     | Note that packets are forwarded based on the inner IPv6 payload. | Note that packets are forwarded based on the inner MPLS payload. |



- Note**
- During the decapsulation of the IPv4, IPv6, and MPLS packets, the following headers are removed:
    - The UDP outer header
    - The IPv4 encapsulation outer header
  - Select different values for each of these protocols. Valid port numbers are from 1000 through 64000.

## Outer-Header Hashing Support for IPoGREoGUE and MPLSoGREoUDP Flows

Table 26: Feature History Table

| Feature Name                                                        | Release Information | Feature Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|---------------------------------------------------------------------|---------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Outer-header hashing support for IPoGREoGUE and MPLSoGREoUDP flows. | Release 7.5.3       | <p>This feature specifies the hashing only on outer IP (L3 and L4) headers for IPoGREoGUE and MPLSoGREoUDP flows.</p> <p>You must enable <b>ip-tunnel</b> mode for GUE decapsulation.</p> <p>This feature enables load-balancing control across the L3 and L4 headers and allows full utilization of the paths.</p> <p>Example for the two flows:</p> <ul style="list-style-type: none"> <li>• Eth + Ipv4 + UDP + IPv4 + GRE + IPv4 + UDP or TCP</li> <li>• Eth + Ipv4 + UDP + IPv4 + GRE + MPLS + IPv4 + UDP or TCP</li> </ul> |

- GUE transit functionality, wherein the router forwards the traffic, works in both default and **ip-tunnel** mode.
- When the **ip-tunnel** mode is enabled for load balancing, the “outer” header is used for hashing. This includes the outer IP SRC/DST and the outer UDP SRC port.
- For more information on hashing, refer [Understand Generic UDP Encapsulation](#).
- For configuring **ip-tunnel** mode, use the following command:

```
Router(config)# hw-module profile load-balance algorithm ip-tunnel
Router(config)# commit
```



**Note** Unlike other **hw-module** commands, the **hw-module profile load-balance algorithm ip-tunnel** command, requires a reload of the system.

### Restrictions

- GUE transit functionality is supported in both default and **ip-tunnel** mode.
- GUE decapsulation functionality is only supported with **ip-tunnel** mode.
- GUE decapsulation is supported only for IPv4 carrier packets.  
Example: Outer header as IPv4 and UDP.
- GUE decapsulation is not supported for IPv6 and MPLS carrier packets.  
Example: Outer header as IPv6 and UDP.

## Running Configuration

For configuration steps see, [Configure GUE](#).

```
Router# show running-config class-map
class-map type traffic match-all gre-1
match destination-address ipv4 225.100.20.0 255.255.255.0
match protocol gre
end-class-map
!
class-map type traffic match-all udp-v4
match destination-address ipv4 220.100.20.0 255.255.255.0
match source-address ipv4 210.100.20.0 255.255.255.0
match protocol udp
match destination-port 6080
end-class-map
!
class-map type traffic match-all udp-v6
match destination-address ipv4 220.100.20.0 255.255.255.0
match source-address ipv4 210.100.20.0 255.255.255.0
match protocol udp
match destination-port 6615
end-class-map
!
class-map type traffic match-all gue_ipv4
match destination-address ipv4 120.0.0.0 255.255.0.0
match source-address ipv4 96.0.0.0 224.0.0.0
match protocol udp
match destination-port 6080
end-class-map
!
class-map type traffic match-all udp-mpls1
match destination-address ipv4 220.100.20.0 255.255.255.0
match source-address ipv4 210.100.20.0 255.255.255.0
match protocol udp
match destination-port 6635
end-class-map
!
Router# show running-config policy-map
policy-map type pbr magic-decap
```

```
class type traffic gre-1
 decapsulate gre
!
class type traffic udp-v4
 decapsulate gue variant 1
!
class type traffic udp-v6
 decapsulate gue variant 1
!
class type traffic udp-mpls1
 decapsulate gue variant 1
!
class type traffic class-default
!
end-policy-map
!

Router# show running-config vrf-policy
vrf-policy
vrf default address-family ipv4 policy type pbr input magic-decap
!
```

## Verification

Run the **show ofa objects sys location 0/0/CPU0 | inc gue** command in the XR Config mode to verify that the unique GUE port numbers have been configured to decapsulate IPv4, IPv6, and MPLS payloads.

```
Router#show ofa objects sys location 0/0/CPU0 | inc gue
uint32_t gue_ipv4_port => 1001
uint32_t gue_ipv6_port => 1002
uint32_t gue_mpls_port => 1003
```





## CHAPTER 13

# Configuring 400G Digital Coherent Optics

The following 400G Digital Coherent QSFP-DD optical modules are supported:

- QDD-400G-ZR-S
- QDD-400G-ZRP-S



### Note

- The information in this chapter applies to all supported 400G Digital Coherent QSFP-DD optical modules unless otherwise specified.
- To determine the transceivers that Cisco hardware device supports, refer to the [Transceiver Module Group \(TMG\) Compatibility Matrix](#) tool.

This chapter describes the 400G Digital Coherent QSFP-DD optical modules and their supported configurations.

The 400G Digital Coherent QSFP-DD optical modules enable wavelength-division multiplexing (WDM) functionality in the router. These optical modules are DWDM C-band (196.1 THz to 191.3 THz) tunable optical modules. They can be used in both transponder and muxponder modes.

Cisco IOS XR software creates optics and coherent DSP controllers to configure and monitor the performance of the 400G Digital Coherent QSFP-DD optical modules. Optics controllers are used to configure and monitor optical parameters, such as frequency, chromatic dispersion, transmitted output power, modulation, and so on. Coherent DSP controllers are used to monitor network performance parameters like pre- and post-forward error correction (FEC) bit-error rate (pre-FEC BER, post-FEC BER), error corrected bits (EC-BITS), and so on. Forward error correction (FEC) is configured using optical controllers and monitored using coherent DSP controllers.

The 400G Digital Coherent QSFP-DD optical modules support traffic configuration and firmware download. The Cisco IOS XR software collects performance monitoring data and alarms using versatile DOM (VDM).

Due to more power consumption by the 400G Digital Coherent QSFP-DD optical modules, the Cisco IOS XR software operates the fans at an higher speed to cool these optical modules.

The 400G Digital Coherent QSFP-DD optical module configuration is divided into the following categories:

- Traffic configuration – Comprises configuring DAC rate, muxponder mode, modulation, and FEC parameters. Applicable for optics controllers:
  - [Configuring DAC Rate, on page 288](#)
  - [Configuring Muxponder Mode, on page 283](#)

- [Configuring Modulation, on page 286](#)
- [Configuring FEC, on page 289](#)
- Optical configuration – Comprises configuring frequency, chromatic dispersion, and optical transmit power. Applicable for optics controllers:
  - [Configuring Frequency, on page 276](#)
  - [Configuring Chromatic Dispersion, on page 278](#)
  - [Configuring Optical Transmit Power, on page 280](#)
- Performance monitoring (PM) – Enables or disables performance monitoring in optical modules. You can also configure PM parameters that comprise signal power, chromatic dispersion, optical signal-to-noise ratio (OSNR), and differential group delay (DGD). Applicable for optics controllers and coherent DSP controllers:
  - [Configuring Performance Monitoring, on page 292](#)
  - [Configuring PM Parameters, on page 292](#)
- Loopback configuration – Configures loopback. Applicable for coherent DSP controller:
  - [Configuring Loopback, on page 291](#)
- Alarms threshold configuration – Configures thresholds for monitoring alarms that include optical signal-to-noise ratio (OSNR), differential group delay (DGD), chromatic dispersion (cd high and low), and so on. Applicable for optics controllers:
  - [Configuring Alarms Threshold, on page 295](#)

The following table contains the possible traffic configuration values for the 400G Digital Coherent QSFP-DD optical modules, in the transponder and muxponder mode:

### Restrictions and Limitations

- 400G Digital Coherent QSFP-DD optical modules are supported on all 400G ports of NC57-MOD-S line cards.
- 400G Digital Coherent QSFP-DD optical modules are supported on all 400G ports of fixed-port routers.
- 400G Digital Coherent QSFP-DD optical modules are supported only on 400G even-numbered ports (at the top row) of the line cards. In addition, the following points describe the limitations of specific line cards:
  - NC57-24DD: All twelve 400G even-numbered ports support 400G Digital Coherent QSFP-DD optical modules.
  - NC57-18DD-SE: Up to a maximum of six 400G Digital Coherent QSFP-DD optical modules are supported in the 400G even-numbered ports.
  - NC57-36H6D-S: Up to a maximum of six 400G Digital Coherent QSFP-DD optical modules are supported in the 400G even-numbered ports.
- The following platform combination doesn't support native 400G speed but can operate in 4x100G mode:

- NCS-57C3-MOD-S/-SE-S with NC57-MPA-2D4H-S in MPA slot1
- NC55-MOD-A-SE-S with NC57-MPA-2D4H-S
- NCS-55A2-MOD-S/-HD-S/-HX-S with NC57-MPA-2D4H-S

**FPD Upgrades Enabled for QDD-400G-ZR-S and QDD-400G-ZRP-S Optical Modules**

*Table 27: Feature History Table*

| Feature Name                                                                                              | Release Information | Feature Description                                                                                                                                                                                                                                                                                                                           |
|-----------------------------------------------------------------------------------------------------------|---------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| FPD Upgrades Enabled for <a href="#">QDD-400G-ZR-S</a> and <a href="#">QDD-400G-ZRP-S</a> Optical Modules | Release 7.3.2       | This feature allows you to perform Field Programmable Device (FPD) upgrades on the QDD-400G-ZR-S and QDD-400G-ZRP-S optical modules to ensure they have the latest fixes and features. For more information about the optic module portfolio, see the <a href="#">Cisco 400G Digital Coherent Optics QSFP-DD Optical Modules Data Sheet</a> . |

Although an FPD upgrade is not mandatory in this release, we recommend upgrading the FPD to the latest version in the subsequent releases to ensure that all the latest fixes and features are enabled on the optical modules.

The QDD-400G-ZR-S and QDD-400G-ZRP-S optical modules have two internal FPD image banks: image banks A and B. These image banks contain running and programmed FPD versions, which are fetched during boot-up. The active image is fetched from bank A, while the standby image is fetched from bank B. To upgrade the optical modules, you must perform the FPD upgrade twice, once for the active image bank and once for the standby image bank. After each upgrade, you must disable and re-enable the QDD-400G-ZR-S and QDD-400G-ZRP-S optical modules using the **controller optics** command to activate the latest firmware.

See the *Upgrading Field-Programmable Device* chapter in the *System Management Configuration Guide for Cisco NCS 5500 Series Routers* for details on the procedure to upgrade the FPD.

**QDD-400G-ZR-S Transponder and Muxponder Configuration Values**

The following table contains the possible Transponder and Muxponder configuration values for the QDD-400G-ZR-S optical module:

*Table 28: QDD-400G-ZR-S Transponder and Muxponder Configuration Values*

| TXP/MXP     | Client                | Trunk         | Modulation | FEC  | DAC Rate |
|-------------|-----------------------|---------------|------------|------|----------|
| 400G-TXP    | 1 client, 400G speed  | 1 trunk, 400G | 16 QAM     | cFEC | 1x1      |
| 4x100G- MXP | 4 clients, 100G speed | 1 trunk, 400G | 16 QAM     | cFEC | 1x1      |

### QDD-400G-ZRP-S Transponder and Muxponder Configuration Values

The following table contains the possible Transponder and Muxponder configuration values for the QDD-400G-ZRP-S optical module:

**Table 29: QDD-400G-ZRP-S Transponder and Muxponder Configuration Values**

| TXP/MXP     | Client                | Trunk               | Modulation | FEC  | DAC Rate |
|-------------|-----------------------|---------------------|------------|------|----------|
| 400G-TXP    | 1 Client, 400G speed  | 1 trunk, 400G speed | 16 QAM     | oFEC | 1x1.25   |
| 400G-TXP    | 1 Client, 400G speed  | 1 trunk, 400G speed | 16 QAM     | cFEC | 1x1      |
| 4x100G- MXP | 4 clients, 100G speed | 1 trunk, 400G speed | 16 QAM     | oFEC | 1x1.25   |
| 4x100G- MXP | 4 clients, 100G speed | 1 trunk, 400G speed | 16 QAM     | cFEC | 1x1      |
| 3x100G-MXP  | 3 clients, 100G speed | 1 trunk, 300G speed | 8 QAM      | oFEC | 1x1.25   |
| 2x100G-MXP  | 2 clients, 100G speed | 1 trunk, 200G speed | QPSK       | oFEC | 1x1.5    |
| 1x100G-MXP  | 1 client, 100G speed  | 1 trunk, 100G speed | QPSK       | oFEC | 1x1.5    |

- [Configuring Frequency, on page 276](#)
- [Configuring Chromatic Dispersion, on page 278](#)
- [Configuring Optical Transmit Power, on page 280](#)
- [Configuring Muxponder Mode, on page 283](#)
- [Configuring Modulation, on page 286](#)
- [Configuring DAC Rate, on page 288](#)
- [Configuring FEC, on page 289](#)
- [Configuring Loopback, on page 291](#)
- [Configuring Performance Monitoring, on page 292](#)
- [Configuring PM Parameters, on page 292](#)
- [Configuring Alarms Threshold, on page 295](#)
- [Alarms Troubleshooting, on page 297](#)

## Configuring Frequency

You can configure frequency on optics controllers. You can select any C band frequency between the range 196.1 to 191.3 THz, in both ITU and NON-ITU channels.



**Note** The 100MHz-grid keyword accepts only frequency values as user input. The 50GHz-grid keyword accepts frequency, ITU-channel, or wavelength values as user input. The Cisco IOS XR software then calculates the frequency for a given wavelength or ITU-channel.

### Frequency Configuration Example

The following example shows how to configure frequency on the optics controller:

```
Router#config
Router(config)#controller optics 0/2/0/16
Router(config-Optics)#dwdm-carrier 100MHz-grid frequency 1921500
Router(config-Optics)#commit
Router(config-Optics)#exit
Router(config)#exit
```

### Running Configuration

This example shows the running configuration:

```
Router#show run controller optics 0/2/0/16
Fri May 28 01:42:32.488 UTC
controller Optics0/2/0/16
 dwdm-carrier 100MHz-grid frequency 1921500
 cd-low-threshold -5000
 cd-high-threshold -5000
!
```

### Verification

This example shows how to verify the frequency configuration:

```
Router#show controller optics 0/2/0/16
Fri May 28 01:47:23.953 UTC
Controller State: Up
Transport Admin State: In Service
Laser State: Off
LED State: Off
FEC State: FEC ENABLED
Optics Status
 Optics Type: QSPDD 400G ZRP
 DWDM carrier Info: C BAND, MSA ITU Channel=80, Frequency=192.15THz,
 Wavelength=1560.200nm
 Alarm Status:

 Detected Alarms: None
 LOS/LOL/Fault Status:
 Alarm Statistics:

 HIGH-RX-PWR = 0 LOW-RX-PWR = 0
 HIGH-TX-PWR = 0 LOW-TX-PWR = 0
 HIGH-LBC = 0 HIGH-DGD = 0
 OOR-CD = 0 OSNR = 0
 WVl-OOL = 0 MEA = 0
 IMPROPER-REM = 0
 TX-POWER-PROV-MISMATCH = 0
 Laser Bias Current = 0.0 mA
 Actual TX Power = -40.00 dBm
 RX Power = -40.00 dBm
 RX Signal Power = -40.00 dBm
 Frequency Offset = 0 MHz
 Laser Temperature = 0.00 Celsius
 Laser Age = 0 %
```

```

DAC Rate = 1x1.25
Performance Monitoring: Enable
THRESHOLD VALUES

Parameter High Alarm Low Alarm High Warning Low Warning

Rx Power Threshold(dBm) 13.0 -24.0 10.0 -22.0
Tx Power Threshold(dBm) 0.0 -16.0 -2.0 -14.0
LBC Threshold(mA) 0.00 0.00 0.00 0.00
Temp. Threshold(celsius) 80.00 -5.00 75.00 0.00
Voltage Threshold(volt) 3.46 3.13 3.43 3.16
LBC High Threshold = 98 %
Configured Tx Power = -10.00 dBm
Configured CD High Threshold = -5000 ps/nm
Configured CD lower Threshold = -5000 ps/nm
Configured OSNR lower Threshold = 9.00 dB
Configured DGD Higher Threshold = 80.00 ps
Baud Rate = 60.1385459900 GBd
Modulation Type: 16QAM
Chromatic Dispersion 0 ps/nm
Configured CD-MIN -26000 ps/nm CD-MAX 26000 ps/nm
Second Order Polarization Mode Dispersion = 0.00 ps^2
Optical Signal to Noise Ratio = 0.00 dB
Polarization Dependent Loss = 0.00 dB
Polarization Change Rate = 0.00 rad/s
Differential Group Delay = 0.00 ps
Temperature = 21.00 Celsius
Voltage = 3.42 V
Transceiver Vendor Details
Form Factor : QSFP-DD
Optics type : QSFPDD 400G ZRP
Name : CISCO-ACACIA
OUI Number : 7c.b2.5c
Part Number : DP04QSDD-E30-19E
Rev Number : 10
Serial Number : ACA244900GN
PID : QDD-400G-ZRP-S
VID : ES03
Firmware Version : 161.06
Date Code(yy/mm/dd) : 20/12/08
!

```

## Configuring Chromatic Dispersion

You can configure chromatic dispersion on optics controllers. When you configure the maximum and minimum values for chromatic dispersion for any data rate, ensure that the minimum difference between the configured values is equal to or greater than 1000 ps/nm.

The following table lists the default CD search range:

**Table 30: Default CD Search Range**

| Muxponder Rate | FEC Value | Default CD Search Range (Min-Max) |
|----------------|-----------|-----------------------------------|
| 400            | OFEC      | -26000 to +26000                  |
| 400            | CFEC      | -2400 to +2400                    |

| Muxponder Rate | FEC Value | Default CD Search Range (Min-Max) |
|----------------|-----------|-----------------------------------|
| 300            | OFEC      | -50000 to +50000                  |
| 200            | OFEC      | -50000 to +50000                  |
| 100            | OFEC      | -80000 to +80000                  |



**Note** For **cd-max** and **cd-min** range details, see the controller optics command.

### Chromatic Dispersion Configuration Example

This example shows how to configure chromatic dispersion on the optics controller:

```
Router#configure
Router(config)#controller optics 0/0/0/13
Router(config-Optics)#cd-max 4000
Router(config-Optics)#cd-min -4000
Router(config-Optics)#commit
Router(config-Optics)#exit
Router(config)#exit
```

### Running Configuration

This example shows the running configuration for the optics controller:

```
Router#show run controller optics 0/0/0/13
Thu May 13 12:24:42.353 UTC
controller Optics0/0/0/13
 cd-min -4000
 cd-max 4000
 !
```

### Verification

This example shows how to verify the configured chromatic dispersion values for the optics controller:

```
Router#show controller optics 0/0/0/13
Controller State: Up
Transport Admin State: In Service
Laser State: On
LED State: Green
FEC State: FEC ENABLED
Optics Status
 Optics Type: QSFPDD 400G ZR
 DWDM carrier Info: C BAND, MSA ITU Channel=61, Frequency=193.10THz,
 Wavelength=1552.524nm
 Alarm Status:

 Detected Alarms: None
 LOS/LOL/Fault Status:
 Alarm Statistics:

 HIGH-RX-PWR = 0 LOW-RX-PWR = 0
 HIGH-TX-PWR = 0 LOW-TX-PWR = 0
 HIGH-LBC = 0 HIGH-DGD = 0
 OOR-CD = 0 OSNR = 35
 WV-LOOL = 0 MEA = 0
 IMPROPER-REM = 0
```

```

TX-POWER-PROV-MISMATCH = 0
Laser Bias Current = 0.0 %
Actual TX Power = -7.87 dBm
RX Power = -8.27 dBm
RX Signal Power = -8.43 dBm
Frequency Offset = 130 MHz
Performance Monitoring: Enable
THRESHOLD VALUES

Parameter High Alarm Low Alarm High Warning Low Warning

Rx Power Threshold(dBm) 1.9 -28.2 0.0 -25.0
Tx Power Threshold(dBm) 0.0 -15.0 -2.0 -16.0
LBC Threshold(mA) 0.00 0.00 0.00 0.00
Temp. Threshold(celsius) 80.00 -5.00 75.00 15.00
Voltage Threshold(volt) 3.46 3.13 3.43 3.16
LBC High Threshold = 98 %
Configured Tx Power = -6.00 dBm
Configured CD High Threshold = 80000 ps/nm
Configured CD lower Threshold = -80000 ps/nm
Configured OSNR lower Threshold = 9.00 dB
Configured DGD Higher Threshold = 80.00 ps
Baud Rate = 59.8437500000 GBd
Modulation Type: 16QAM
Chromatic Dispersion 0 ps/nm
Configured CD-MIN -4000 ps/nm CD-MAX 4000 ps/nm
Second Order Polarization Mode Dispersion = 5.00 ps^2
Optical Signal to Noise Ratio = 36.30 dB
Polarization Dependent Loss = 0.40 dB
Polarization Change Rate = 0.00 rad/s
Differential Group Delay = 4.00 ps
Temperature = 54.00 Celsius
Voltage = 3.37 V
Transceiver Vendor Details
Form Factor : QSFP-DD
Optics type : QSFPDD 400G ZR
Name : CISCO-ACACIA
OUI Number : 7c.b2.5c
Part Number : DP04QSDD-E20-19E
Rev Number : 10
Serial Number : ACA2447003L
PID : QDD-400G-ZR-S
VID : ES03
Firmware Version : 61.12
Date Code(yy/mm/dd) : 20/12/02

```

## Configuring Optical Transmit Power

You can set the transmit power of the optical signal.

Each 400G Digital Coherent QSFP-DD optical module has its own optical transmit (TX) power range. User can change the optical transmit (TX) power value based on the module capability. For "Transmitter specifications", see the following data sheets:

- [Cisco 400G Digital Coherent Optics QSFP-DD Optical Modules Data Sheet](#)



Table 31: Optical Transmit Power Values

| Optical Module | Trunk Speed | Optical Transmit Power (Tx) Shaping | Interval | Supported Range of Optical Transmit Power (Tx) Values (in units of 0.1dBm) <sup>1</sup> |                         |                            |
|----------------|-------------|-------------------------------------|----------|-----------------------------------------------------------------------------------------|-------------------------|----------------------------|
|                |             |                                     |          | Minimum Value                                                                           | Maximum Value - Typical | Maximum Value - Worst Case |
| QDD-400G-ZR-S  | 400G        | No                                  | 1        | -150                                                                                    | -100                    | -100                       |
| QDD-400G-ZRPS  | 400G        | Yes                                 | 1        | -150                                                                                    | -110                    | -130                       |
|                | 300G        |                                     |          | -150                                                                                    | -104                    | -119                       |
|                | 200G        |                                     |          | -150                                                                                    | -90                     | -105                       |
|                | 100G        |                                     |          | -150                                                                                    | -59                     | -75                        |

1. The default optical transmit power (Tx) value is -10 dBm, however with TX shaping enabled the maximum power in 1x400G, 4x100G, 3x100G, 2x100G, and 1x100G modes may be less than -10 dBm.

### Transmitting Power Configuration Example

The following example shows how to configure the optical transmit (TX) power on the optics controller:

```
Router#config
Router(config)#controller optics 0/2/0/16
Router(config-Optics)#transmit-power -125
Router(config-Optics)#commit
Router(config-Optics)#exit
Router(config)#exit
```

### Running Configuration

This example shows the running configuration for the optics controller:

```
Router#show run controller optics 0/2/0/16
Thu May 13 12:52:35.020 UTC
controller Optics0/0/0/1
 cd-min -4000
 cd-max 4000
 transmit-power -125
!
```

### Verification

This example shows how to verify the configured optical transmit power for the optics controller:

```
Router#show controller optics 0/2/0/16
Fri May 28 02:52:06.182 UTC
Controller State: Up
Transport Admin State: In Service
Laser State: Off
LED State: Off
FEC State: FEC ENABLED
Optics Status
 Optics Type: QSFPDD 400G ZRP
 DWDM carrier Info: C BAND, MSA ITU Channel=80, Frequency=192.15THz,
 Wavelength=1560.200nm
 Alarm Status:

```

Detected Alarms: None  
 LOS/LOL/Fault Status:  
 Alarm Statistics:

```

HIGH-RX-PWR = 0 LOW-RX-PWR = 0
HIGH-TX-PWR = 0 LOW-TX-PWR = 0
HIGH-LBC = 0 HIGH-DGD = 0
OOR-CD = 0 OSNR = 0
WVL-OOL = 0 MEA = 0
IMPROPER-REM = 0
TX-POWER-PROV-MISMATCH = 0
Laser Bias Current = 0.0 mA
Actual TX Power = -40.00 dBm
RX Power = -40.00 dBm
RX Signal Power = -40.00 dBm
Frequency Offset = 0 MHz
Laser Temperature = 0.00 Celsius
Laser Age = 0 %
DAC Rate = 1x1.25
Performance Monitoring: Enable

```

THRESHOLD VALUES

```

Parameter High Alarm Low Alarm High Warning Low Warning

Rx Power Threshold(dBm) 13.0 -24.0 10.0 -22.0
Tx Power Threshold(dBm) 0.0 -16.0 -2.0 -14.0
LBC Threshold(mA) 0.00 0.00 0.00 0.00
Temp. Threshold(celsius) 80.00 -5.00 75.00 0.00
Voltage Threshold(volt) 3.46 3.13 3.43 3.16
LBC High Threshold = 98 %

```

**Configured Tx Power = -12.50 dBm**

```

Configured CD High Threshold = -5000 ps/nm
Configured CD lower Threshold = -5000 ps/nm
Configured OSNR lower Threshold = 9.00 dB
Configured DGD Higher Threshold = 80.00 ps
Baud Rate = 60.1385459900 GBd
Modulation Type: 16QAM
Chromatic Dispersion 0 ps/nm
Configured CD-MIN -4000 ps/nm CD-MAX 4000 ps/nm
Second Order Polarization Mode Dispersion = 0.00 ps^2
Optical Signal to Noise Ratio = 0.00 dB
Polarization Dependent Loss = 0.00 dB
Polarization Change Rate = 0.00 rad/s
Differential Group Delay = 0.00 ps
Temperature = 20.00 Celsius
Voltage = 3.41 V

```

Transceiver Vendor Details

```

Form Factor : QSFP-DD
Optics type : QSFPDD 400G ZRP
Name : CISCO-ACACIA
OUI Number : 7c.b2.5c
Part Number : DP04QSDD-E30-19E
Rev Number : 10
Serial Number : ACA244900GN
PID : QDD-400G-ZRP-S
VID : ES03
Firmware Version : 161.06
Date Code(yy/mm/dd) : 20/12/08

```

# Configuring Muxponder Mode

By default, the Cisco IOS XR software configures the 400G Digital Coherent QSFP-DD optical modules in the 400G transponder mode.

However, you can configure muxponder mode on optics controllers. Based on the muxponder mode, you can choose the modulation.

**Table 32: Supported Ports and Command for Configuring Muxponder Mode**

| Fixed-Port Routers | Ports                                                          | Command                                                                                                                                            |
|--------------------|----------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| NCS-57B1-6D24H-S   | 400G ports (ports 24-29)                                       | <b>controller optics</b> <i>0/x/{mpa}/{port}</i> <b>breakout muxponder mode</b>                                                                    |
| NCS-57B1-5D24H-SE  | 400G ports (ports 24-28)                                       | <b>controller optics</b> <i>0/x/{mpa}/{port}</i> <b>breakout muxponder mode</b>                                                                    |
| Line Cards         | Ports                                                          | Command                                                                                                                                            |
| NC57-24DD          | 400G even-numbered ports (port 0, port 2, port 4, and so on)   | <b>controller optics</b> <i>0/x/{mpa}/{port}</i> <b>breakout muxponder mode</b>                                                                    |
| NC57-18DD-SE       | 400G ports (port 18 through port 23)                           | <b>controller optics</b> <i>0/x/{mpa}/{port}</i> <b>breakout muxponder mode</b>                                                                    |
|                    | remaining 400G even-numbered ports (port 0, port 2, and so on) | <b>hw-module port-range</b> <i>{start port   end port}</i> <b>instance</b> <i>{mpa}</i> <b>location</b> <i>0/x/CPU0</i> <b>mode muxponder mode</b> |
| NC57-36H6D-S       | 400G even-numbered (ports 24-35) ports                         | <b>hw-module port-range</b> <i>{start port   end port}</i> <b>instance</b> <i>{mpa}</i> <b>location</b> <i>0/x/CPU0</i> <b>mode muxponder mode</b> |

**Table 33: Other Platform Combinations: Supported Ports and Commands for Configuring Muxponder Mode**

| Platform Combination                                    | MPA Slots /Ports       | Mode                 | Command                                                                                                                                          |
|---------------------------------------------------------|------------------------|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------|
| NC57-MPA-2D4H-S (800G mode MPA) in NCS-57C3-MOD-S/-SE-S | Slots 2, 3 / Ports 0-3 | 400G, 4x100G, 3x100G | <b>hw-module port-range</b> <i>{0 1   2 3}</i> <b>instance</b> <i>{mpa}</i> <b>location</b> <i>0/x/CPU0</i> <b>mode</b> <i>{400 4x100 3x100}</i> |
| NC57-MPA-2D4H-S (800G mode MPA) in NC57-MOD-S           | Slots 1, 2 / Ports 0-3 | 2x100G, 1x100G       | <b>controller optics</b> <i>0/x/{mpa}/{0/1/2/3}</i> <b>breakout</b> <i>{2x100 1x100}</i>                                                         |

| Platform Combination                                                               | MPA Slots /Ports      | Mode                                                                                                                | Command                                                                                                |
|------------------------------------------------------------------------------------|-----------------------|---------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------|
| NC57-MPA-2D4H-S (400G mode MPA) in<br>NCS-57C3-MOD-S/-SE-S                         | Slot 1 /<br>Ports 0-3 | 4x100G                                                                                                              | <b>hw-module port-range</b> {0   3} <b>instance</b> {mpa} <b>location</b> 0/0/CPU0 <b>mode</b> {4x100} |
|                                                                                    |                       | 3x100G                                                                                                              | <b>hw-module port-range</b> {0   2} <b>instance</b> {mpa} <b>location</b> 0/0/CPU0 <b>mode</b> {3x100} |
| 2x100G                                                                             |                       | <b>hw-module port-range</b> {0   1   2   3} <b>instance</b> {mpa} <b>location</b> 0/0/CPU0 <b>mode</b> {2x100-PAM4} |                                                                                                        |
| 1x100G                                                                             |                       | <b>controller optics</b> 0/0/{mpa}/{0 1 2 3} <b>breakout</b> {1x100}                                                |                                                                                                        |
| NC57-MPA-2D4H-S (400G mode MPA) in<br>NCS-55A2-MOD-S/<br>-HD-S/-SE-S/-SE-H-S/-HX-S |                       |                                                                                                                     |                                                                                                        |
| NC57-MPA-2D4H-S (400G mode MPA) in<br>NC55-MOD-A-S/-SE-S                           |                       |                                                                                                                     |                                                                                                        |



**Note** The following line cards do not support CVR-QSFP-SFP10G and any 1Gbps optics:

- NCS-57B1-6D24-SYS
- NCS-57B1-5DSE-SYS
- NC57-24DD
- NC57-18DD-SE
- NC57-36H-SE
- NC57-36H6D
- NC57-MOD-S

Muxponder mode options available for QDD-400G-ZR-S are:

- 4x100

Muxponder mode options available for QDD-400G-ZRP-S are:

- 4x100
- 3x100
- 2x100 (or 2x100-PAM4)
- 1x100

See the following tables for the modulation values, based on the muxponder mode:

- [Table 28: QDD-400G-ZR-S Transponder and Muxponder Configuration Values, on page 275](#)
- [Table 29: QDD-400G-ZRP-S Transponder and Muxponder Configuration Values, on page 276](#)

Using the **no breakout muxponder mode** command, you can switch from the muxponder mode to the transponder mode, on optics controllers.

### Muxponder Mode Configuration Example

The following example shows how to configure muxponder mode on the optics controller:

```
Router#config
Router(config)#controller optics 0/0/0/13
Router(config-Optics)#breakout 4x100
Router(config-Optics)#commit
Router(config-Optics)#exit
Router(config)#exit
```



**Note** In the above example, the Cisco IOS XR software creates four Ethernet clients with 100GE speed, which can be verified using the **show interfaces brief | include R/S/I/P** command.

### Running Configuration

This example shows the running configuration for the optics controller:

```
Router#show run controller optics 0/0/0/13
Thu May 13 12:24:42.353 UTC
controller Optics0/0/0/13
 cd-min -4000
 cd-max 4000
 breakout 4x100
!
```

### Verification

This example shows how to verify the muxponder mode configuration:

```
Router#show interfaces brief | include 0/0/0/13
Hu0/0/0/13/0 up up ARPA 1514 100000000
Hu0/0/0/13/1 up up ARPA 1514 100000000
Hu0/0/0/13/2 up up ARPA 1514 100000000
Hu0/0/0/13/3 up up ARPA 1514 100000000
```

### Transponder Mode Configuration Example

The following example shows how to switch to the transponder mode, on the optics controller:

```
Router#config
Router(config)#controller optics 0/0/0/13
Router(config-Optics)#no breakout 4x100
Router(config-Optics)#commit
Router(config-Optics)#exit
Router(config)#exit
```



**Note** The Cisco IOS XR software creates a single 400GE interface, which can be verified using the **show interfaces brief | include R/S/I/P** command.

### Running Configuration

This example shows the running configuration for the optics controller. The breakout configuration is absent in the running configuration.

```
Router#show run controller optics 0/0/0/13
Thu May 13 13:51:20.330 UTC
controller Optics0/0/0/13
 cd-min -4000
 cd-max 4000
 transmit-power -100
!
```

### Verification

This example shows how to verify the transponder mode configuration:

```
Router#show interfaces brief | include 0/0/0/13
FH0/0/0/13 up up ARPA 1514 400000000
```

## Configuring Modulation

You can configure modulation on optics controllers. Based on the muxponder mode, you can choose the modulation.




---

**Note** The system accepts any modulation value that is entered. However, if the modulation value is outside the supported range, it is not configured on the optical module. Instead, the optical module is auto-configured with a valid modulation value. To view this value, use the **show controller optics R/S/I/P** command.

---

See the following tables for the supported modulation values:

- [Table 28: QDD-400G-ZR-S Transponder and Muxponder Configuration Values, on page 275](#)
- [Table 29: QDD-400G-ZRP-S Transponder and Muxponder Configuration Values, on page 276](#)

### Modulation Configuration Example

The following example shows how to configure modulation on the optics controller:

```
Router#config
Router(config)#controller optics 0/0/0/1
Router(config-Optics)#modulation 16Qam
Router(config-Optics)#commit
Router(config-Optics)#exit
Router(config)#exit
```

### Running Configuration

This example shows the running configuration:

```
Router#show run controller optics 0/0/0/1
controller Optics0/0/0/1
 cd-min -4000
 cd-max 4000
 transmit-power -100
 modulation 16Qam
!
```




---

**Note** Use the **show controller optics R/S/I/P** command to verify the modulation value of the optical module.

---

### Verification

This example shows how to verify the configured modulation value for the optics controller:

```
Router#show controller optics 0/0/0/1
Controller State: Up
Transport Admin State: In Service
Laser State: On
LED State: Green
FEC State: FEC ENABLED
Optics Status
 Optics Type: QSPFDD 400G ZR
 DWDM carrier Info: C BAND, MSA ITU Channel=61, Frequency=193.10THz,
 Wavelength=1552.524nm
 Alarm Status:

 Detected Alarms: None
 LOS/LOL/Fault Status:
 Alarm Statistics:

 HIGH-RX-PWR = 0 LOW-RX-PWR = 0
 HIGH-TX-PWR = 0 LOW-TX-PWR = 0
 HIGH-LBC = 0 HIGH-DGD = 0
 OOR-CD = 0 OSNR = 35
 WVL-OOL = 0 MEA = 0
 IMPROPER-REM = 0
 TX-POWER-PROV-MISMATCH = 0
 Laser Bias Current = 0.0 %
 Actual TX Power = -7.87 dBm
 RX Power = -8.27 dBm
 RX Signal Power = -8.43 dBm
 Frequency Offset = 130 MHz
 Performance Monitoring: Enable
 THRESHOLD VALUES

 Parameter High Alarm Low Alarm High Warning Low Warning

 Rx Power Threshold(dBm) 1.9 -28.2 0.0 -25.0
 Tx Power Threshold(dBm) 0.0 -15.0 -2.0 -16.0
 LBC Threshold(mA) 0.00 0.00 0.00 0.00
 Temp. Threshold(celsius) 80.00 -5.00 75.00 15.00
 Voltage Threshold(volt) 3.46 3.13 3.43 3.16
 LBC High Threshold = 98 %
 Configured Tx Power = -6.00 dBm
 Configured CD High Threshold = 80000 ps/nm
 Configured CD lower Threshold = -80000 ps/nm
 Configured OSNR lower Threshold = 9.00 dB
 Configured DGD Higher Threshold = 80.00 ps
 Baud Rate = 59.8437500000 GBd
Modulation Type: 16QAM
 Chromatic Dispersion 0 ps/nm
 Configured CD-MIN -4000 ps/nm CD-MAX 4000 ps/nm
 Second Order Polarization Mode Dispersion = 5.00 ps^2
 Optical Signal to Noise Ratio = 36.30 dB
 Polarization Dependent Loss = 0.40 dB
 Polarization Change Rate = 0.00 rad/s
 Differential Group Delay = 4.00 ps
 Temperature = 54.00 Celsius
 Voltage = 3.37 V
Transceiver Vendor Details
 Form Factor : QSFP-DD
 Optics type : QSPFDD 400G ZR
 Name : CISCO-ACACIA
 OUI Number : 7c.b2.5c
```

```

Part Number : DP04QSDD-E20-19E
Rev Number : 10
Serial Number : ACA2447003L
PID : QDD-400G-ZR-S
VID : ES03
Firmware Version : 61.12
Date Code (yy/mm/dd) : 20/12/02

```

## Configuring DAC Rate

You can set the DAC (digital to analog conversion) sampling rate on optics controllers. You can modify the DAC sampling rate only on the QDD-400G-ZRP-S optical modules.



**Note** QDD-400G-ZR-S supports 1x1 dac-rate in cFEC mode. QDD-400G-ZRP-S optical modules support 1x1 dac-rate in cFEC mode and 1x1.25 dac-rate in oFEC mode.

### DAC Rate Configuration Example

The following example shows how to set the DAC rate on the optics controller:

```

Router#config
Router(config)#controller optics 0/0/0/1
Router(config-Optics)#dac-rate 1x1

```

### Verification

This example shows the running configuration:

```

Router#show run controller optics 0/0/0/1
Thu May 13 12:52:35.020 UTC
controller Optics0/0/0/1
 cd-min -4000
 cd-max 4000
 transmit-power -100
 modulation 16Qam
 DAC-Rate 1x1
!
!

```

### Verification

This example shows how to verify the configured DAC rate for the optics controller:

```

Router#show controller optics 0/0/0/1
Controller State: Up
Transport Admin State: In Service
Laser State: On
LED State: Green
FEC State: FEC ENABLED
Optics Status
 Optics Type: QSFPDD 400G ZR
 DWDM carrier Info: C BAND, MSA ITU Channel=61, Frequency=193.10THz,
 Wavelength=1552.524nm
 Alarm Status:

 Detected Alarms: None
 LOS/LOL/Fault Status:
 Alarm Statistics:

```



```

HIGH-RX-PWR = 0 LOW-RX-PWR = 0
HIGH-TX-PWR = 0 LOW-TX-PWR = 0
HIGH-LBC = 0 HIGH-DGD = 0
OOR-CD = 0 OSNR = 35
WVL-OOL = 0 MEA = 0
IMPROPER-REM = 0
TX-POWER-PROV-MISMATCH = 0
Laser Bias Current = 0.0 %
Actual TX Power = -7.87 dBm
RX Power = -8.27 dBm
RX Signal Power = -8.43 dBm
Frequency Offset = 130 MHz
DAC Rate = 1x1
Performance Monitoring: Enable
THRESHOLD VALUES

Parameter High Alarm Low Alarm High Warning Low Warning

Rx Power Threshold(dBm) 1.9 -28.2 0.0 -25.0
Tx Power Threshold(dBm) 0.0 -15.0 -2.0 -16.0
LBC Threshold(mA) 0.00 0.00 0.00 0.00
Temp. Threshold(celsius) 80.00 -5.00 75.00 15.00
Voltage Threshold(volt) 3.46 3.13 3.43 3.16
LBC High Threshold = 98 %
Configured Tx Power = -6.00 dBm
Configured CD High Threshold = 80000 ps/nm
Configured CD lower Threshold = -80000 ps/nm
Configured OSNR lower Threshold = 9.00 dB
Configured DGD Higher Threshold = 80.00 ps
Baud Rate = 59.8437500000 GBd
Modulation Type: 16QAM
Chromatic Dispersion 0 ps/nm
Configured CD-MIN -4000 ps/nm CD-MAX 4000 ps/nm
Second Order Polarization Mode Dispersion = 5.00 ps^2
Optical Signal to Noise Ratio = 36.30 dB
Polarization Dependent Loss = 0.40 dB
Polarization Change Rate = 0.00 rad/s
Differential Group Delay = 4.00 ps
Temperature = 54.00 Celsius
Voltage = 3.37 V
Transceiver Vendor Details
Form Factor : QSFP-DD
Optics type : QSFPDD 400G ZR
Name : CISCO-ACACIA
OUI Number : 7c.b2.5c
Part Number : DP04QSDD-E20-19E
Rev Number : 10
Serial Number : ACA2447003L
PID : QDD-400G-ZR-S
VID : ES03
Firmware Version : 61.12
Date Code(yy/mm/dd) : 20/12/02

```

## Configuring FEC

You can configure forward error correction (FEC) only on optics controllers. You can modify FEC only on the QDD-400G-ZRP-S optical modules. FEC is a feature that is used for controlling errors during data transmission. This feature works by adding data redundancy to the transmitted message using an algorithm. This redundancy allows the receiver to detect and correct a limited number of errors occurring anywhere in the message, instead of having to ask the transmitter to resend the message.



**Note** QDD-400G-ZR-S supports cFEC (concatenated forward error correction). QDD-400G-ZRP-S support cFEC and oFEC (open forward error correction).

### FEC Configuration Example

The following sample shows how to configure FEC on the optics controller:

```
Router#configure
Router(config)#controller optics 0/0/0/13
Router(config-Optics)#fec CFEC
Router(config-Optics)#commit
Router(config-Optics)#exit
Router(config)#exit
```

### Running Configuration

This example shows the running configuration:

```
Router#show controllers optics 0/0/0/13
controller Optics0/0/0/1
 cd-min -4000
 cd-max 4000
 transmit-power -100
 fec CFEC
 modulation 16Qam
 DAC-Rate 1x1.25
!
```

### Verification

This example shows how to verify the FEC configuration for the optics controller:

```
Router#show controller coherentdsp 0/0/0/13
Thu May 27 17:28:51.960 UTC
Port : CoherentDSP 0/0/0/13
Controller State : Down
Inherited Secondary State : Normal
Configured Secondary State : Maintenance
Derived State : Maintenance
Loopback mode : Internal
BER Thresholds : SF = 1.0E-5 SD = 1.0E-7
Performance Monitoring : Enable
Bandwidth : 400.0Gb/s

Alarm Information:
LOS = 6 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0 SF_BER = 0
SD_BER = 0 BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0 FLEXO_GIDM = 0
FLEXO-MM = 0 FLEXO-LOM = 0 FLEXO-RDI = 0
FLEXO-LOF = 5
Detected Alarms : LOS
Bit Error Rate Information
 PREFEC BER : 5.0E-01
 POSTFEC BER : 0.0E+00
 Q-Factor : 0.00 dB
 Q-Margin : -7.20dB
 OTU TTI Received

FEC mode : C_FEC
```

# Configuring Loopback

You can configure internal or line loopback on coherent DSP controllers. Loopback can be performed only in the maintenance mode.

## Loopback Configuration Example

This example shows how to enable internal loopback configuration on coherent DSP controllers:

```
Router#config
Router(config)#controller coherentDSP 0/0/0/4
Router(config-CoDSP)#secondary-admin-state maintenance
Router(config-CoDSP)#loopback internal
Router(config-CoDSP)#commit
```

## Running Configuration

This example shows the running configuration on coherent DSP controllers:

```
Router#show run controller coherentdsp 0/0/0/4
Thu May 13 19:51:08.175 UTC
controller CoherentDSP0/0/0/4
 secondary-admin-state maintenance
 loopback internal
!
```

## Verification

This example shows how to verify the loopback configuration on coherent DSP controllers:

```
Router#show controller coherentdsp 0/0/0/4
Thu May 27 17:28:51.960 UTC
Port : CoherentDSP 0/0/0/4
Controller State : Down
Inherited Secondary State : Normal
Configured Secondary State : Maintenance
Derived State : Maintenance
Loopback mode : Internal
BER Thresholds : SF = 1.0E-5 SD = 1.0E-7
Performance Monitoring : Enable
Bandwidth : 400.0Gb/s
Alarm Information:
LOS = 6 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0 SF_BER = 0
SD_BER = 0 BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0 FLEXO_GIDM = 0
FLEXO-MM = 0 FLEXO-LOM = 0 FLEXO-RDI = 0
FLEXO-LOF = 5
Detected Alarms : LOS
Bit Error Rate Information
PREFEC BER : 5.0E-01
POSTFEC BER : 0.0E+00
Q-Factor : 0.00 dB
Q-Margin : -7.20dB
OTU TTI Received :
FEC mode : C_FEC
```

## Configuring Performance Monitoring

Performance monitoring (PM) parameters are used by service providers to gather, store, set thresholds for, and report performance data for early detection of problems. The user can retrieve both current and historical PM counters for the various controllers in 30-second, 15-minute, and 24-hour intervals.

Performance monitoring can be configured on optics controllers and coherent DSP controllers.

To stop performance monitoring on optics or coherent DSP controllers, use the **perf-mon disable** keyword.

## Configuring PM Parameters

The performance monitoring (PM) threshold and the threshold crossing alert (TCA) reporting status can be configured for optics controllers and coherent DSP controllers:

**Table 34: PM Thresholds and TCA Report Status for Optics Controllers**

| PM Parameters | Description                                                                               |
|---------------|-------------------------------------------------------------------------------------------|
| CD            | Sets the CD (chromatic dispersion) threshold or TCA reporting status.                     |
| DGD           | Sets the DGD (differential group delay) threshold or TCA reporting status.                |
| LBC           | Sets the LBC (laser bias current) threshold or TCA reporting status in mA.                |
| FREQ-OFF      | Sets the FREQ-OFF (low signal frequency offset) threshold or TCA reporting status in Mhz. |
| OPR           | Sets the OPR (optical power RX) threshold or TCA reporting status in uW or dbm.           |
| OPT           | Sets the OPT (optical power TX) threshold or TCA reporting status in uW or dbm.           |
| OSNR          | Sets the OSNR (optical signal-to-noise ratio) threshold or TCA reporting status.          |
| PCR           | Sets the PCR (polarization change rate) threshold or TCA reporting status.                |
| PDL           | Sets the PDL (polarization dependent loss) threshold or TCA reporting status.             |
| RX-SIG        | Sets the RX-SIG (receiving signal power) threshold or TCA reporting status in uW or dbm.  |
| SNR           | Sets the SNR (signal-to-noise ratio) threshold or TCA reporting status.                   |

| PM Parameters | Description                                                                                   |
|---------------|-----------------------------------------------------------------------------------------------|
| SOPMD         | Sets the SOPMD (second order polarization mode dispersion) threshold or TCA reporting status. |

**Table 35: PM Thresholds TCA Report Status for Coherent DSP Controllers**

| PM Parameters | Description                                                                |
|---------------|----------------------------------------------------------------------------|
| Q             | Sets the Q threshold or TCA reporting status.                              |
| Q-margin      | Sets the Q margin threshold or TCA reporting status.                       |
| EC-BITS       | Sets the EC-BITS (error corrected bits) threshold or TCA reporting status. |
| PostFEC BER   | Sets the post-FEC BER threshold or TCA reporting status.                   |
| PreFEC BER    | Sets the pre-FEC BER threshold or TCA reporting status.                    |
| UC-WORDS      | Sets the UC-WORDS (uncorrected words) threshold or TCA reporting status.   |

### Performance Monitoring Configuration Example

This example shows how to enable performance monitoring and set PM thresholds on the optics controller:

```
Router#config
Router(config)#controller optics 0/2/0/16
Router(config-Optics)#perf-mon enable
Router(config-Optics)#pm 30-sec optics threshold cd max 100
Router(config-Optics)#pm 30-sec optics threshold cd min -100
Router(config-Optics)#commit
```

### Running Configuration

This example shows the running configuration on optics controllers:

```
Router#show run controller optics 0/2/0/16
Thu May 13 20:18:55.957 UTC
controller Optics0/2/0/16
pm 30-sec optics threshold cd max 100
pm 30-sec optics threshold cd min -100
perf-mon enable
!
```

### Verification

This example shows how to verify the PM parameters on optics controllers. Verify the configuration changes in the Configured Threshold fields:

```
Router#show controller optics 0/2/0/16 pm current 30-sec optics 1
Thu May 27 17:58:49.889 UTC
Optics in the current interval [17:58:30 - 17:58:49 Thu May 27 2021]
Optics current bucket type : Valid
 MIN AVG MAX Operational Configured TCA Operational
 Configured TCA
 Threshold(min) Threshold(min) (min) Threshold(max)
```

```

Threshold(max) (max)
LBC[mA] : 0.0 0.0 0.0 0.0 NA NO 100.0
 NA NO
OPT[dBm] : -9.98 -9.98 -9.98 -15.09 NA NO 0.00
 NA NO
OPR[dBm] : -40.00 -40.00 -40.00 -30.00 NA NO 8.00
 NA NO
CD[ps/nm] : 0 0 0 -80000 -100 NO 100
 100 NO
DGD[ps] : 0.00 0.00 0.00 0.00 NA NO 80.00
 NA NO
SOPMD[ps^2] : 0.00 0.00 0.00 0.00 NA NO 2000.00
 NA NO
OSNR[dB] : 0.00 0.00 0.00 0.00 NA NO 40.00
 NA NO
PDL[dB] : 0.00 0.00 0.00 0.00 NA NO 7.00
 NA NO
PCR[rad/s] : 0.00 0.00 0.00 0.00 NA NO 2500000.00
 NA NO
RX_SIG[dBm] : -40.00 -40.00 -40.00 -30.00 NA NO 1.00
 NA NO
FREQ_OFF[Mhz] : 0 0 0 -3600 NA NO 3600
 NA NO
SNR[dB] : 0.00 0.00 0.00 7.00 NA NO 100.00
 NA NO

```

Last clearing of "show controllers OPTICS" counters never  
!

### Performance Monitoring Configuration Example

This example shows how to enable performance monitoring and set PM thresholds and TCA reporting status on the coherent DSP controller:

```

Router#config
Router(config)#controller CoherentDSP0/2/0/16
Router(config-CoDSP)#perf-mon enable
Router(config-CoDSP)#pm 30-sec fec report Q max-tca enable
Router(config-CoDSP)#pm 30-sec fec report Q-margin max-tca enable
Router(config-CoDSP)#pm 30-sec fec report Q min-tca enable
Router(config-CoDSP)#pm 30-sec fec report Q-margin min-tca enable
Router(config-CoDSP)#pm 30-sec fec threshold Q max 1200
Router(config-CoDSP)#pm 30-sec fec threshold Q-margin max 500
Router(config-CoDSP)#pm 30-sec fec threshold Q min 900
Router(config-CoDSP)#pm 30-sec fec threshold Q-margin min 280
Router(config-CoDSP)#commit

```

### Running Configuration

This example shows the running configuration on coherent DSP controllers:

```

Router#show run controller coherentdsp 0/2/0/16
Thu May 13 19:56:09.136 UTC
controller CoherentDSP0/2/0/16
 pm 30-sec fec report Q max-tca enable
 pm 30-sec fec report Q-margin max-tca enable
 pm 30-sec fec report Q min-tca enable
 pm 30-sec fec report Q-margin min-tca enable
 pm 30-sec fec threshold Q max 1200
 pm 30-sec fec threshold Q-margin max 500
 pm 30-sec fec threshold Q min 900
 pm 30-sec fec threshold Q-margin min 280
 perf-mon enable
!

```

## Verification

This example shows how to verify the PM parameters on coherent DSP controllers. Verify the configuration changes in the highlighted fields:

```
Router#show controllers coherentdsp 0/2/0/16 pm current 30-sec fec
Thu May 27 23:04:54.167 UTC
g709 FEC in the current interval [23:04:30 - 23:04:54 Thu May 27 2021]
FEC current bucket type : Valid
 EC-BITS : 0 Threshold : 111484000000 TCA(enable) :
YES
 UC-WORDS : 0 Threshold : 5 TCA(enable) :
YES
 Threshold TCA MIN AVG MAX Threshold TCA
 (max) (enable)
 PreFEC BER : 0E-15 0E-15 0E-15 0E-15 0E-15 NO
 0E-15 NO
 PostFEC BER : 0E-15 0E-15 0E-15 0E-15 0E-15 NO
 0E-15 NO
 Q[dB] : 0.00 0.00 0.00 0.00 9.00 YES 120.00 YES
 Q_Margin[dB] : 0.00 0.00 0.00 0.00 2.80 YES 5.00 YES
 !
```

# Configuring Alarms Threshold

The alarms threshold can be configured for monitoring alarms on optics controllers:

**Table 36: Alarms Threshold Parameters for Optics Controllers**

| Alarm Threshold Parameters | Description                                                                                  |
|----------------------------|----------------------------------------------------------------------------------------------|
| CD                         | Sets the CD (chromatic dispersion) alarm threshold (cd-low-threshold and cd-high-threshold). |
| DGD                        | Sets the DGD (differential group delay) alarm threshold.                                     |
| LBC                        | Sets the LBC (laser bias current) threshold in mA.                                           |
| OSNR                       | Sets the OSNR (optical signal-to-noise ratio) alarm threshold.                               |

## Alarm Threshold Configuration Example

This example shows how to configure alarm threshold on the optics controller:

```
Router#config
Router(config)#controller optics 0/2/0/16
Router(config-Optics)#cd-low-threshold -2000
Router(config-Optics)#cd-high-threshold 2000
Router(config-Optics)#commit
```

## Running Configuration

This example shows the running configuration on the optics controller:

```

Router#show run controller optics 0/2/0/16
Thu May 13 20:18:55.957 UTC
controller Optics0/2/0/16
 cd-low-threshold 2000
 cd-high-threshold 2000
!

```

### Verification

This example shows how to verify the alarm threshold on optics controllers:

```

Router#show controller optics 0/2/0/16
Fri May 28 01:04:33.604 UTC
Controller State: Up
Transport Admin State: In Service
Laser State: Off
LED State: Off
FEC State: FEC ENABLED
Optics Status
 Optics Type: QSFPDD 400G ZRP
 DWDM carrier Info: C BAND, MSA ITU Channel=61, Frequency=193.10THz,
 Wavelength=1552.524nm
 Alarm Status:

 Detected Alarms: None
 LOS/LOL/Fault Status:
 Alarm Statistics:

 HIGH-RX-PWR = 0 LOW-RX-PWR = 0
 HIGH-TX-PWR = 0 LOW-TX-PWR = 0
 HIGH-LBC = 0 HIGH-DGD = 0
 OOR-CD = 0 OSNR = 0
 WVl-OOL = 0 MEA = 0
 IMPROPER-REM = 0
 TX-POWER-PROV-MISMATCH = 0
 Laser Bias Current = 0.0 mA
 Actual TX Power = -40.00 dBm
 RX Power = -40.00 dBm
 RX Signal Power = -40.00 dBm
 Frequency Offset = 0 MHz
 Laser Temperature = 0.00 Celsius
 Laser Age = 0 %
 DAC Rate = 1x1.25
 Performance Monitoring: Enable
 THRESHOLD VALUES

 Parameter High Alarm Low Alarm High Warning Low Warning

 Rx Power Threshold(dBm) 13.0 -24.0 10.0 -22.0
 Tx Power Threshold(dBm) 0.0 -16.0 -2.0 -14.0
 LBC Threshold(mA) 0.00 0.00 0.00 0.00
 Temp. Threshold(celsius) 80.00 -5.00 75.00 0.00
 Voltage Threshold(volt) 3.46 3.13 3.43 3.16
 LBC High Threshold = 98 %
 Configured Tx Power = -10.00 dBm
Configured CD High Threshold = -5000 ps/nm
Configured CD lower Threshold = -5000 ps/nm
 Configured OSNR lower Threshold = 9.00 dB
 Configured DGD Higher Threshold = 80.00 ps
 Baud Rate = 60.1385459900 GBd
 Modulation Type: 16QAM
 Chromatic Dispersion 0 ps/nm
 Configured CD-MIN -26000 ps/nm CD-MAX 26000 ps/nm
 Second Order Polarization Mode Dispersion = 0.00 ps^2
 Optical Signal to Noise Ratio = 0.00 dB

```



```
Polarization Dependent Loss = 0.00 dB
Polarization Change Rate = 0.00 rad/s
Differential Group Delay = 0.00 ps
Temperature = 21.00 Celsius
Voltage = 3.42 V
Transceiver Vendor Details
Form Factor : QSFP-DD
Optics type : QSFPDD 400G ZRP
Name : CISCO-ACACIA
OUI Number : 7c.b2.5c
Part Number : DP04QSDD-E30-19E
Rev Number : 10
Serial Number : ACA244900GN
PID : QDD-400G-ZRP-S
VID : ES03
Firmware Version : 161.06
Date Code (yy/mm/dd) : 20/12/08
```

!

## Alarms Troubleshooting

This section contains the procedures for troubleshooting alarms.

### CD Alarm

Default Severity: Minor (MN), Non-Service-Affecting (NSA)

Logical Object: Controller

The Chromatic Dispersion (CD) alarm is raised when the detected chromatic dispersion value is above or below the configured threshold values.

### Clear the CD Alarm

---

Configure threshold value within range if CD value is not within the threshold range.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

### DGD Alarm

Default Severity: Minor (MN), Non-Service-Affecting (NSA)

Logical Object: Controller

The Differential Group Delay (DGD) alarm is raised when the value of the differential group delay read by the pluggable port module exceeds the configured threshold value.

## Clear the DGD Alarm

---

Configure the threshold value within range if DGD value is not within the threshold range.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## FLEXO\_LOF

Default Severity: Critical

Logical Object: OTN

Flexo LOF alarm is raised when loss of alignment is detected on the Flexo frame for more than 3ms.

## Clear the FLEXO\_LOF Alarm

---

Identify and correct the underlying cause of mis-alignment. The Flexo LOF (Loss of Frame) alarm is cleared when good alignment is detected on the Flexo frame for more than 3ms.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## FLEXO\_LOM

Default Severity: Critical

Logical Object: OTN

Flexo LOM (Loss of Multi-Frame) is raised when loss of multi-frame alignment is detected on the Flexo multi-frame for more than 10ms

## Clear the FLEXO\_LOM Alarm

---

Identify and correct the underlying cause of mis-alignment. The Flexo LOM alarm is cleared when good multi-frame alignment is detected on the Flexo multi-frame.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## HI-LASERBIAS Alarm

Default Severity: Minor (MN), Non-Service-Affecting (NSA)

Logical Object: Controller

The HI-LASERBIAS alarm is raised when the physical pluggable port laser detects a laser bias value beyond the configured high threshold.

## Clear the HI-LASERBIAS Alarm

---

Configure the threshold value within range if high laser bias threshold value is not within the threshold range.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## HI-RXPOWER Alarm

Default Severity: Minor (MN), Non-Service-Affecting (NSA)

Logical Object: Controller

The HI-RXPOWER alarm occurs on the client optics controller when the measured individual lane optical signal power of the received signal exceeds the default threshold. The HI-RXPOWER alarm occurs on the trunk optics controller when the total optical signal power of the received signal exceeds the default threshold.

## Clear the HI-RXPOWER Alarm

---

Physically verify by using a standard power meter that the optical input power is overcoming the expected power threshold. Connect an attenuator accordingly.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## HI-RXPOWER Warn

Default Severity: Minor (MN), Non-Service-Affecting (NSA)

Logical Object: Software

The HI-RXPOWER warning occurs on the client optics controller when the measured individual lane optical signal power of the received signal exceeds the default threshold. The HI-RXPOWER warning occurs on the trunk optics controller when the total optical signal power of the received signal exceeds the default threshold.

## Clear the HI-RXPOWER Warn Alarm

---

Physically verify by using a standard power meter that the optical input power is overcoming the expected power threshold. Connect an attenuator accordingly.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## HI-TEMP Alarm

Default Severity: Critical

Logical Object: Software

The HI-TEMP alarm occurs when the optical module temperature exceeds the default threshold.

### Clear the HI-TEMP Alarm

---

Verify the fan is intact and empty slots are blocked for cooling.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## HI-TEMP Warn

Default Severity: Minor (MN), Non-Service-Affecting (NSA)

Logical Object: Software

The HI-TEMP warning occurs when the optical module temperature exceeds the default threshold.

### Clear the HI-TEMP Warn Alarm

---

Verify the fan is intact and empty slots are blocked for cooling

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## HI-TXPOWER Alarm

Default Severity: Minor (MN), Non-Service-Affecting (NSA)

Logical Object: Controller

The HI-TXPOWER alarm occurs on the client optics controller when the measured individual lane optical signal power of the transmitted signal exceeds the default threshold. The HI-TXPOWER alarm occurs on the trunk optics controller when the total optical signal power of the transmitted signal exceeds the default threshold.

## Clear the HI-TXPOWER Alarm

---

Physically verify by using a standard power meter that the optical output power is overcoming the expected power threshold.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## HI-TXPOWER Warn

Default Severity: Minor (MN), Non-Service-Affecting (NSA)

Logical Object: Software

The HI-TXPOWER warning occurs on the client optics controller when the measured individual lane optical signal power of the transmitted signal exceeds the default threshold. The HI-TXPOWER warning occurs on the trunk optics controller when the total optical signal power of the transmitted signal exceeds the default threshold.

## Clear the HI-TXPOWER Warn Alarm

---

Physically verify by using a standard power meter that the optical output power is overcoming the expected power threshold.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## IMPROPER-REM

Default Severity: Critical

Logical Object: Software

The Improper Removal alarm is raised when a physical pluggable is not present on a service-provisioned port.

## Clear the IMPROPER-REM Alarm

---

Insert the appropriate QSFP.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## LOF

Default Severity: Critical

Logical Object: OTN

Flexo LOF alarm is raised when loss of alignment is detected on the Flexo frame for more than 3ms.

### Clear the LOF Alarm

---

Identify and correct the underlying cause of mis-alignment. The Flexo LOF (Loss of Frame) alarm is cleared when good alignment is detected on the Flexo frame for more than 3ms.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## LOL

Default Severity: Major

Logical Object: Software

LOL alarm is raised when loss of lock is detected on the receive side of the CDR (Clock and Data Recovery)

### Clear the LOL Alarm

---

Verify the fiber and power levels.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## LOM

Default Severity: Critical

Logical Object: OTN

Flexo LOM (Loss of Multi-Frame) is raised when loss of multi-frame alignment is detected on the Flexo multi-frame for more than 10ms

### Clear the LOM Alarm

---

Identify and correct the underlying cause of mis-alignment. The Flexo LOM alarm is cleared when good multi-frame alignment is detected on the Flexo multi-frame.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## LO-RXPOWER Alarm

Default Severity: Minor (MN), Non-Service-Affecting (NSA)

Logical Object: Controller

The LO-RXPOWER alarm is raised on the client or trunk optics controller when the measured individual lane optical signal power of the received signal falls below the default threshold.

### Clear the LO-RXPOWER Alarm

Verify that the trunk-rx port is cabled correctly and clean the fiber connecting the faulty TXP/MXP card to the drop port of the DWDM card.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## LO-RXPOWER Warn

Default Severity: Minor (MN), Non-Service-Affecting (NSA)

Logical Object: Software

The LO-RXPOWER warning is raised on the client or trunk optics controller when the measured individual lane optical signal power of the received signal falls below the default threshold.

### Clear the LO-RXPOWER Warn Alarm

Verify that the trunk-rx port is cabled correctly and clean the fiber connecting the faulty TXP/MXP card to the drop port of the DWDM card.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## LOS

Default Severity: Major

Logical Object: Software

This alarm occurs when there is a loss of signal

## Clear the LOS Alarm

---

Identify and correct the underlying cause of signal LOS. The alarm is cleared when signal is improved.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## LOS-P

Default Severity: Minor

Logical Object: OTN

This alarm occurs when there is a loss of signal.

## Clear the LOS-P Alarm

---

Identify and correct the underlying cause of signal LOS. The alarm is cleared when signal is improved.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## LO-TXPOWER Alarm

Default Severity: Minor (MN), Non-Service-Affecting (NSA)

Logical Object: Controller

The LO-TXPOWER alarm is raised on the client or trunk optics controller when the measured individual lane optical signal power of the transmitted signal falls below the default threshold.

## Clear the LO-TXPOWER Alarm

---

Verify the optics detection and any failures.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## LO-TXPOWER Warn

Default Severity: Minor (MN), Non-Service-Affecting (NSA)

Logical Object: Software



The LO-TXPOWER warning is raised on the client or trunk optics controller when the measured individual lane optical signal power of the transmitted signal falls below the default threshold.

## Clear the LO-TXPOWER Warn Alarm

---

Verify the optics detection and any failures.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## OOR\_CD

Default Severity: Minor

Logical Object: Controller

This alarm occurs when the Chromatic Dispersion is out of range

## Clear the OOR\_CD Alarm

---

Configure threshold value within range if CD value is not within the threshold range.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## OSNR Alarm

Default Severity: Minor (MN), Non-Service-Affecting (NSA)

Logical Object: Controller

The Optical Signal Noise Ratio (OSNR) alarm occurs when the measured OSNR falls below the threshold.

## Clear the OSNR Alarm

---

**Step 1** Verify the value of the minimum acceptable OSNR value of NCS 5500 using the show controller optics R/S/I/P command.

**Step 2** If the value is not within the OSNR threshold range, configure the minimum acceptable OSNR value using the controller optics R/S/I/P osnr-low-threshold command in the config mode. The range is 0–4000 (in units of 01db).

**Step 3** If the value is within the range of the minimum acceptable OSNR, contact TAC.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## UNC-WORD Alarm

Default Severity: Not Reported (NR), Not-Alarmed, Non-Service-Affecting (NSA)

Logical Object: OTN

The Uncorrected FEC Word (UNC-WORD) condition is raised when the FEC is unable to correct the frame.

### Clear the UNC-WORD Alarm

---

- Step 1** Ensure that the fiber connector for the card is completely plugged in.
- Step 2** Ensure that the ports on the far end and near end nodes have the same port rates and FEC settings.
- Step 3** If the BER threshold is correct and at the expected level, use an optical test set to measure the power level of the line to ensure it is within guidelines. For specific procedures to use the test set equipment, consult the manufacturer.
- Step 4** If the optical power level is good, verify that the optical receive levels are within the acceptable range.
- Step 5** If receive levels are good, clean the fibers at both ends.
- Step 6** If the condition does not clear, verify that a single-mode fiber is used.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---

## WVL-00L

Default Severity: Major

Logical Object: Controller

The Wavelength Out of Lock alarm is raised when the port detects the optical input frequency to be out of range.

### Clear the WVL-00L Alarm

---

- Step 1** Verify the wavelength configuration.
- Step 2** Verify whether the pluggable is inserted properly.

If the alarm does not clear, log into the Technical Support Website at <http://www.cisco.com/c/en/us/support/index.html> for more information or call Cisco TAC (1 800 553-2447).

---



## CHAPTER 14

# Configuring Controllers

This chapter describes the Optics Controller and Coherent DSP Controller for the 6-port Coherent Line Card (NC55-6X200-DWDM-S). This chapter also describes the procedures used to configure the controllers.



---

**Note** When you plan to replace a configured optical module with a different type of optical module, you must clear the configurations of the old module before installing the new optical module.

---



---

**Note** When two MACsec enabled Cisco NCS 5500 routers with Coherent Line Cards are connected, there is no compatibility between Coherent Line Cards of IOS XR Release version 6.5.x (or lower) and 6.6.1 (or higher).

---

- [Optics Controllers, on page 307](#)
- [Maintenance Mode, on page 308](#)
- [Performance Monitoring, on page 309](#)
- [How to Configure Controllers, on page 309](#)
- [Verify Controller Details, on page 321](#)
- [Replace Optical Module, on page 322](#)

## Optics Controllers

Controllers are represented in the *rack/slot/instance/port* format (*r/s/i/p*); for example, 0/3/0/1. Each port has an optics controller that is created on startup.



---

**Note** You must shut down the optics controller before you perform any of the following tasks:

- Configure the controller
- Restore a saved configuration
- Upgrade the DSP processor or CFP2 optics module Field Programmable Device (FPD)

---

### CFP2 DCO Optics Support

There are two hardware versions of the CFP DCO optics (A0 and B0). You can identify the version A0 and B0 using a show coherent driver internal location 0/0/CPU0 command and looking at "VID".

A0 = V01

B0 = V02

The CFP2 DCO version A0 optics support the following traffic types:

| Traffic Type Index | Speed | Modulation | Forward Error Correction | Differential |
|--------------------|-------|------------|--------------------------|--------------|
| 1                  | 100G  | qpsk       | 15sdfec                  | disable      |
| 2                  | 100G  | qpsk       | 15sdfecde                | enable       |
| 3                  | 200G  | 16qam      | 15sdfec                  | disable      |
| 4                  | 200G  | 8qam       | 15sdfec                  | disable      |

The CFP2 DCO version B0 optics support the following traffic-types:

| Traffic Type Index | Speed | Modulation | Forward Error Correction | Differential |
|--------------------|-------|------------|--------------------------|--------------|
| 1                  | 100G  | qpsk       | 15sdfec                  | disable      |
| 2                  | 100G  | qpsk       | 15sdfecde                | enable       |
| 3                  | 100G  | qpsk       | otu7staircase            | enable       |
| 4                  | 200G  | 16qam      | 15sdfec                  | disable      |
| 5                  | 200G  | 8qam       | 15sdfec                  | disable      |

The 100G/Staircase FEC traffic-type is supported with CFP2 DCO version B0 optics.

## Maintenance Mode

Coherent DSP controllers can be placed in maintenance mode. Use the **controller coherentDSP secondary-admin-state maintenance** command to place controllers in maintenance mode.

Use the **show controllers optics r/s/i/p** command to view optics parameter values, laser state, controller state, admin state, and trunk alarms on the card, and threshold values for the different optics parameters.

Use the **show controllers coherentDSP r/s/i/p** command to view the DSP controller state and alarm status and statistics.




---

**Note** In maintenance mode, all alarms are suppressed and the **show alarms** command does not display alarm details. However, traffic is not affected in maintenance mode.

---



---

**Note** The FEC is disabled for 25G and 50G optics in NC57-MPA-12L-S MPA when connected on 55A2-MOD-SE-S/-SE-H-S router, and in Line card NC57-MOD-S while verifying the FEC status using **show controllers { TwentyfiveGigE | FiftyGigE }**

---

## Performance Monitoring

Performance monitoring (PM) parameters are used by service providers to gather, store, set thresholds for, and report performance data for early detection of problems. The user can retrieve both current and historical PM counters for the various controllers in 30-second, 15-minute, and 24-hour intervals.

PM for optical parameters include input signal power and transmit power, optical signal-to-noise ratio, chromatic dispersion, polarization dependent loss, second order polarization mode dispersion, differential group delay, and transmitter laser bias current.

PM for DSP parameters include:

- FEC: error corrected bits, uncorrectable blocks, pre-FEC BER (block errors ratio)
- OTN: errored seconds, severely effected seconds, unavailable seconds, failed counts

These parameters simplify troubleshooting operations and enhance data that can be collected directly from the equipment.

## How to Configure Controllers

This section contains the following procedures:

### Configuring Optics Controller

You can configure parameters such as performance monitoring, high power threshold, and wavelength for Optics controller.

To configure the Optics controller, use the following commands:

#### Before you begin

You must shut down the optics controller before you perform any of the following tasks:

- Configure the controller
- Restore a saved configuration
- Upgrade the DSP processor or CFP2 optics module Field Programmable Device (FPD)

#### SUMMARY STEPS

1. **configure**
2. **controller optics** *r/s/i/p*

3. **shutdown**
4. **commit**
5. **rx-high-threshold** *rx-high*
6. **tx-high-threshold** *tx-high*
7. **no shutdown**
8. **commit**

## DETAILED STEPS

|               | Command or Action                                                                                                           | Purpose                                                                                                         |
|---------------|-----------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <b>configure</b><br><b>Example:</b><br>RP/0/RP0/CPU0:router# configure terminal                                             | Enters global configuration mode.                                                                               |
| <b>Step 2</b> | <b>controller optics</b> <i>r/s/i/p</i><br><b>Example:</b><br>RP/0/RP0/CPU0:router(config)# controller optics<br>0/3/0/1    | Enters optics controller configuration mode.                                                                    |
| <b>Step 3</b> | <b>shutdown</b><br><b>Example:</b><br>RP/0/RP0/CPU0:router(config-Optics)# shutdown                                         | Shuts down the optics controller.                                                                               |
| <b>Step 4</b> | <b>commit</b><br><b>Example:</b><br>RP/0/RP0/CPU0:router(config-Optics)# commit                                             | Saves the configuration changes to the running configuration file and remains within the configuration session. |
| <b>Step 5</b> | <b>rx-high-threshold</b> <i>rx-high</i><br><b>Example:</b><br>RP/0/RP0/CPU0:router(config-Optics)#<br>rx-high-threshold 200 | Configures the high receive power threshold. The range is -400 to 300 (in the units of 0.1 dBm).                |
| <b>Step 6</b> | <b>tx-high-threshold</b> <i>tx-high</i><br><b>Example:</b><br>RP/0/RP0/CPU0:router(config-Optics)#<br>tx-high-threshold 300 | Configures the high transmit power threshold. The range is -400 to 300 dBm (in the units of 0.1 dBm).           |
| <b>Step 7</b> | <b>no shutdown</b><br><b>Example:</b><br>RP/0/RP0/CPU0:router(config-Optics)# no shutdown                                   | Removes the shutdown configuration on the optics controller.                                                    |

|               | Command or Action                                                                          | Purpose                                                                                                         |
|---------------|--------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|
| <b>Step 8</b> | <b>commit</b><br><b>Example:</b><br><pre>RP/0/RP0/CPU0:router(config-Optics)# commit</pre> | Saves the configuration changes to the running configuration file and remains within the configuration session. |



**Note** When you bring up the local optics controller, you might briefly see transient loss of signal (LOS) alarms on the console. This behavior might be observed during the initial tuning of the channel.

```
PKT_INFRA-FM-2-FAULT_CRITICAL : ALARM_CRITICAL :LOS-P :DECLARE :CoherentDSP0/3/0/1:
PKT_INFRA-FM-2-FAULT_CRITICAL : ALARM_CRITICAL :LOS-P :CLEAR :CoherentDSP0/3/0/1:
```

During the laser-on process, you might briefly see transient loss of line (LOL) alarms on the console. This alarm is cleared when the laser-on process is complete.

```
PKT_INFRA-FM-3-FAULT_MAJOR : ALARM_MAJOR :CTP2 RX LOL :DECLARE ::
PKT_INFRA-FM-3-FAULT_MAJOR : ALARM_MAJOR :CTP2 RX LOL :CLEAR ::
```

The laser-on process can take up to 120 seconds to complete.

## Restrictions and Usage Guidelines for Port Modes

This section provides the restrictions and usage guidelines for the supported port modes.

Table 37: Restrictions and Usage Guidelines for Port Modes

| Port Mode | Usage Guidelines                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | Restrictions                                                                                                                                                                                                                                                         |
|-----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 25Gbps    | <ul style="list-style-type: none"> <li>• NCS-55A2-MOD-S</li> <li>• NCS-55A2-MOD-HD-S</li> <li>• NCS-55A2-MOD-SE-S</li> <li>• NC55A2-MOD-SE-H-S</li> <li>• NCS-55A2-MOD-HX-S</li> <li>• NCS-55A1-48Q6H</li> <li>• NCS-55A1-24Q6H-S</li> <li>• NC55-MOD-A-SE-S</li> <li>• NC55-MOD-A-S</li> <li>• NC55-32T16Q4H-A</li> <li>• N540-24Z8Q2C-M</li> <li>• N540X-ACC-SYS</li> <li>• N540-ACC-SYS</li> <li>• N540-28Z4C-SYS</li> </ul> <ul style="list-style-type: none"> <li>• The 25Gbps mode is divided into four quads (0-3). Each quad consists of the following ports: <ul style="list-style-type: none"> <li>• Quad 0 - Ports 24-27</li> <li>• Quad 1 - Ports 28-31</li> <li>• Quad 2 - Ports 32-35</li> <li>• Quad 3 - Ports 36-39</li> </ul> </li> <li>• 10Gbps mode supports both 1Gbps and 10Gbps port speed.</li> </ul> | <ul style="list-style-type: none"> <li>• 25Gbps mode is the default mode set on the quad.</li> <li>• Port speeds of 1Gbps and 10Gbps are incompatible with a 25Gbps port speed within the same quad. They cannot be configured to operate simultaneously.</li> </ul> |

## Configure Port Mode Speed

Each port on the 6-port Coherent Line Card can support 100 Gbps (DWDM QPSK), 150Gbps (DWDM 8 QAM), or 200Gbps (DWDM 16 QAM) WDM signals.





**Note** You might rarely see up to five syslog messages mentioning that the recovery mechanism got triggered to recover the port. These messages are about a port in down state due to auto-negotiation mismatch with the peer port and other port-down scenarios. You can ignore such syslog messages as they will not affect the functionality of the ports.



**Note** The line card has three Digital Signal Processors (DSPs), one for each pair of ports:

- Ports 0 and 1 – DSP0
- Ports 2 and 3 – DSP1
- Ports 4 and 5 – DSP2

When you configure the port-mode speed for 150Gbps (8 QAM), the port pairs belonging to a DSP are coupled. Ensure that you configure the port-mode speed on each port of the port pair that belongs to the same DSP.

To configure the port mode speed, use the following commands:

#### Before you begin

Ensure that you shut down the controller before you configure the controller or restore a saved configuration.

#### SUMMARY STEPS

1. **configure**
2. **controller optics** *r/s/i/p*
3. **shutdown**
4. **commit**
5. **port-mode speed** { 100G | 150G | 200G } **mod** { 16qam | 8qam | qpsk } **fec** { 15sdfec | 15sdfecde | 25sdfec | otu7staircase } **diff** { enable | disable }
6. **no shutdown**
7. **commit**

#### DETAILED STEPS

|        | Command or Action                                                                                                               | Purpose                                     |
|--------|---------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------|
| Step 1 | <b>configure</b><br><b>Example:</b><br>RP/0/RP0/CPU0:router# <b>configure</b>                                                   | Enters global configuration mode.           |
| Step 2 | <b>controller optics</b> <i>r/s/i/p</i><br><b>Example:</b><br>RP/0/RP0/CPU0:router(config)# <b>controller optics</b><br>0/3/0/0 | Enters optics controller configuration mode |

|               | Command or Action                                                                                                                                                                                                                                                          | Purpose                                                                                                         |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|
| <b>Step 3</b> | <b>shutdown</b><br><b>Example:</b><br>RP/0/RP0/CPU0:router(config-Optics)# <b>shutdown</b>                                                                                                                                                                                 | Shuts down the optics controller.                                                                               |
| <b>Step 4</b> | <b>commit</b><br><b>Example:</b><br>RP/0/RP0/CPU0:router(config-Optics)# <b>commit</b>                                                                                                                                                                                     | Saves the configuration changes to the running configuration file and remains within the configuration session. |
| <b>Step 5</b> | <b>port-mode speed { 100G   150G   200G } mod { 16qam   8qam   qpsk } fec { 15sdfec   15sdfecde   25sdfec   otu7staircase } diff { enable   disable }</b><br><b>Example:</b><br>RP/0/RP0/CPU0:router(config-Optics)# <b>port-mode speed 100G mod qpsk fec 15sdfec diff</b> | Configures the port mode speed.                                                                                 |
| <b>Step 6</b> | <b>no shutdown</b><br><b>Example:</b><br>RP/0/RP0/CPU0:router(config-Optics)# <b>no shutdown</b>                                                                                                                                                                           | Removes the shutdown configuration on the optics controller.                                                    |
| <b>Step 7</b> | <b>commit</b><br><b>Example:</b><br>RP/0/RP0/CPU0:router(config-Optics)# <b>commit</b>                                                                                                                                                                                     | Saves the configuration changes to the running configuration file.                                              |



**Note** When you bring up the local optics controller, you might briefly see transient loss of signal (LOS) alarms on the console. This behavior might be observed during the initial tuning of the channel.

```
PKT_INFRA-FM-2-FAULT_CRITICAL : ALARM_CRITICAL :LOS-P :DECLARE :CoherentDSP0/3/0/1:
PKT_INFRA-FM-2-FAULT_CRITICAL : ALARM_CRITICAL :LOS-P :CLEAR :CoherentDSP0/3/0/1:
```

During the laser-on process, you might briefly see transient loss of line (LOL) alarms on the console. This alarm clears when the laser-on process is complete.

```
PKT_INFRA-FM-3-FAULT_MAJOR : ALARM_MAJOR :CTP2 RX LOL :DECLARE ::
PKT_INFRA-FM-3-FAULT_MAJOR : ALARM_MAJOR :CTP2 RX LOL :CLEAR ::
```



**Note** On NCS-55A2-MOD-S and NC55-MOD-A-S with CFP2-DCO optics:

- During the laser-on process, you might briefly see Optical Transport Network (OTN) alarms on the console. This alarm clears when the laser-on process is complete.

```
PKT_INFRA-FM-6-FAULT_INFO : OTUK-BDI :DECLARE :CoherentDSP0/0/2/2:
PKT_INFRA-FM-6-FAULT_INFO : OTUK-BDI :CLEAR :CoherentDSP0/0/2/2:
```

- During the laser-on process, you might briefly see transient transmit power and receive power alarms on the console. These alarms are cleared when the laser-on process is complete.

```
PKT_INFRA-FM-4-FAULT_MINOR : ALARM_MINOR :LO-RXPOWER :DECLARE :Optics0/0/2/0:
PKT_INFRA-FM-4-FAULT_MINOR : ALARM_MINOR :LO-TXPOWER :DECLARE :Optics0/0/2/0:
PKT_INFRA-FM-4-FAULT_MINOR : ALARM_MINOR :HI-RXPOWER :DECLARE :Optics0/0/2/0:
```

```
PKT_INFRA-FM-4-FAULT_MINOR : ALARM_MINOR :LO-RXPOWER :CLEAR :Optics0/0/2/0:
PKT_INFRA-FM-4-FAULT_MINOR : ALARM_MINOR :HI-RXPOWER :CLEAR :Optics0/0/2/0:
PKT_INFRA-FM-4-FAULT_MINOR : ALARM_MINOR :LO-TXPOWER :CLEAR :Optics0/0/2/0:
```

- When you bring up the local optics controller, you might see repeated remote faults on the console.

```
PLATFORM-DPA-2-RX_FAULT : Interface HundredGigE0/0/2/2/0, Detected Remote Fault
PLATFORM-DPA-2-RX_FAULT : Interface HundredGigE0/0/2/2/1, Detected Remote Fault
PLATFORM-DPA-2-RX_FAULT : Interface HundredGigE0/0/2/2/0, Detected Local Fault
PLATFORM-DPA-2-RX_FAULT : Interface HundredGigE0/0/2/2/1, Detected Local Fault
PLATFORM-DPA-2-RX_FAULT : Interface HundredGigE0/0/2/2/0, Detected Remote Fault
PLATFORM-DPA-2-RX_FAULT : Interface HundredGigE0/0/2/2/1, Detected Remote Fault
```

If you need to change the port-mode speed, ensure that you remove the existing port mode speed configuration by entering the **no port-mode** command. You can then change the port mode speed.

The following example shows how to change the port mode speed to 100Gbps.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# controller optics 0/3/0/0
RP/0/RP0/CPU0:router(config-Optics)# shutdown
RP/0/RP0/CPU0:router(config-Optics)# commit
RP/0/RP0/CPU0:router(config-Optics)# no port-mode
RP/0/RP0/CPU0:router(config-Optics)# commit
RP/0/RP0/CPU0:router(config-Optics)# port-mode speed 100G mod qpsk fec 15sdfec diff enable
RP/0/RP0/CPU0:router(config-Optics)# commit
RP/0/RP0/CPU0:router(config-Optics)# no shutdown
RP/0/RP0/CPU0:router(config-Optics)# commit
RP/0/RP0/CPU0:router(config-Optics)# exit
RP/0/RP0/CPU0:router(config)#
```

**To modify the default 25Gbps mode into 10Gbps mode, perform the below configuration:**

Before Cisco IOS XR Release 7.5.1:

```
RP/0/RP0/CPU0:router(config)# hw-module quad 0 location 0/0/CPU0
RP/0/RP0/CPU0:router(config-quad-0x0)# mode 10g
```

From Cisco IOS XR Release 7.5.1:

```
RP/0/RP0/CPU0:router(config)# hw-module quad 0 location 0/0/CPU0 instance 1 mode 10g
RP/0/RP0/CPU0:router(config-quad-0x0)# mode 10g
```




---

**Note** A quad number always starts from 0 to the maximum supported number. The number of quads supported varies from platform to platform and the CLI validates it. For example, the NCS 540 Series Router supports two quads (0 and 1). If you enter X=3, the CLI returns an error.

---

Here, `instance` indicates the MPA card instance. It can range from 0-5. For Cisco NCS 540 Series Routers, it is always 0. Whereas, for Cisco NCS 5500 Series Routers, the instance can be between 0-5, adding 1 for every MPA instance. The default value is 0.

Starting with Cisco IOS XR Release 24.2.1, you can configure 50Gpbs mode only on NC-57-48Q2D-S and NC-57-48Q2D-SE-S line cards.




---

**Note** A quad number always starts from 0 to the maximum supported number. The number of quads supported varies from platform to platform and the CLI validates it. For example, the NCS 540 Series Router supports two quads (0 and 1). If you enter X=3, the CLI returns an error.

---

Here, `instance` indicates the MPA card instance. It can range from 0-5. For Cisco NCS 540 Series Routers, it is always 0. Whereas, for Cisco NCS 5500 Series Routers, the instance can be between 0-5, adding 1 for every MPA instance. The default value is 0.




---

**Note** To revert to the default 25Gbps mode, use the `no` form of the `hw-module quad` command.

---

After you configure the port-mode speed, you can configure the following interfaces:

- 100G – Each optics controller configuration creates a single 100GE port:
  - **interface HundredGigE** `r/s/i/p/0` (where `p` = CTP2 port 0-5)
    - `0/3/0/0/0`
    - `0/3/0/1/0`
    - `0/3/0/2/0`
    - `0/3/0/3/0`
    - `0/3/0/4/0`
    - `0/3/0/5/0`
- 200G – Each optics controller configuration creates two 100GE ports:
  - **interface HundredGigE** `r/s/i/p/0`, `r/s/i/p/1` (where `p` = CTP2 port 0-5)
    - `0/3/0/0/0`, `0/3/0/0/1`
    - `0/3/0/1/0`, `0/3/0/1/1`
    - `0/3/0/2/0`, `0/3/0/2/1`
    - `0/3/0/3/0`, `0/3/0/3/1`
    - `0/3/0/4/0`, `0/3/0/4/1`

0/3/0/5/0, 0/3/0/5/1

- 150G (coupled) – Coupled optics controller configuration creates three 100GE port:
  - **interface HundredGigE** *r/s/i/p/0, r/s/i/p/1, r/s/i/p+1/0* (where *p* = CTP2 port: 0, 2, 4 [port *p* and *p* +1 are coupled])
    - 0/3/0/0/0, 0/3/0/0/1, 0/3/0/1/0
    - 0/3/0/2/0, 0/3/0/2/1, 0/3/0/3/0
    - 0/3/0/4/0, 0/3/0/4/1, 0/3/0/5/0

For more information, see the *Configuring Ethernet Interfaces* chapter.

## Configuring Wavelength

To configure wavelength, use the following commands:

### Before you begin

- Before configuring the wavelength, use the **show controllers optics** *r/s/i/p* **dwdm-carrier-map** command to display the wavelength and channel mapping for optics controllers.
- You must shut down the controller before you configure the controller or restore a saved configuration.

### SUMMARY STEPS

1. **configure**
2. **controller optics** *r/s/i/p*
3. **shutdown**
4. **commit**
5. **dwdm-carrier** {100MHz-grid frequency *frequency* } | {50GHz-grid [ frequency *frequency* | channel-number ] }
6. **no shutdown**
7. **commit**

### DETAILED STEPS

|        | Command or Action                                                                                                               | Purpose                                      |
|--------|---------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------|
| Step 1 | <b>configure</b><br><b>Example:</b><br>RP/0/RP0/CPU0:router# <b>configure</b>                                                   | Enters global configuration mode.            |
| Step 2 | <b>controller optics</b> <i>r/s/i/p</i><br><b>Example:</b><br>RP/0/RP0/CPU0:router(config)# <b>controller optics</b><br>0/3/0/1 | Enters optics controller configuration mode. |

|               | Command or Action                                                                                                                                                                                                            | Purpose                                                                                                         |
|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|
| <b>Step 3</b> | <b>shutdown</b><br><b>Example:</b><br>RP/0/RP0/CPU0:router(config-Optics)# <b>shutdown</b>                                                                                                                                   | Shuts down the optics controller.                                                                               |
| <b>Step 4</b> | <b>commit</b><br><b>Example:</b><br>RP/0/RP0/CPU0:router(config-Optics)# <b>commit</b>                                                                                                                                       | Saves the configuration changes to the running configuration file and remains within the configuration session. |
| <b>Step 5</b> | <b>dwdm-carrier {100MHz-grid frequency frequency }   {50GHz-grid [ frequency frequency   channel-number ] }</b><br><b>Example:</b><br>RP/0/RP0/CPU0:router(config-Optics)# <b>dwdm-carrier 100MHz-grid frequency 1960875</b> | Configures the frequency on the trunk port.                                                                     |
| <b>Step 6</b> | <b>no shutdown</b><br><b>Example:</b><br>RP/0/RP0/CPU0:router(config-Optics)# <b>no shutdown</b>                                                                                                                             | Removes the shutdown configuration on the optics controller.                                                    |
| <b>Step 7</b> | <b>commit</b><br><b>Example:</b><br>RP/0/RP0/CPU0:router(config-Optics)# <b>commit</b>                                                                                                                                       | Saves the configuration changes to the running configuration file and remains within the configuration session. |

To configure a DWDM carrier with the required frequency:

```
RP/0/RP0/CPU0:router#config
RP/0/RP0/CPU0:router(config)#controller Optics0/3/0/0
RP/0/RP0/CPU0:router(config-Optics)#dwdm-carrier
RP/0/RP0/CPU0:router(config-Optics)#dwdm-carrier 100MHz-grid
RP/0/RP0/CPU0:router(config-Optics)#dwdm-carrier 100MHz-grid frequency
RP/0/RP0/CPU0:router(config-Optics)#dwdm-carrier 100MHz-grid frequency 1960625
```

The output of show run controller optics 0/3/0/0 command is:

```
RP/0/RP0/CPU0:router#show run controller optics 0/3/0/0
Wed Nov 6 13:47:33.178 UTC
controller Optics0/3/0/0
transmit-power -7
port-mode speed 100G mod qpsk fec 25sdfec diff disable
dwdm-carrier 100MHz-grid frequency 1960625
```



**Note** When you bring up the local optics controller, you might briefly see transient loss of signal (LOS) alarms on the console. This behavior might be observed during the initial tuning of the channel.

```
PKT_INFRA-FM-2-FAULT_CRITICAL : ALARM_CRITICAL :LOS-P :DECLARE :CoherentDSP0/3/0/1:
PKT_INFRA-FM-2-FAULT_CRITICAL : ALARM_CRITICAL :LOS-P :CLEAR :CoherentDSP0/3/0/1:
```

During the laser-on process, you might briefly see transient loss of line (LOL) alarms on the console. This alarm is cleared when the laser-on process is complete.

```
PKT_INFRA-FM-3-FAULT_MAJOR : ALARM_MAJOR :CTP2 RX LOL :DECLARE ::
PKT_INFRA-FM-3-FAULT_MAJOR : ALARM_MAJOR :CTP2 RX LOL :CLEAR ::
```

## Configuring Coherent DSP Controller

You can configure the administrative state for the Coherent DSP controller. To configure the Coherent DSP controller, use the following commands.



**Note** The coherent DSP controller doesn't support Q factor, Q margin, and post FEC BER reporting. Therefore, no threshold crossing alert (TCA) is raised for these parameters.

### SUMMARY STEPS

1. **configure**
2. **controller coherentDSP** *r/s/i/p*
3. **secondary-admin-state** *admin-state*
4. **commit**

### DETAILED STEPS

|        | Command or Action                                                                                                                      | Purpose                                                                                                    |
|--------|----------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------|
| Step 1 | <b>configure</b><br><b>Example:</b><br>RP/0/RP0/CPU0:router# <b>configure</b>                                                          | Enters global configuration mode.                                                                          |
| Step 2 | <b>controller coherentDSP</b> <i>r/s/i/p</i><br><b>Example:</b><br>RP/0/RP0/CPU0:router(config)# <b>controller coherentDSP</b> 0/3/0/1 | Enters Coherent DSP optics controller configuration mode.                                                  |
| Step 3 | <b>secondary-admin-state</b> <i>admin-state</i><br><b>Example:</b>                                                                     | Configures the administrative state of the controller indicating that the controller is under maintenance. |

|               | Command or Action                                                                           | Purpose                                                                                                         |
|---------------|---------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|
|               | RP/0/RP0/CPU0:router (config-CoDSP) #<br><b>secondary-admin-state maintenance</b>           |                                                                                                                 |
| <b>Step 4</b> | <b>commit</b><br><b>Example:</b><br><br>RP/0/RP0/CPU0:router (config-CoDSP) # <b>commit</b> | Saves the configuration changes to the running configuration file and remains within the configuration session. |

## Configuring Performance Monitoring

You can configure the performance monitoring parameters for the optics and Coherent DSP controllers. To configure PM parameters, use the following commands.

### SUMMARY STEPS

1. **configure**
2. **controller** { **optics** | **coherentDSP** } *r/s/i/p*
3. **pm** { **30-sec** | **15-min** | **24-hour** } { **optics** | **fec** | **otn** } [ **report** | **threshold value** ]
4. **commit**

### DETAILED STEPS

|               | Command or Action                                                                                                                                                                                                                                                  | Purpose                                                                                                         |
|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <b>configure</b><br><b>Example:</b><br><br>RP/0/RP0/CPU0:router# <b>configure</b>                                                                                                                                                                                  | Enters global configuration mode.                                                                               |
| <b>Step 2</b> | <b>controller</b> { <b>optics</b>   <b>coherentDSP</b> } <i>r/s/i/p</i><br><b>Example:</b><br><br>RP/0/RP0/CPU0:router (config) # <b>controller</b><br><b>coherentDSP 0/3/0/1</b>                                                                                  | Enters optics or Coherent DSP controller configuration mode.                                                    |
| <b>Step 3</b> | <b>pm</b> { <b>30-sec</b>   <b>15-min</b>   <b>24-hour</b> } { <b>optics</b>   <b>fec</b>   <b>otn</b> } [ <b>report</b>   <b>threshold value</b> ]<br><b>Example:</b><br><br>RP/0/RP0/CPU0:router (config-CoDSP) # <b>pm 15-min otn</b><br><b>threshold es-ne</b> | Configures the performance monitoring parameters.                                                               |
| <b>Step 4</b> | <b>commit</b><br><b>Example:</b><br><br>RP/0/RP0/CPU0:router (config-CoDSP) # <b>commit</b>                                                                                                                                                                        | Saves the configuration changes to the running configuration file and remains within the configuration session. |



## Verify Controller Details

Execute the **show controllers coherentDSP** command to display status and configuration information for interfaces configured as coherent DSP controllers.

```
Router#show controllers coherentDSP 0/0/0/13
Thu May 27 06:56:37.505 UTC

Port : CoherentDSP 0/0/0/13
Controller State : Up
Inherited Secondary State : Normal
Configured Secondary State : Normal
Derived State : In Service
Loopback mode : None
BER Thresholds : SF = 1.0E-5 SD = 1.0E-7
Performance Monitoring : Enable
Bandwidth : 400.0Gb/s

Alarm Information:
LOS = 32 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0 SF_BER = 0
SD_BER = 0 BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0 FLEXO_GIDM = 0
FLEXO-MM = 0 FLEXO-LOM = 0 FLEXO-RDI = 0
FLEXO-LOF = 43
Detected Alarms : None

Bit Error Rate Information
PREFEC BER : 8.5E-04
POSTFEC BER : 0.0E+00
Q-Factor : 9.90 dB

Q-Margin : 2.70dB

OTU TTI Received
```

Execute the **show controllers optics** command to display status and configuration information about the interfaces configured as optics controller.

```
Router#show controllers optics 0/0/0/7
Controller State: Up
Transport Admin State: In Service
Laser State: On
LED State: Green
FEC State: FEC ENABLED
Optics Status
 Optics Type: QSFPDD 400G ZR
 DWDM carrier Info: C BAND, MSA ITU Channel=61, Frequency=193.10THz,
 Wavelength=1552.524nm
 Alarm Status:

 Detected Alarms: None
 LOS/LOL/Fault Status:
 Alarm Statistics:

 HIGH-RX-PWR = 0 LOW-RX-PWR = 0
 HIGH-TX-PWR = 0 LOW-TX-PWR = 0
 HIGH-LBC = 0 HIGH-DGD = 0
 OOR-CD = 0 OSNR = 55
 WVl-OOL = 0 MEA = 0
 IMPROPER-REM = 0
```

```

TX-POWER-PROV-MISMATCH = 0
Laser Bias Current = 0.0
Actual TX Power = -8.16 dBm
RX Power = -7.85 dBm
RX Signal Power = -7.55 dBm
Frequency Offset = 5 MHz
Performance Monitoring: Enable
THRESHOLD VALUES

Parameter High Alarm Low Alarm High Warning Low Warning

Rx Power Threshold(dBm) 1.9 -28.2 0.0 -25.0
Tx Power Threshold(dBm) 0.0 -15.0 -2.0 -16.0
LBC Threshold(mA) 0.00 0.00 0.00 0.00
Temp. Threshold(celsius) 80.00 -5.00 75.00 15.00
Voltage Threshold(volt) 3.46 3.13 3.43 3.16
LBC High Threshold = 98 %
Configured Tx Power = -6.00 dBm
Configured CD High Threshold = 80000 ps/nm
Configured CD lower Threshold = -80000 ps/nm
Configured OSNR lower Threshold = 9.00 dB
Configured DGD Higher Threshold = 80.00 ps
Baud Rate = 59.8437500000 GBd
Modulation Type: 16QAM
Chromatic Dispersion 2 ps/nm
Configured CD-MIN -2400 ps/nm CD-MAX 2400 ps/nm
Second Order Polarization Mode Dispersion = 87.00 ps^2
Optical Signal to Noise Ratio = 36.30 dB
Polarization Dependent Loss = 0.40 dB
Polarization Change Rate = 0.00 rad/s
Differential Group Delay = 2.00 ps
Temperature = 51.00 Celsius
Voltage = 3.36 V
Transceiver Vendor Details
Form Factor : QSFP-DD
Optics type : QSFPDD 400G ZR
Name : CISCO-ACACIA
OUI Number : 7c.b2.5c
Part Number : DP04QSDD-E20-19E
Rev Number : 10
Serial Number : ACA2449003P
PID : QDD-400G-ZR-S
VID : ES03
Firmware Version : 61.12
Date Code(yy/mm/dd) : 20/12/03

```

## Replace Optical Module

In this example, we are replacing QSFP-100G-SR4-S QSFP optics configured for 4x25 breakout with QSFP-40G-SR4 optics and configure it for 4x10 breakout.

1. Delete the optical module configuration using the **no breakout** command.

```

Router# configure
Router(config)# controller optics 0/2/0/35
Router(config-Optics)# no breakout 4x25
Router(config-Optics)# commit

```

2. Replace the QSFP-100G-SR4-S QSFP optical module with QSFP-40G-SR4 optical module.
3. Configure 4x10 breakout for QSFP-40G-SR4 optical module.

```
Router# configure
Router(config)# controller optics 0/2/0/35
Router(config-Optics)# breakout 4x10
Router(config-Optics)# commit
```





## CHAPTER 15

# Global Navigation Satellite System

---

This chapter describes the Global Navigation Satellite System (GNSS) NCS-55A2-MOD-SE-S Line Card. This chapter also describes the procedures used to configure the GNSS port.

This chapter describes the Global Navigation Satellite System (GNSS) and the procedures used to configure the GNSS port for fixed-port routers.

- [Configuring the Global Navigation Satellite System, on page 326](#)
- [Information About GNSS, on page 326](#)
- [Configure GNSS, on page 329](#)

# Configuring the Global Navigation Satellite System

Table 38: Feature History Table

| Feature Name                                                                          | Release       | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|---------------------------------------------------------------------------------------|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Extending GNSS Functionality to Cisco Network Convergence System 5700 Series variants | Release 7.5.1 | <p>A Global Navigation Satellite System (GNSS) receiver receives radio signals from GNSS satellites and decodes the information from multiple satellites to determine its distance from each satellite. Based on this data, the GNSS receiver identifies the location of each satellite.</p> <p>This feature is supported on the following variants of Cisco Network Convergence System 5700 Series fixed chassis:</p> <ul style="list-style-type: none"> <li>• NCS-57B1-6D24-SYS</li> <li>• NCS-57B1-5DSE-SYS</li> </ul> <p>For more information about the Cisco NCS 5700 series fixed chassis, see the <a href="#">Cisco NCS-57B1 Fixed Chassis Data Sheet</a>.</p> |

In typical telecom networks, synchronization works in a hierarchal manner where the core network is connected to a stratum-1 clock. The stratum-1 clock is then distributed along the network in a tree-like structure. However, with a GNSS receiver, clocking is changed to a flat architecture, where access networks can directly take clock from satellites in sky by using an on-board GPS chip.

To optimize the GNSS system, it requires all the systems to share a common time scale and coordinated system. If all the systems do not have a common time, the receiver sees a time offset and then the receiver will have to select only one constellation having common time scale. Then there will be a requirement to add more satellites to increase the coverage of the constellation itself.

This capability simplifies network synchronization planning, provides flexibility and resilience in resolving network synchronization issues in the hierarchical network.

These Cisco IOS XR routers now support on board GNSS receiver to recover time.

## Information About GNSS

### Overview of GNSS

The following routers support the GNSS receiver:

- NCS-55A2-MOD-S
- NCS-55A2-MOD-HD-S
- NCS-55A2-MOD-HX-S
- NCS-55A2-MOD-SE-S

Starting from Cisco IOS XR Release 7.5.1, the following 57B1 variants of NCS 5700 fixed chassis support GNSS receiver:

- NCS-57B1-6D24-SYS
- NCS-57B1-5DSE-SYS

No license is required to enable the GNSS module. The GNSS LED on the front panel indicates the status of the module. The following table describes the different status of GNSS LED:

| LED Status | Description                              |
|------------|------------------------------------------|
| Green      | GNSS NormalState.Selfsurvey is complete. |
| Amber      | All other states                         |

The following table describes the different status of GNSS LED for 5700 Series variants:

| LED Status | Description                                   |
|------------|-----------------------------------------------|
| Green      | GNSS is Locked to an RF input (Satellites)    |
| Amber      | GNSS Receiver not Locked RF input(Satellites) |
| OFF        | GNSS not configured                           |

NCS-57B1-6D24-SYS and NCS-57B1-5DSE-SYS can also receive 1PPS, 10 MHz, and ToD signals from an external clocking and timing source. However, the timing signals from the GNSS module (when enabled) take precedence over those of the external source. For more information, see the [Cisco NCS 5700 series: NCS-57B1 Fixed Chassis Data Sheet](#).



**Note** NCS-55A2-MOD-SE-S can also receive 1PPS, 10 MHz, and ToD signals from an external clocking and timing source. However, the timing signals from the GNSS module (when enabled) take precedence over those of the external source.



**Note** We do not recommend that you configure both the front panel (10M, 1PPS and ToD) input configuration and the GNSS input configuration.

By default, anti-jamming is enabled on the GNSS module.

## Operation of GNSS Module

The GNSS module has the following stages of acquiring and providing timing signals to the Cisco router:

- Self-survey mode - When the router is reset, the GNSS module comes up in self-survey mode. It tries to lock on to a minimum of four different satellites and computes approximately 2000 different positions of the satellites to obtain a 3-D location (Latitude, Longitude, and Height) of its current position. This operation takes about 35 to 40 minutes. During this stage also, the module is able to generate accurate timing signals and achieve a Normal or Phase-locked state.
- Over determined clock mode - The router switches to over determined (OD) mode when the self-survey mode is complete and the position information is stored in non-volatile memory on the router. In this mode, the module only processes the timing information based on satellite positions captured in self-survey mode.

The router saves the tracking data, which is retained even when the router is reloaded.

The GNSS module stays in the OD mode unless one of the following conditions occur:

- A position relocation of the antenna of more than 100 meters is detected. This detection causes an automatic restart of the self-survey mode.
- A manual restart of the self-survey mode or when the stored reference position is deleted.
- A worst-case recovery option after a jamming-detection condition that cannot be resolved with other methods.

You can configure the GNSS module to automatically track any satellite or configure it to explicitly use a specific constellation. However, the module uses configured satellites only in the OD mode.




---

**Note** GLONASS and BeiDou satellites cannot be enabled simultaneously.

---

When the router is reloaded, it always comes up in the OD mode unless:

- The router is reloaded when the self-survey mode is in progress.
- The physical location of the router is changed to more than 100 m from its pre-reloaded condition.

When the system restarts GNSS self-survey by using the default `gnss slot R0/R1` command in config mode, the 10MHz, 1PPS, and ToD signals are not changed and remain up.

## Prerequisites for GNSS

To use GNSS, the antenna must see as much as possible from the sky. For proper timing, a minimum of four satellites must be locked. For more information, see the *Cisco NCS 5500 Series Router Hardware Installation Guide* or *Cisco NCS 5700 Series Router Hardware Installation Guide*.

## Restrictions for GNSS

- The GNSS module is not supported through SNMP; all configurations are performed through commands.
- The GNSS holdover performance is one microsecond in two hours of holdover after twelve hours of GNSS lock time.
- TDEV fails marginally on NCS-55A2-MOD-SE-S with GNSS input.



# Configure GNSS

## Configuration Example

You can configure any of the following constellation options for a router:

- GPS
- Galileo
- GLONASS
- BeiDou
- QZSS

Based on your configuration, the output displays the status of the GNSS receiver on the router models.

This section describes how you can configure GNSS for a router.

```
/* Enable the GNSS receiver and enter the gnss-receiver submode */
```

```
Router(config)# gnss-receiver 0 location 0/0/CPU0
Router(config-gnss)# frequency synchronization
Router(config-gnss-freqsync)# selection input
```

## Optional Configuration Example

```
Router(config)# gnss-receiver 0 location 0/0/CPU0
Router(config-gnss)# anti-jam disable
Router(config-gnss)# constellation GPS
Router(config-gnss)# snr threshold 10
Router(config-gnss)# frequency synchronization
Router(config-gnss-freqsync)# selection input
Router(config-gnss-freqsync)# priority 5
Router(config-gnss-freqsync)# wait-to-restore 0
```

## Running Configuration

```
gnss-receiver 0 location 0/RP0/CPU0
frequency synchronization
 selection input
 priority 1
 wait-to-restore 0
 quality receive exact itu-t option 1 PRC
!
```

## Verification

The following is the output of the **show gnss-receiver** command on the router models.

```
show gnss-receiver
GNSS-receiver 0 location 0/RP0/CPU0
 Status: Available, Up
 Position: 741:12.12 N 4451:39.60 E 0.827km
```

```

Time: 2019:01:17 14:43:08 (UTC offset: 18s)
Firmware version: 1.4
Lock Status: Phase Locked, Receiver Mode: 3D-fix
Survey Progress: 100, Holdover Duration: 0
Major Alarm: Not used
Minor Alarm: Not used
Anti-jam: Enabled, Cable-delay compensation: 0
1PPS polarity: Positive
PDOP: 6.000, HDOP: 0.000, VDOP: 0.000, TDOP: 1.000
Constellation: GPS, Satellite Count: 10

```

```
Router# show gnss-receiver
```

```
Fri Jan 17 07:27:34.804 UTC
```

```
GNSS-receiver 0 location 0/RP0/CPU0
```

```
Status: Available, Up
```

```
Position: 12:56.18 N 77:41.77 E 0.823km
```

```
Time: 2020:01:17 07:31:41 (UTC offset: 0s)
```

```
Locked at: 2020:01:15 17:15:28
```

```
Firmware version: TIM 1.10
```

```
Lock Status: Phase Locked, Receiver Mode: Time fix only
```

```
Survey Progress: 100, Holdover Duration: Unknown
```

```
Major Alarms: Unknown
```

```
Minor Alarms: Unknown
```

```
Anti-jam: Enabled, Cable-delay compensation: 0
```

```
1PPS polarity: Positive
```

```
PDOP: 99.990, HDOP: 99.990, VDOP: 99.990, TDOP: 0.240
```

```
Constellation: GPS, Satellite Count: 17
```

```
Satellite Thresholds:
```

```
SNR - 0 dB-Hz, Elevation - 0 degrees, PDOP - 0, TRAIM - 0 us
```

```
Satellite Info:
```

```
CHN: Channel, AQUN: Aquisition, EPH: Ephemeris
```

| PRN<br>No. | CHN<br>No. | AQUN<br>Flag | EPH<br>Flag | SV<br>Type | Signal<br>Strength | Elevat'n | Azimuth |
|------------|------------|--------------|-------------|------------|--------------------|----------|---------|
| 1          | n/a        | On           | On          | GPS        | 44.000             | 19.000   | 220.000 |
| 3          | n/a        | On           | On          | GPS        | 48.000             | 62.000   | 299.000 |
| 4          | n/a        | On           | On          | GPS        | 46.000             | 30.000   | 338.000 |
| 7          | n/a        | On           | On          | GPS        | 47.000             | 9.000    | 261.000 |
| 8          | n/a        | On           | On          | GPS        | 41.000             | 17.000   | 172.000 |
| 9          | n/a        | On           | On          | GPS        | 44.000             | 7.000    | 317.000 |
| 11         | n/a        | On           | On          | GPS        | 42.000             | 10.000   | 202.000 |
| 14         | n/a        | On           | On          | GPS        | 42.000             | 22.000   | 90.000  |
| 16         | n/a        | On           | On          | GPS        | 46.000             | 66.000   | 59.000  |
| 22         | n/a        | On           | On          | GPS        | 47.000             | 71.000   | 238.000 |
| 23         | n/a        | On           | On          | GPS        | 46.000             | 27.000   | 332.000 |
| 26         | n/a        | On           | On          | GPS        | 48.000             | 40.000   | 40.000  |



# CHAPTER 16

## Configuring WAN-PHY Controllers

This module describes the configuration of WAN-PHY controllers on the Cisco NCS 5500 Series Routers.

**Table 39: Feature History for Configuring WAN-PHY Controllers**

| Release       | Modification                 |
|---------------|------------------------------|
| Release 7.2.1 | This feature was introduced. |

- [WAN-PHY Controller](#) , on page 331
- [Restrictions](#), on page 332
- [Configuring SONET Mode on an Interface](#), on page 332
- [Configuring SDH Mode on an Interface](#), on page 334
- [TSoP Smart SFP for SDH and SONET Encapsulation](#), on page 337
- [Prerequisites for TSoP](#), on page 338
- [Restrictions for TSoP](#), on page 338
- [Guidelines for TSoP Smart SFP](#), on page 338
- [De-jitter Buffer](#) , on page 339
- [Configuration for TSoP](#), on page 340

## WAN-PHY Controller

**Table 40: Feature History Table**

| Feature Name                                  | Release Information | Feature Description                                                                                                                                                                              |
|-----------------------------------------------|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| WAN-PHY SONET Controller                      | Release 7.2.1       | WAN-PHY renders 10 Gigabit Ethernet frames compatible with the SONET OC-192 or SDH STM-64 container format as defined by ANSI.<br><br>In this release WAN-PHY supports only SONET OC-192 format. |
| Support for SDH mode under WAN-PHY Controller | Release 7.2.2       | In this release, support of SDH STM-64 format is added for WAN-PHY Controllers. Support of SONET OC-192 formats existed in earlier releases.                                                     |

WAN-PHY renders 10 Gigabit Ethernet compatible with the SONET STS-192c and SDH STM-64 container format as defined by ANSI. WAN-PHY effectively bridges the asynchronous world of Ethernet data with synchronous SONET/SDH transport allowing 10 Gigabit Ethernet to be transparently carried over current DWDM networks without having to directly map the Ethernet frames into SONET/SDH.

To achieve this compatibility, a WAN Interface Sublayer (WIS) is inserted between the 10 Gigabit Ethernet Physical Coding Sublayer (PCS) and the serial Physical Medium Attachment sublayer/Physical Medium Dependent sublayer (PMA/PMD).

When the controller is in SONET mode the WIS sublayer transports 10 Gigabit Ethernet frames in an OC-192c SONET payload which can interoperate with SONET section or line level repeaters.

When the controller is in SDH mode the WIS sublayer transports 10 Gigabit Ethernet frames in an STM-64 payload which can interoperate with SDH section or line level repeaters.

WAN-PHY is supported on NC55-MPA-12T-S card and 10G pluggables.

This table lists modular line cards and 2-RU systems that support NC55-MPA-12T-S card:

**Table 41: Supported MOD Line Cards and 2-RU Systems:**

| Modular Line Card                                                                           | 2-RU Systems                                                                                                                                                                             |
|---------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"> <li>• NC55-MOD-A-S</li> <li>• NC55-MOD-A-SE-S</li> </ul> | <ul style="list-style-type: none"> <li>• NCS-55A2-MOD-S</li> <li>• NCS-55A2-MOD-SE-S</li> <li>• NCS-55A2-MOD-HX-S</li> <li>• NCS-55A2-MOD-SE-H-S</li> <li>• NCS-55A2-MOD-HD-S</li> </ul> |

## Restrictions

Consider these limitations before configuring WAN-PHY mode:

- WAN-PHY feature works on these 10G pluggables:
  - SFP-10G-SR-X
  - SFP-10G-LR-X
  - SFP-10G-ZR
- SONET or SDH configurations are rejected if the port has 1G optics.
- SONET or SDH configurations are rejected if MACsec is already configured on that port.

## Configuring SONET Mode on an Interface

This task describes how to configure WAN-PHY in the SONET mode on the NC55-MPA-12T-S.

To enable WAN-PHY in SONET mode on an interface, configure **port-mode sonet framing WIS** command in the controller optics mode:

```
controller Optics 0/0/1/10
 port-mode sonet framing WIS
!
```

### Verification

To verify the WHY-PHY SONET configuration, run these show commands:

- **show portmode all**
- **show controllers OC192 0/0/1/10**
- **show controllers STS192c 0/0/1/10**

```
RP/0/RP0/CPU0:ios#show portmode all
Tue Apr 28 11:45:55.671 UTC
```

```
Portmode Information

Port Name Portmode Type Framing Mapping PT type
 Rate
None
Optics0_0_1_10 Sonet WIS framing type None mapping type NA
 None
```

In above show command, the Framing column confirms that the framing type is WIS (WAN Interface Sublayer). When the controller is in WAN-PHY mode the WIS sublayer transports 10 Gigabit Ethernet frames in an OC-192c SONET payload.

When the online help function (?) on the router is used against the **show controllers oc192** or **show controllers STS192c** command, the list of WAN-PHY enabled card locations are displayed. This is a simple way to verify if WAN-PHY is enabled on the router.

For example:

```
RP/0/RP0/CPU0:ios # show controllers oc192 ?
0/0/1/10 OC192 Interface Instance
R/S/I/P Forward interface in Rack/Slot/Instance/Port format
```

```
RP/0/RP0/CPU0:ios#show controllers OC192 0/0/1/10
Port OC1920/0/1/10:
```

```
Status:
Primary State: Up
Configured Sec admin State: Normal
Inherited Sec admin State: Normal
Derived State: In Service
performace_monitoring enabled
```

Loopback: None

```
SECTION
 LOF = 0 LOS = 0 TIM-S = 0 BIP(B1) = 0
Overhead
J0 Transmit: (0)
J0 Receive: (0)
```

```

J0 Expected: (0)

LINE
 AIS = 0 RDI = 0 FEBE = 0 BIP(B2) = 0

Last clearing of "show controllers SONET" counters never

Detected Alarms: None
Masked Alarms: None
Detected Alerts: None
Masked Alerts: None

```

**Framing: SONET**

```

BER thresholds: SF = 1.0E-3 SD = 1.0E-6
TCA thresholds: B1 = 1.0E-6 B2 = 1.0E-6
 Clock source: internal (actual) line (configured)

```

Finally, the **show controllers STS192c 0/0/1/10** command is used to check SONET STS-192c format and data rate:

```

RP/0/RP0/CPU0:ios#show controllers STS192c 0/0/1/10

 Primary State: Up

 Sec Admin State: Normal

 Derived State: In Service

PATH
 FEBE = 0 BIP(B3) = 0
 NEWPTR = 0 PSE = 0 NSE = 0
Detected Alarms: None

Mask for Detected->Asserted: None

Detected Alerts: None
Mask for Detected->Reported: None
Payload Scrambling: Disabled
C2 State: Stable C2_rx = 0x0 (0) C2_tx = 0x0 (0) / Scrambling Derived
B3 = 10e-6
Overhead J1
Transmit : (0)
Received : (0)
Expected : (0)

 performace_monitoring enabled

```

The purpose of WAN-PHY is to render 10 Gigabit Ethernet compatible with the SONET STS-192c format and data rate, as defined by ANSI.

## Configuring SDH Mode on an Interface

This task describes how to configure SDH mode on the NC55-MPA-12T-S:

```

controller Optics 0/0/2/1
 port-mode sdh framing WIS
!
```

## Verification

To verify the SDH configuration, run these show commands:

- show portmode all
- show controllers STM64 0/0/2/1
- show controllers vc4-64c 0/0/2/1

```
RP/0/RP0/CPU0:ios#show portmode all
```

```
Portmode Information
```

```

Port Name Portmode Type Framing Mapping PT type
 Rate
Optics0_0_2_1 SDH WIS framing type None mapping type NA
None
```

The show command confirms that the Portmode type is SDH. When the controller is in SDH mode, WIS transports 10GE frames in an STM-64 payload.

The **show controllers STM64** command is used to check STM64 format and data rate:

```
RP/0/RP0/CPU0:ios#show controllers STM64 0/0/2/1
```

```
Mon Dec 7 11:13:31.697 UTC
```

```
Port STM640/0/2/1:
```

```
Status:
```

```
Primary State: Down
```

```
Configured Sec admin State: Normal
```

```
Inherited Sec admin State: Normal
```

```
Derived State: In Service
```

```
performace_monitoring enabled
```

```
Loopback: None
```

```
REGENERATOR SECTION
```

```
LOF = 1 LOS = 0 RS-TIM = 0 RS-BIP = 0
```

```
Overhead
```

```
J0 Transmit: (0)
```

```
J0 Receive: (0)
```

```
J0 Expected: (0)
```

```
MULTIPLEX SECTION
```

```
AIS = 0 RDI = 0 FEBE = 0 MS-BIP = 0
```

```
Last clearing of "show controllers SDH" counters never
```

```
Detected Alarms: LOF
```

```
Masked Alarms: None
```

```
Detected Alerts: None
```

```
Masked Alerts: None
```

```
Framing: SDH
```

```
BER thresholds: SF = 1.0E-3 SD = 1.0E-6
```

```
TCA thresholds: B1 = 1.0E-6 B2 = 1.0E-6
```

```
Clock source: internal (actual) line (configured)
```

```
RP/0/RP0/CPU0:ios#show controllers vc4-64c 0/0/2/1
Mon Dec 7 11:15:26.535 UTC

Primary State: Down

Sec Admin State: Normal

Derived State: In Service

PATH
 FEBE = 0 BIP(B3) = 0
 NEWPTR = 0 PSE = 0 NSE = 0
Detected Alarms: AU-LOP

Mask for Detected->Asserted: None

Detected Alerts: None
Mask for Detected->Reported: None
Payload Scrambling: Disabled
C2 State: Stable C2_rx = 0x0 (0) C2_tx = 0x0 (0) / Scrambling Derived
B3 = 10e-6
Overhead J1
Transmit : (0)
Received : (0)
Expected : (0)

performace_monitoring enabled
```



# TSoP Smart SFP for SDH and SONET Encapsulation

Table 42: Feature History Table

| Feature Name                                   | Release        | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|------------------------------------------------|----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| TSoP Smart SFP for SDH and SONET Encapsulation | Release 7.11.1 | <p>Introduced in this release on NCS 5500 fixed port routers</p> <p>This release introduces support for the Clear Channel Synchronous Transport Module Level-1 (STM1) Smart SFP (SFP-TS-OC3STM1-I) for the Transparent SONET or SDH over Packet (TSoP) protocol. This allows you to leverage your existing packet-switched network to transport traditional time-division multiplexing (TDM) traffic. TSoP Smart SFPs offer the following advantages:</p> <ul style="list-style-type: none"> <li>• Encapsulation of SDH or SONET bit streams into packet-switched network format</li> <li>• Improved suitability for pseudowire transport over an Ethernet network</li> </ul> |

The TSoP Smart SFP (SFP-TS-OC3STM1-I) is a special type of optical transceiver that allows for the transparent encapsulation of SDH or SONET bit streams into a packet format. This format is suitable for transporting pseudowires over an Ethernet network. The TSoP pseudowires can be manually configured or set up using the PWE3 control protocol [RFC4447].

TSoP provides packetization, de-packetization, and clock recovery. It translates the TDM bit stream into fixed-size data blocks (810 octets) and vice versa.

TSoP follows the SAToP method described in [RFC4553] for transporting E1/DS1 pseudowires over a packet-switched network. With TSoP, the entire OC-3 or STM-1 is encapsulated in a single circuit emulating pseudowire traffic. This traffic is then transported to a single destination across the Ethernet network.



- 
- Note** The TSoP Smart SFP is used on any of the front panel ports of the 8-port Gigabit Ethernet SFP Interface Module (8x1GE).
- The Smart SFP transceivers are compatible with the Small Form Factor Pluggable 20-pin Multi-Source Agreement (MSA).
  - TSoP Smart SFP (SFP-TS-OC3STM1-I) transports up to 155 Mbps on a L1.1 (40km) optical data link.
- 

## Prerequisites for TSoP

This section provides information about the prerequisites that apply to TSoP.

- Single mode optical fiber must be used to connect TSoP Smart SFP with the OC-3 port.
- The TSoP smart SFP pseudowire endpoints must use the same configuration parameters.

## Restrictions for TSoP

This section provides information about the restrictions that apply to TSoP.

- TSoP is not supported on the 25GE ports, but it's supported only on 1GE and 10GE ports.
- The RTP clock source value is Ethernet, by default. The **Clock source internal/line** under the **controller STM1 r/s/i/p**, is not applicable.
- Ensure that there is at least 2 minutes time-delay between the swaps, during a quick OIR of TSoP smart SFP with other SFPs on the same port.
- Before you insert TSoP, always ensure that the Ethernet interface is in a **shutdown** state. If TSoP doesn't come up when the Ethernet interface is accidentally present in **no shut**, then remove TSoP and insert Gigabit Ethernet SFP to move it to the shutdown state, and then reinsert TSoP as a workaround.
- TSoP only supports Differential Clock Recovery (DCR) by default and doesn't support any other clock configuration.
- Both sides of PE only support TSoP. Currently, interoperability with other devices is not supported.

## Guidelines for TSoP Smart SFP

The TSoP is compatible with the following SFPs supported on the OC-3 interface module. We recommend using the specified attenuator:

- ONS-SI-155-I1 - For a cable length of 15km, use a 2 dB attenuator. Use an 8 dB attenuator for short distances to avoid receiver overload.
- ONS-SI-155-L1 - For a cable length of 40km, no attenuator is needed. Use a 10 dB attenuator for short distances to avoid receiver overload.

- ONS-SI-155-L2 - For a cable length of 40km, use a 2 dB attenuator. Use a 10 dB attenuator for short distances to avoid receiver overload.



---

**Note** Multimode SFP is not supported with TSoP.

---

## De-jitter Buffer

A de-jitter buffer is a component in a packet-based network that helps mitigate the effects of jitter. Jitter refers to the variation in the arrival time of packets, which can lead to inconsistent delays and packet loss in real-time applications like voice or video.

The de-jitter buffer works by temporarily storing incoming packets and then releasing them at a regulated pace. It smooths out the variations in packet arrival times and ensures a more consistent and reliable stream of packets for playback.

The primary purpose of the de-jitter buffer is to ensure a steady and continuous playback of real-time data, such as voice packets in a voice-over-IP (VoIP) call. By absorbing and compensating for the variable delay, it helps maintain a stable voice quality without noticeable gaps or interruptions in the audio stream.

The size of the de-jitter buffer is a critical parameter to consider in network design. It should be large enough to handle the maximum expected delay variation (jitter) while still providing an acceptable level of delay. However, the buffer cannot be too large, as excessive buffering can introduce additional delay and affect the real-time nature of the application.

The de-jitter buffer for the TSoP can be configured to the following values:

- 292
- 627
- 810
- 1296
- 2633

### Example configuration

This example shows how to configure De-jitter buffer.

```
sh controll1 cem 0/0/0/10 payload-dejitter-mapping
Thu Jan 19 12:40:16.206 UTC
Client type : STM1
Client rate (in kbps) : 155000
Client default payload (in bytes) : 810
Client default dejitter (in usec) : 1296

Payload (in bytes) Possible dejitter values (in microseconds)

810 2633, 1296, 627, 292

For configuring out of range, we will get the below error:
Int CEM <>
```

```

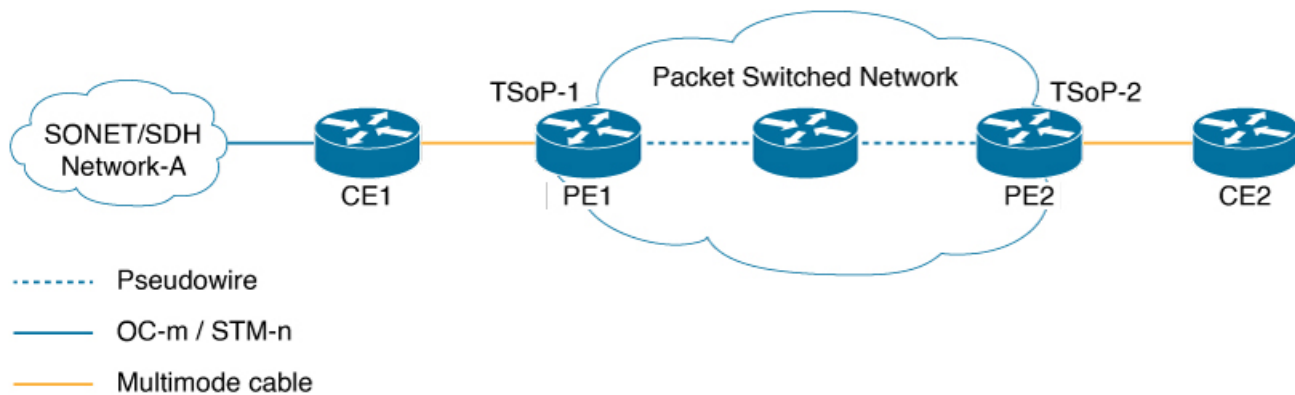
cem payload 810 dejitter 696
!

```

## Configuration for TSoP

Here is a sample configuration for TSoP:

**Figure 21: TSoP in Packet Switched Network**



### Pre-requisites

These are the prerequisites for configuring PE routers and CE (SONET or SDH) routers for TSoP:

- The following are the pre-requisites on CE routers ASR 903:
  - The CE side routers must be ASR 903 operating on RSP4 or RSP3
  - SFP-OC3-SR must be plugged into ASR 903 for transmitting SONET or SDH packets
  - Supported Interface modules are:
    - NCS4200-1T8S-20CS
    - A900-IMA3G-1MSG
- The PE side routers must be NCS 5500.
- PE and CE routers must be connected using multimode cable
- Only static L2VPN tunnel configuration is supported
- Sync-E should be operational



**Note** Traffic switch over is not supported even when multiple links are available. Traffic flow occurs as per the adjacent labels or hops defined by the user during static L2VPN tunnel configuration

## Configuration for

1. This example shows how to configure clock synchronization on the PE routers.

- Configuring Smart SFP in PE1

```
Router(config)# frequency synchronization quality itu-t option 1
Router(config)# interface GigabitEthernet0/0/0/5
Router(config)# no shut
Router(config)# frequency synchronization
Router(config)# quality transmit exact itu-t option 1 PRC
```

- Configuring Smart SFP in PE2

```
Router(config)# frequency synchronization quality itu-t option 1
Router(config)# interface GigabitEthernet0/0/0/18
Router(config)# frequency synchronization
Router(config)# selection input
Router(config)# priority 1
Router(config)# wait-to-restore 0
```

2. This example shows how to configure PE and CE interfaces.

- Configuration at CE1

```
Router(config)# interface CEM0/0/0/10
Router(config)# l2transport
```

- Configuration at PE1

```
Router(config)# interface GigabitEthernet0/0/0/5
Router(config)# ipv4 address 10.1.1.1 255.255.255.0
Router(config)# no shut
```

- Configuration at CE2

```
Router(config)# interface CEM0/0/0/17
Router(config)# l2transport
```

- Configuration at PE2

```
Router(config)# interface GigabitEthernet0/0/0/18
Router(config)# ipv4 address 10.1.1.2 255.255.255.0
Router(config)# no shut
```

3. This example shows how to configure loopback interface.

- Configuration at PE1

```
Router(config)# interface Loopback0
Router(config)# ipv4 address 1.1.1.1 255.255.255.255
```

- Configuration at PE2

```
Router(config)# interface Loopback0
Router(config)# ipv4 address 1.1.1.4 255.255.255.255
```

4. This example shows how to configure the ISIS IGP and advertise loopback and core interfaces.

- Configuration at PE1

```
Router(config)# router isis core
Router(config)# is-type level-2-only
Router(config)# net 49.0000.0000.0000.0001.00
Router(config)# nsr
```

```

Router(config)#nsf cisco
Router(config)#log adjacency changes
Router(config)#address-family ipv4 unicast
Router(config)#metric-style wide
Router(config)#segment-routing mpls sr-prefer
Router(config)#segment-routing bundle-member-adj-sid
Router(config)#address-family ipv6 unicast
Router(config)#metric-style wide
Router(config)#segment-routing mpls sr-prefer
Router(config)#segment-routing bundle-member-adj-sid
Router(config)#interface Loopback0
Router(config)#point-to-point
Router(config)#address-family ipv4 unicast
Router(config)#prefix-sid index 1
Router(config)#address-family ipv6 unicast
Router(config)#interface GigabitEthernet0/0/0/5
Router(config)#point-to-point
Router(config)#address-family ipv4 unicast
Router(config)#adjacency-sid absolute 28121
Router(config)#address-family ipv6 unicast

```

- Configuration at PE2

```

Router(config)#router isis core
Router(config)#is-type level-2-only
Router(config)#net 49.0000.0000.0000.0004.00
Router(config)#nsr
Router(config)#nsf cisco
Router(config)#log adjacency changes
Router(config)#address-family ipv4 unicast
Router(config)#metric-style wide
Router(config)#segment-routing mpls sr-prefer
Router(config)#segment-routing bundle-member-adj-sid
Router(config)#address-family ipv6 unicast
Router(config)#metric-style wide
Router(config)#segment-routing mpls sr-prefer
Router(config)#segment-routing bundle-member-adj-sid
Router(config)#interface Loopback0
Router(config)#point-to-point
Router(config)#address-family ipv4 unicast
Router(config)#prefix-sid index 4
Router(config)#address-family ipv6 unicast
Router(config)#interface GigabitEthernet0/0/0/18
Router(config)#point-to-point
Router(config)#address-family ipv4 unicast
Router(config)#adjacency-sid absolute 28211
Router(config)#address-family ipv6 unicast

```

5. This example shows how to configure Circuit-styled Segment routing traffic engineering tunnels.

- Configuration at PE1

```

Router(config)#segment-routing
Router(config)#global-block 80000 111999
Router(config)#local-block 25000 28999
Router(config)#traffic-eng
Router(config)#segment-list pe1-pe2-forward-path
Router(config)#index 1 mpls label 28121
Router(config)#segment-list pe1-pe2-reverse-path
Router(config)#index 1 mpls label 28211
Router(config)#policy pe1-pe2-circuit-styled-srte
Router(config)#color 10 end-point ipv4 1.1.1.4

```

```

Router(config)#path-protection
Router(config)#candidate-paths
Router(config)#preference 10
Router(config)#explicit segment-list pe1-pe2-forward-path
Router(config)#reverse-path segment-list pe1-pe2-reverse-path

```

- Configuration at PE2

```

Router(config)#segment-routing
Router(config)#global-block 80000 111999
Router(config)#local-block 25000 28999
Router(config)#traffic-eng
Router(config)#segment-list pe1-pe2-forward-path
Router(config)#index 1 mpls label 28121
Router(config)#segment-list pe1-pe2-reverse-path
Router(config)#index 1 mpls label 28211
Router(config)#policy pe1-pe2-circuit-styled-srte
Router(config)#color 10 end-point ipv4 1.1.1.4
Router(config)#path-protection
Router(config)#candidate-paths
Router(config)#preference 10
Router(config)#explicit segment-list pe1-pe2-forward-path
Router(config)#reverse-path segment-list pe1-pe2-reverse-path

```

6. This example shows how to configure BGP EVPN neighbor session.

- Configuration at PE1

```

Router(config)#router bgp 100
Router(config)#bgp router-id 1.1.1.1
Router(config)#bgp graceful-restart
Router(config)#address-family ipv4 unicast
Router(config)#address-family l2vpn evpn
Router(config)#neighbor 1.1.1.4
Router(config)#remote-as 100
Router(config)#update-source Loopback0
Router(config)#graceful-restart
Router(config)#address-family l2vpn evpn

```

- Configuration at PE2

```

Router(config)#bgp router-id 1.1.1.4
Router(config)#bgp graceful-restart
Router(config)#address-family ipv4 unicast
Router(config)#address-family l2vpn evpn
Router(config)#neighbor 1.1.1.1
Router(config)#remote-as 100
Router(config)#update-source Loopback0
Router(config)#graceful-restart
Router(config)#address-family l2vpn evpn

```

7. This example shows how to configure EVPN xconnect.

- Configuration at PE1

```

Router(config)#l2vpn
Router(config)#pw-class pw-cs-srte
Router(config)#encapsulation mpls
Router(config)#preferred-path sr-te policy pe1-pe2-circuit-styled-srte
Router(config)#xconnect group evpn_vpws
Router(config)#p2p p1
Router(config)#interface CEM0/0/0/10

```

```
Router(config)#neighbor evpn evi 10 target 1 source 2
Router(config)#pw-class pw-cs-srte
```

- Configuration at PE2

```
Router(config)#l2vpn
Router(config)#pw-class pw-cs-srte
Router(config)#encapsulation mpls
Router(config)#preferred-path sr-te policy pe1-pe2-circuit-styled-srte
Router(config)#xconnect group evpn_vpws
Router(config)#p2p p1
Router(config)#interface CEM0/0/0/17
Router(config)#neighbor evpn evi 10 target 2 source 1
Router(config)#pw-class pw-cs-srte
```

## 8. This example shows how to configure Dejitteer.

- Configuration at PE1

```
Router(config)#int cem0/0/0/10
Router(config)#cem payload 810 dejitter 696
```

- Configuration at PE2

```
Router(config)#int cem0/0/0/17
Router(config)#cem payload 810 dejitter 696
```

## Verification

Use the **show inventory** command to display all TSoP Smart SFPs installed on the router.

```
Router#show inventory
NAME: "GigabitEthernet0/0/0/17", DESCR: "Cisco SFP TSOP STM1 Pluggable Optics Module"
PID: SFP-TS-OC3STM1-I , VID: V01, SN: OEA2536001J
```

Use the **show hw-module fpd** command to display TSoP Smart SFPs FPD version.

```
Router#show hw-module fpd
Auto-upgrade:Disabled
FPD Versions
=====
Location Card type HWver FPD device ATR Status Running Programd

0/RP0 NCS-55A2-MOD-SE-S 1.0 MB-MIFPGA CURRENT 0.21 0.21
0/RP0 NCS-55A2-MOD-SE-S 1.0 SSFP_OC3_STM1_6 CURRENT 12.01 12.01
0/RP0 NCS-55A2-MOD-SE-S 1.0 SSFP_STM1_TSOP_17 CURRENT 13.00 13.00 <<<<
0/RP0 NCS-55A2-MOD-SE-S 1.0 Bootloader CURRENT 1.18 1.18
0/RP0 NCS-55A2-MOD-SE-S 1.0 CPU-IOFPGA CURRENT 1.27 1.27
0/RP0 NCS-55A2-MOD-SE-S 1.0 MB-IOFPGA NEED UPGD 0.18 0.18
0/RP0 NCS-55A2-MOD-SE-S 1.0 SATA-INTEL_240G NEED UPGD 1120.00 1120.00
0/PM0 NC55-1200W-ACFW 1.0 LIT-PrimCU-ACFW CURRENT 2.09 2.09
0/PM1 NC55-1200W-ACFW LIT-PrimCU-ACFW NOT READY
RP0/RP0/CPU0:ios#
```

Use the **show ipv4 interface brief** command to display the interface status.

```
Router#show ipv4 interface brief
Interface IP-Address Status Protocol Vrf-Name
MgmtEth0/RP0/CPU0/0 unassigned Shutdown Down default
TenGigE0/0/0/0 unassigned Shutdown Down default
GigabitEthernet0/0/0/1 unassigned Shutdown Down default
GigabitEthernet0/0/0/2 unassigned Shutdown Down default
TenGigE0/0/0/3 unassigned Shutdown Down default
```



```
TenGigE0/0/0/4 unassigned Shutdown Down default
GigabitEthernet0/0/0/5 unassigned Shutdown Down default
GigabitEthernet0/0/0/6 unassigned Up Up default
GigabitEthernet0/0/0/7 unassigned Shutdown Down default
GigabitEthernet0/0/0/8 unassigned Shutdown Down default
GigabitEthernet0/0/0/9 unassigned Shutdown Down default
GigabitEthernet0/0/0/10 unassigned Shutdown Down default
TenGigE0/0/0/11 unassigned Shutdown Down default
TenGigE0/0/0/12 unassigned Shutdown Down default
TenGigE0/0/0/13 unassigned Shutdown Down default
TenGigE0/0/0/14 unassigned Shutdown Down default
TenGigE0/0/0/15 unassigned Shutdown Down default
GigabitEthernet0/0/0/16 unassigned Up Up default
CEM0/0/0/17 unassigned Up Up default <<<
GigabitEthernet0/0/0/18 unassigned Shutdown Down default
TenGigE0/0/0/19 unassigned Shutdown Down default
```

Use the **show controller cem** verifying TSoP smart SFP stats and Dejitteer configuration.

```
Router#show controller cem 0/0/0/17
Interface : CEM0/0/0/17
Admin state : Up
Oper state : Up
Port bandwidth : 155000 kbps
Dejitteer buffer (oper/in-use) : 1296/180064 usec <<<<<<
Payload size (oper) : 810 bytes
PDV (min/max/avg) : 51589/142643/97116 usec
Dummy mode : last-frame
Dummy pattern : 0x0
Idle pattern : 0x0
Signalling : No CAS
RTP : Not Enabled
Clock type : Differential
Detected Alarms : None

Statistics Info

Ingress packets : 0, Ingress packets drop : 0
Egress packets : 0, Egress packets drop : 0
Total error : 1145472000
Missing packets : 572736000, Malformed packets : 0
Jitter buffer underrun : 572736000, Jitter buffer overrun : 0
Misorder drops : 0
Reordered packets : 0, Frames fragmented : 0
Error seconds : 0, Severely error seconds : 0
Unavailable seconds : 0, Failure counts : 0

Generated L bits : 0, Received L bits : 0
Generated R bits : 0, Received R bits : 0

Endpoint Info

Passthrough : No

//Run the same command on CEM interface of PE1 router.

Router#show controll CEM0/0/0/10
Sat Mar 25 10:49:19.471 UTC
Interface : CEM0/0/0/6
Admin state : Up
Oper state : Up
Port bandwidth : 155000 kbps
Dejitteer buffer (oper/in-use) : 1296/180645 usec
Payload size (oper) : 810 bytes
PDV (min/max/avg) : 51589/142643/97116 usec
```

```

Dummy mode : last-frame
Dummy pattern : 0xab
Idle pattern : 0xff
Signalling : No CAS
RTP : Enabled
Clock type : Differential
Detected Alarms : None

Statistics Info

Ingress packets : 0, Ingress packets drop : 0
Egress packets : 0, Egress packets drop : 0
Total error : 385536000
 Missing packets : 192768000, Malformed packets : 0
 Jitter buffer underrun : 192768000, Jitter buffer overrun : 0
 Misorder drops : 0
Reordered packets : 0, Frames fragmented : 0
Error seconds : 0, Severely error seconds : 0
Unavailable seconds : 0, Failure counts : 0

Generated L bits : 0, Received L bits : 0
Generated R bits : 0, Received R bits : 0

Endpoint Info

Passthrough : No
RP/0/RP0/CPU0:Router#

```