



Interface and Hardware Component Configuration Guide for Cisco 8000 Series Routers, IOS XR Release 26.1.x

First Published: 2026-02-28

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)

© 2026 Cisco Systems, Inc. All rights reserved.



CONTENTS

PREFACE

Preface xix

Changes to This Document xix

Communications, Services, and Additional Information xix

CHAPTER 1

New and Changed Feature Information 1

Interface and Hardware Component Features Added or Modified in IOS XR Release 26.x.x 1

CHAPTER 2

YANG Data Models for Interfaces and Hardware Component Features 3

Using YANG Data Models 3

Guidelines to configure interfaces using YANG data models 3

CHAPTER 3

Preconfiguring Physical Interfaces 5

Prerequisites for Preconfiguring Physical Interfaces 5

Information About Preconfiguring Physical Interfaces 6

Physical Interface Preconfiguration Overview 6

Benefits of Interface Preconfiguration 6

Use of the Interface Preconfigure Command 6

Active and Standby RPs and Virtual Interface Configuration 7

How to Preconfigure Physical Interfaces 7

CHAPTER 4

Advanced Configuration and Modification of the Management Ethernet Interface 9

Prerequisites for Configuring Management Ethernet Interfaces 9

Information About Configuring Management Ethernet Interfaces 10

Default Interface Settings 10

How to Perform Advanced Management Ethernet Interface Configuration 10

Configure a Management Ethernet Interface 10

Verify Management Ethernet Interface Configuration	13
Configuration Examples for Management Ethernet Interfaces	13
Configuring a Management Ethernet Interface: Example	13

CHAPTER 5**Configuring Ethernet Interfaces 15**

Prerequisites for Configuring Ethernet Interfaces	16
Information About Configuring Ethernet	17
Cisco 8000 Modular Line Cards	17
Default Configuration Values for 100-Gigabit Ethernet	17
Network Interface Speed and Auto-negotiation	18
Configuring network interface speed	19
QoS with auto-negotiated speeds	26
Benefits of QoS with auto-negotiated speeds	27
Supported interfaces for QoS with auto-negotiation	27
Restrictions for QoS support with auto-negotiated speeds	28
Enable QoS with auto-negotiated speeds	28
Layer 2 VPN on Ethernet Interfaces	29
Gigabit Ethernet Protocol Standards Overview	30
IEEE 802.3 Physical Ethernet Infrastructure	30
IEEE 802.3ae 10-Gbps Ethernet	30
IEEE 802.3ba 100 Gbps Ethernet	30
MAC Address	30
Ethernet MTU	30
IP MTU	31
IP MTU Checks	32
IP MTU Configuration Guidelines	33
IP MTU Limitations and Feature Support	36
IP MTU Scale	37
Configure IP MTU	37
Flow Control on Ethernet Interfaces	39
802.1Q VLAN	39
Interfaces and subinterfaces on the router	40
Layer 2, Layer 3, and EFPs	43
Untagged L2 subinterface	46

Enhanced Performance Monitoring for Layer 2 Subinterfaces (EFPs)	49
Other Performance Management Enhancements	50
Frequency Synchronization and SyncE	50
LLDP	51
LLDP Frame Format	53
LLDP TLV Format	53
Specifying User-Defined LLDP TLV Values	53
LLDP Operation	55
Supported LLDP Functions	55
Unsupported LLDP Functions	56
Setting the carrier delay on physical interfaces	57
Guidelines and Restrictions for Setting the Carrier Delay on Physical Interfaces	59
Configure the Carrier-delay Timer	60
How to Configure Ethernet	60
Configuring LLDP	61
LLDP Default Configuration	61
Enabling LLDP Per Interface	61
Enabling LLDP Globally	62
Configuring Global LLDP Operational Characteristics	63
Disabling Transmission of Optional LLDP TLVs	65
Disabling LLDP Receive and Transmit Operation for an Interface	66
Verifying the LLDP Configuration	67
Verifying the LLDP Global Configuration	68
Verifying the LLDP Interface Configuration	68
Configuring LLDP Snoop	69
Configuration Examples for Ethernet	74
Configuring an Ethernet Interface: Example	74
Configuring LLDP: Examples	75
Configuring a Layer 2 VPN AC: Example	75
Configuring Physical Ethernet Interfaces	75
Viewing Interface Counters Report	79
Instant Display of Traffic Rates for all the Physical Interfaces	80
Display of traffic rates for bundle interfaces	81
Guidelines and restrictions for display of traffic rates for bundle interfaces	81

- View bundle interface statistics 81
- How to Configure Interfaces in Breakout Mode 82
 - Information About Breakout 82
 - Configure Breakout in a Port 83
 - Remove the Breakout Configuration 83
 - Verify a Breakout Configuration 83
- Ethernet Interface Route Statistics 84

CHAPTER 6

Configuring Ethernet OAM 89

- Information About Configuring Ethernet OAM 89
 - Ethernet Link OAM 90
 - Neighbor Discovery 92
 - EFD 92
 - MIB Retrieval 93
 - Miswiring Detection (Cisco-Proprietary) 93
 - SNMP Traps 93
 - Link Monitoring 93
 - Remote Loopback 93
 - Configuration Examples for Ethernet OAM 93
 - Configuring Ethernet OAM Features on an Individual Interface: Example 94
 - Configuring an Ethernet OAM Profile Globally: Example 94
 - Configuring Ethernet OAM Features to Override the Profile on an Individual Interface: Example 95
 - Recovering from error-disable: Example 95
 - Clearing Ethernet OAM Statistics on an Interface: Example 96
 - Enabling SNMP Server Traps on a Router: Example 96
- Ethernet CFM 96
 - Maintenance Domains 101
 - Services 104
 - Maintenance Points 104
 - MEP and CFM Processing Overview 104
 - CFM Protocol Messages 106
 - Continuity Check (IEEE 802.1ag and ITU-T Y.1731) 106
 - Loopback (IEEE 802.1ag and ITU-T Y.1731) 110
 - Linktrace (IEEE 802.1ag and ITU-T Y.1731) 112

Configurable Logging	114
CFM hardware offload	114
Unidirectional link detection protocol	116
UDLD operation	117
Types of fault detection	118
UDLD modes of operation	118
UDLD aging mechanism	118
UDLD state machines	119
Limitations	120
Configure UDLD	120
How to Configure Ethernet OAM	121
Configuring Ethernet OAM	121
Configuring an Ethernet OAM Profile	122
Attaching an Ethernet OAM Profile to an Interface	127
Configuring Ethernet OAM at an Interface and Overriding the Profile Configuration	127
Verifying the Ethernet OAM Configuration	129
Configuring Ethernet CFM	129
Configuring a CFM Maintenance Domain	130
Configuring services for a CFM maintenance domain	131
Enabling and Configuring Continuity Check for a CFM Service	133
Configuring Cross-Check on a MEP for a CFM Service	135
Configuring Other Options for a CFM Service	136
Configuring CFM MEPs	138
Configuring Y.1731 AIS	140
Verifying the CFM Configuration	143
CFM Over Bundles	143
Ethernet SLA Statistics Measurement in a Profile	144
Ethernet frame delay measurement for L2VPN services	148
Minimum delay bin	153
Configure minimum delay bin support	155
Link loss forwarding	156
Link state monitor and propagation by CFM	158
Restrictions for link loss forwarding for CFM	159
Configure link loss forwarding for CFM	159

Remote link state propagation	162
Configure Link Loss Forwarding for Layer 2 Transport	164

CHAPTER 7**Configure Integrated Routing and Bridging 165**

Prerequisites to configure IRB	167
Guidelines and restrictions to configure IRB	167
Information about configuring IRB	168
IRB introduction	168
Bridge-Group Virtual Interface	169
Bridge-Group Virtual Interface	169
Supported features on BVI	169
BVI MAC address	170
BVI interface and line protocol states	170
Packet flows using IRB	171
Packet flows when host A sends to host B on the bridge domain	171
Packet flows when host A sends to host C from the bridge domain to a routed interface	172
Packet flows when host C sends to host B from a routed interface to the bridge domain	172
Supported environments for IRB	172
Additional IPv4-specific environments supported for IRB	173
Additional IPv6-specific environments supported for IRB	173
How to configure IRB	173
Configure BVI	174
Configuration guidelines	174
Configure Layer 2 AC interfaces	175
Configure a bridge group and assign interfaces to a bridge domain	177
Associate BVI as a routed Interface on a bridge domain	178
Display information about BVI	179
Configuration examples for IRB	179
Basic IRB configuration: example	179
IRB using ACs with VLANs: example	180
IPv4 addressing on a BVI supporting multiple IP networks: example	181
Comprehensive IRB configuration with BVI bundle interfaces and multicast configuration: example	181
IRB with BVI and VRRP configuration: example	182

6PE or 6VPE with BVI configuration: example 183

CHAPTER 8

IP Event Dampening 185

- IP Event Dampening Overview 187
- Interface State Change Events 187
 - Suppress Threshold 188
 - Half-Life Period 188
 - Reuse Threshold 188
 - Maximum Suppress Time 188
- Affected Components 188
 - Route Types 188
 - Supported Protocols 189
- How to Configure IP Event Dampening 189
 - Enabling IP Event Dampening 189
 - Verifying IP Event Dampening 190

CHAPTER 9

Configure Link Bundling 191

- Limitations and Compatible Characteristics of Ethernet Link Bundles 193
- Prerequisites for Configuring Link Bundling on a Router 195
- Information About Configuring Link Bundling 195
 - Link Bundling Overview 195
 - Link Aggregation Through LACP 196
 - IEEE 802.3ad Standard 196
 - Configuring LACP Fallback 197
 - LACP Short Period Time Intervals 198
 - Load Balancing 198
 - Layer 3 Egress Load Balancing on Link Bundles 199
 - Configuring the Default LACP Short Period Time Interval 199
 - Configuring Custom LACP Short Period Time Intervals 201
 - QoS and Link Bundling 202
 - Link Bundle Configuration Overview 202
 - Nonstop Forwarding During Card Failover 203
 - Link Failover 203
 - Link Switchover 203

- LACP Fallback 204
- Designate a Member Link as Unviable 204
- Guidelines and Restrictions for Designating Member Links as Unviable 206
- How to Configure Link Bundling 206
 - Configuring Ethernet Link Bundles 206
 - Configuring VLAN Bundles 210
 - 211
 - VLANs on an Ethernet Link Bundle 214
 - Configuring a Member Link as Unviable 214
- Configuration Examples for Link Bundling 215
 - Example: Configuring an Ethernet Link Bundle 215
 - Example: Configuring a VLAN Link Bundle 217

CHAPTER 10

- Configuring Virtual Loopback and Null Interfaces 219**
 - Prerequisites for Configuring Virtual Interfaces 219
 - Information About Configuring Virtual Interfaces 219
 - Virtual Loopback Interface Overview 220
 - Null Interface Overview 220
 - Virtual Management Interface Overview 220
 - Active and Standby RPs and Virtual Interface Configuration 221
 - How to Configure Virtual Interfaces 221
 - Configuring Virtual Loopback Interfaces 221
 - Configuring Null Interfaces 222
 - Configuring Virtual IPv4 Interfaces 222
 - Configuration Examples for Virtual Interfaces 223
 - Configuring a Loopback Interface: Example 223
 - Configuring a Null Interface: Example 224
 - Configuring a Virtual IPv4 Interface: Example 224

CHAPTER 11

- Configure GRE Tunnels 225**
 - GRE tunnels 225
 - Supported Features on a GRE Tunnel 229
 - Limitations for Configuring GRE Tunnels 229
 - Configure GRE Tunnels 230

Unidirectional GRE Encapsulation (GREv4)	231
Unidirectional GRE Decapsulation (GREv4)	231
ECMP and LAG Hashing for NVGRE Flows	232

CHAPTER 12**Configuring 802.1Q VLAN Interfaces 235**

Prerequisites for Configuring 802.1Q VLAN Interfaces	235
Information About Configuring 802.1Q VLAN Interfaces	236
802.1Q VLAN Overview	236
Subinterfaces	237
Subinterface MTU	238
Native VLAN	238
Layer 2 VPN on VLANs	238
How to Configure 802.1Q VLAN Interfaces	239
Configuring 802.1Q VLAN Subinterfaces	239
Configuring an Attachment Circuit on a VLAN	241
Removing an 802.1Q VLAN Subinterface	242
Configuration Examples for VLAN Interfaces	243
VLAN Subinterfaces: Example	243
Layer 2 interface VLAN encapsulation using VLAN ranges and lists	245
Configure a VLAN range and a VLAN list on Layer 2 subinterfaces	248

CHAPTER 13**Configure IP-in-IP tunnels 251**

Restrictions for IP-in-IP tunnel configuration	254
Configuration example for IPv4 tunnel	254
Configuration example for IPv6 tunnel	255
Configure improved scale for IP-in-IP tunnels	257
Controlling the TTL Value of Inner Payload Header	258
Time-to-Live uniform mode	259
Configuration guidelines for TTL uniform mode	260
Use cases for TTL uniform mode on a router	260
IP-in-IP Decapsulation	263
Decapsulation using tunnel source direct	267
Guidelines and Limitations	268
Configure Decapsulation Using Tunnel Source Direct	269

Configure Tunnel Destination with an Object Group	270
Hashing for load balancing	273
Configure hash rotation value	274
Hashing functions for load balancing	275
How enhanced hashing functions for load balancing work	276
Benefits of enhanced hashing functions	277
Configuration guidelines for enhanced hashing functions	277
Restrictions for enhanced hashing functions	277
Configure ECMP and SPA seed values	277
Configure extended entropy profile	279
ECMP Hashing Support for Load Balancing	280
User-defined fields for ECMP hashing	280
Configure User-Defined Fields for ECMP Hashing	283
<hr/>	
CHAPTER 14	Configure Generic UDP Encapsulation 287
Understand UDP encapsulation	289
Generic UDP encapsulation	289
Usage guidelines for GUE	292
Restrictions	292
Configure GUE IPv4	293
Configure GUE IPv6	295
Outer IP Header-Driven Hash Computation for Incoming GUE Packets	297
Configure Outer IP Header-Driven Hash Computation for Incoming GUE Packets	298
Flexible Assignment of UDP Port Numbers for Decapsulation	299
Guidelines for Setting up Decapsulation Using Flexible Port Numbers	300
Restrictions	301
Configuring Port Numbers for Decapsulation	301
Verification	307
GUEv1 static tunnel configuration over IPv4 networks	307
Configuration guidelines for GUEv1 static tunnels	308
Restrictions for GUEv1 static tunnels	308
Configure IPv4 and IPv6 GUEv1 static tunnels	309
Configure global UDP ports for GUEv1 static tunnels	310

CHAPTER 15**Configuring 400G Digital Coherent Optics 313**

- Configuring Frequency 324
- Configuring Chromatic Dispersion 326
- Configuring Optical Transmit Power 327
- Configuring Muxponder Mode 330
- Configure 2x200G DACs with 2x200 Breakout 332
 - Configuring 2x200G DACs with 2x200 Breakout 333
- Configure 100G operating modes with 200G DAC 334
 - Configure 100G operational modes with 200G and 4x100 DAC 336
- Configuring Modulation 337
- Configuring DAC Rate 339
- Configuring FEC 341
- Configuring Loopback 342
- Disable Auto-Squelching 344
- Configuring Performance Monitoring 346
- Configuring PM Parameters 347
- Configuring Alarms Threshold 351
- Configuring FEC Alarm Threshold 353
 - Guidelines and Restrictions for Setting the FEC Alarm Thresholds 357
 - Configuration Examples to Set FEC Alarm Threshold 358
 - Configuring FDD Alarm Thresholds 358
 - Configuring FED Alarm Thresholds 359
- Enhanced SOP tolerance mode for ULH optics 359
 - Benefits of SOP tolerance mode for ULH optics 360
 - Configure enhanced SOP tolerance mode 361
- Media Link-down PreFEC Degrade Enablement 362
 - Configure Media Link-down PreFEC Degrade 363
- Application select code provisioning 365
 - Configuration guidelines and restrictions for AppSel code provisioning 366
 - Configuration guidelines for AppSel code provisioning 366
 - Restrictions for AppSel code provisioning 367
 - How AppSel code provisioning works 367
 - Configure an AppSel code on an optical module 368

Alarms Troubleshooting	371
CD Alarm	372
Clear the CD Alarm	372
DGD Alarm	372
Clear the DGD Alarm	372
FLEXO_LOF	373
Clear the FLEXO_LOF Alarm	373
FLEXO_LOM	373
Clear the FLEXO_LOM Alarm	373
HI-LASERBIAS Alarm	373
Clear the HI-LASERBIAS Alarm	374
HI-RXPOWER Alarm	374
Clear the HI-RXPOWER Alarm	374
HI-RXPOWER Warn	374
Clear the HI-RXPOWER Warn Alarm	375
HI-TEMP Alarm	375
Clear the HI-TEMP Alarm	375
HI-TEMP Warn	375
Clear the HI-TEMP Warn Alarm	375
HI-TXPOWER Alarm	376
Clear the HI-TXPOWER Alarm	376
HI-TXPOWER Warn	376
Clear the HI-TXPOWER Warn Alarm	376
IMPROPER-REM	377
Clear the IMPROPER-REM Alarm	377
LOF	377
Clear the LOF Alarm	377
LOL	377
Clear the LOL Alarm	378
LOM	378
Clear the LOM Alarm	378
LO-RXPOWER Alarm	378
Clear the LO-RXPOWER Alarm	379
LO-RXPOWER Warn	379

Clear the LO-RXPOWER Warn Alarm	379
LOS	379
Clear the LOS Alarm	379
LOS-P	380
Clear the LOS-P Alarm	380
LO-TXPOWER Alarm	380
Clear the LO-TXPOWER Alarm	380
LO-TXPOWER Warn	380
Clear the LO-TXPOWER Warn Alarm	381
OOR_CD	381
Clear the OOR_CD Alarm	381
OSNR Alarm	381
Clear the OSNR Alarm	381
UNC-WORD Alarm	382
Clear the UNC-WORD Alarm	382
WVL-OOL	382
Clear the WVL-OOL Alarm	383

CHAPTER 16
Configuring 800G Optics 385

Configure 2x400G and 8x100G breakout modes	385
Guidelines for using Cisco QSFP-DD800 optical modules	386
Supported port pairings on Cisco 8212-48FH-M Router	386
Configure 800G optics	387
Configure 8x100G breakout mode	388
Support for DP08QSDD-ZRB-19B optics on 88-LC1-36EH line card	389

CHAPTER 17
Configuring Controllers 391

How to Configure Controllers	392
Configuring Optics Controller	392
Disabling Optical Modules	392
Diagnostic Parameters for Optical Transceivers	394
View Coherent Optical Transceiver Module Parameters	401
Display of Alarms for Coherent Optical Transceivers	405
Restrictions	406

- Verification 406
- Pseudo Random Binary Sequence 408
 - PRBS operational modes 408
 - Limitations for PRBS on optics controller 409
 - Configure PRBS on optics controller 409
 - Configure PRBS on coherent DSP controller 410
- Loopback on Optical Transceivers 411
- Media Side Input Loopback Configuration 413
- Media Side Output Loopback 414
- Host Side Input Loopback Configuration 415
- Host Side Output Loopback Configuration 417

CHAPTER 18

- Configuring QDD Optical Line System 419**
 - Overview of QDD optical line system 420
 - Supported Wavelength or Frequency Configuration 422
 - Functional Description of QDD OLS 423
 - QDD OLS Configurations 424
 - Configuring the Operational Mode, Amplifier Gain, and Amplifier Output Power 424
 - Configuring the Low-Threshold Power 427
 - Configuring the Optical Safety Remote Interlock (OSRI) 429
 - Configuring Safety Control Mode 430
 - Configuring Force APR 432
 - Use Case for QDD OLS pluggable 434
 - 8-Channel Optical Line System 434
 - OLS Alarms Troubleshooting 435

CHAPTER 19

- Managing Router Hardware 439**
 - MPA Reload 439
 - RP Redundancy and Switchover 440
 - Establishing RP Redundancy 440
 - Determining the Active RP in a Redundant Pair 441
 - Role of the Standby RP 442
 - Summary of Redundancy Commands 442
 - Automatic Switchover 442

RP Redundancy During RP Reload	443
Manual Switchover	443
Communicating with a Standby RP	444
NPU Power Optimization	445
Limitations	446
Configuring NPU Power Mode	446
Dynamic Power Management	449
Disabling Dynamic Power Management	457
On-demand transfer of Redundant Power Modules to Power Reservation Pool	457
Power Redundancy Protection	462
Guidelines and Restrictions for Power Redundancy Protection	464
Configure Power Redundancy Protection	464
Ability to Set Maximum Power Limit for the Router	466
Configuring the Compatibility Mode for Various NPU Types	468
Storage Media Sanitization	473
Factory reset of routers to remove SSD data	473
Guidelines and restrictions for factory reset functionality	474
Perform factory reset on a router	475
Secure erase of router SSD data	478
Restriction for secure erase functionality	479
Perform secure erase on a router	480
Excluding Sensitive Information in Show Running Configurations Output	481
Fabric Link Management for Uncorrectable Errors	483
Monitor FEC Fabric Links	485
Fault recovery handling	487
How to Configure the Fault Recovery Attempts	488
Periodic syslog messages for shutdowns due to fault-recovery failures	490
Limitations and restrictions for periodic shutdown syslog messages	491
Machine check error notifications	491
Limitations and restrictions for MCE major errors	493
Viewing error details in the cisco feature navigator error messages tool	493
Viewing error details in the MCE log file	494
Guidelines for Online Insertion and Removal on Cisco 8700 Series routers	494



Preface

This guide describes the interface and hardware component configuration details for Cisco IOS XR software. This chapter contains details on the changes made to this document.

- [Changes to This Document, on page xix](#)
- [Communications, Services, and Additional Information, on page xix](#)

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
February 2026	Initial release of this document

Communications, Services, and Additional Information

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

Cisco Bug Search Tool

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



CHAPTER 1

New and Changed Feature Information

This table summarizes the new and changed feature information for the *Interfaces Configuration Guide for Cisco 8000 Series Routers* for Cisco 8000 Series Routers, and tells you where they are documented.

- [Interface and Hardware Component Features Added or Modified in IOS XR Release 26.x.x](#), on page 1

Interface and Hardware Component Features Added or Modified in IOS XR Release 26.x.x

This table summarizes the new and changed feature information for the *Interfaces Configuration Guide for Cisco 8000 Series Routers* for Cisco 8000 Series Routers, and tells you where they are documented.

Table 2: New and Changed Features

Feature	Description	Introduced in Release	Where Documented
None	None	Not applicable	Not applicable



CHAPTER 2

YANG Data Models for Interfaces and Hardware Component Features

This chapter provides information about the YANG data models for Interface and Hardware Component features.

- [Using YANG Data Models, on page 3](#)
- [Guidelines to configure interfaces using YANG data models, on page 3](#)

Using YANG Data Models

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

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

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

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

Guidelines to configure interfaces using YANG data models

To configure and manage interfaces, it is very important to understand the specific restrictions and guidelines.

Guidelines

- Use index 0 exclusively for representing IP data on the main interfaces, and reserve index 1 onwards for subinterfaces.



CHAPTER 3

Preconfiguring Physical Interfaces

This module describes the preconfiguration of physical interfaces.

The system supports preconfiguration for the following interfaces:

- 10-Gigabit Ethernet
- 40-Gigabit Ethernet
- 100-Gigabit Ethernet
- 400-Gigabit Ethernet
- Management Ethernet

Preconfiguration allows you to configure line cards before you insert them into the router. When you insert the cards, they are instantly configured. The system creates the preconfiguration information 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 might be some preconfiguration data that you cannot verify unless the line card is present. This is because the verifiers themselves run only on the line card. You can verify such preconfiguration data when you insert the line card and initiate the verifiers. The system rejects a configuration if errors are found when you copy the configuration from the preconfiguration area to the active area.



Note Gigabit Ethernet interface is not supported. You can only preconfigure physical interfaces.

- [Prerequisites for Preconfiguring Physical Interfaces, on page 5](#)
- [Information About Preconfiguring Physical Interfaces, on page 6](#)
- [How to Preconfigure Physical Interfaces, on page 7](#)

Prerequisites for Preconfiguring Physical Interfaces

Before preconfiguring physical interfaces, ensure that you meet the following condition(s):

- Preconfiguration drivers and files are installed. Although it might 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.

Information About Preconfiguring Physical Interfaces

To preconfigure interfaces, you must understand the following concepts:

Physical Interface Preconfiguration Overview

Preconfiguration is the process of configuring interfaces before they are present in the system. You cannot verify or apply preconfigured interfaces until you insert the actual interface into the router with the matching location. The location can be the rack, slot, or module. When you insert the anticipated line card and create the interface, the system verifies the precreated configuration information. If the verification is successful, the system immediately applies the running configuration of the router.



Note When you plug the anticipated line card in, ensure that you verify any preconfiguration by using the appropriate **show** commands.

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



Note We recommend filling out preconfiguration information in your site planning guide. This allows you to compare the anticipated configuration with the actual preconfigured interfaces when you install the card 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 the valid configuration (best effort). Some configuration might fail due to semantic errors, but the valid configuration still comes up.

Benefits of Interface Preconfiguration

Preconfigurations reduce downtime when you add new cards to the system. With preconfiguration, you can instantly configure the new modular services card that actively runs during the line card bootup.

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

Use of the Interface Preconfigure Command

To preconfigure the interfaces that are not yet present in the system, use the **interface preconfigure** command in global configuration mode.

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

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



Note It is possible that you are not able to verify some configurations until you insert 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.

You must provide names during preconfiguration that matches with the name of the interface that is created. If the interface names do not match, the system does not apply preconfiguration 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.

Active and Standby RPs and Virtual Interface Configuration

The standby RP is available and is in a state in which it can take the load from the an active RP, if required. Following are the conditions when a standby RP becomes an active RP:

- Failure detection by a watchdog.
- Standby RP is administratively commanded to take over.
- Removal of the active RP from the chassis.

If a second RP is not present in the chassis while the first is in operation, the system may insert a second RP. The second RP then automatically becomes the standby RP. The standby RP may also be removed from the chassis with no effect on the system other than loss of RP redundancy.

After failover, the virtual interfaces become available on the standby (now active) RP. Their state and configuration is unchanged, and there is no loss of forwarding (in the case of tunnels) over the interfaces during the failover. The routers use nonstop forwarding (NSF) over tunnels through the failover of the host RP.



Note You do not need to configure anything to guarantee that the standby interface configurations are maintained.

How to Preconfigure Physical Interfaces

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

```
/* Enter global configuration mode. */  
RP/0/RP0/CPU0:router:router:hostname# configure
```

```
/* 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. */

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

/* Assign an IP address and mask to the interface. Use one of the following commands:
- ipv4 address ip-address subnet-mask
- ipv4 address ip-address/prefix */
RP/0/RP0/CPU0:router(config-if-pre)# ipv4 address 192.168.1.2/31
RP/0/RP0/CPU0:router(config-if-pre)# end
or
RP/0/RP0/CPU0:router(config-if-pre)# commit
RP/0/RP0/CPU0:router# show running-config
```

- 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.



CHAPTER 4

Advanced Configuration and Modification of the Management Ethernet Interface

This module describes the configuration of Management Ethernet interfaces.

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



Note By default, the Management Ethernet interfaces are present on the system. However, you must configure these interfaces to:

- Access the router.
- Use 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 9](#)
 - [Information About Configuring Management Ethernet Interfaces, on page 10](#)
 - [How to Perform Advanced Management Ethernet Interface Configuration, on page 10](#)
 - [Configuration Examples for Management Ethernet Interfaces, on page 13](#)

Prerequisites for Configuring Management Ethernet Interfaces

Before you perform the Management Ethernet interface configuration procedures that are described in this chapter, ensure that you meet the following tasks and conditions:

- 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, ensure that both the active and standby Management Ethernet interfaces are physically connected to the same LAN or switch.

Information About Configuring Management Ethernet Interfaces

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

Default Interface Settings

This table describes the default Management Ethernet interface settings that you can change with manual configuration. The system does not display the default settings in the **show running-config** command output.

Table 3: Management Ethernet Interface Default Settings

Parameter	Default Value	Configuration File Entry
Speed in Mbps	Default speed is 1G with autonegotiated.	Speed is non-configurable.
Duplex mode	Default duplex mode is full-duplex with autonegotiated.	Duplex mode is non-configurable.
MAC address	MAC address is read from the hardware burned-in address (BIA).	MAC address is non-configurable.

How to Perform Advanced Management Ethernet Interface Configuration

This section contains the following procedures:

Configure a Management Ethernet Interface

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



Note The maximum MTU value for the management interface MgmtEth0/RP0/CPU0/0 is 9678 bytes.

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

/* Enter interface configuration mode and specify the Ethernet interface name and notation
  rack/slot/module/port. */
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 (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. You can specify the network mask in either of the 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 system indicates the network mask 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 the network address.

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



Note (Optional) The maximum transmission unit (MTU) value for the management interface is 9678 bytes.

- The default is 1514 bytes.
- The range for the Management Ethernet interface Interface **mtu** values is from 64 through 9678 bytes.

```
/* Remove the shutdown configuration, which removes the forced administrative down on the
interface, enabling it to move to an up or down state. */
RP/0/RP0/CPU0:router(config-if)# no shutdown
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.

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

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)# 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 the 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
!
```

Verify Management Ethernet Interface Configuration

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

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

Configuration Examples for Management Ethernet Interfaces

This section provides the following configuration examples:

Configuring a Management Ethernet Interface: Example

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

```
RP/0//CPU0:router# configure
RP/0//CPU0:router(config)# interface MgmtEth 0/RP0/CPU0/0
RP/0//CPU0:router(config)# ipv4 address 172.29.52.70 255.255.255.0
RP/0//CPU0:router(config-if)# no shutdown
RP/0//CPU0:router(config-if)# commit
RP/0//CPU0:Mar 26 01:09:28.685 :ifmgr[190]:%LINK-3-UPDOWN :Interface MgmtEth 0/RP0/CPU0/0,
  changed state to Up
RP/0//CPU0:router(config-if)# end

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

MMgmtEth0//CPU0/0 is up, line protocol is up
  Hardware is Management Ethernet, address is 0011.93ef.e8ea (bia 0011.93ef.e8ea
)
  Description: Connected to Lab LAN
  Internet address is 172.29.52.70/24
  MTU 1514 bytes, BW 100000 Kbit
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set,
  ARP type ARPA, ARP timeout 04:00:00
  Last clearing of "show interface" counters never
  5 minute input rate 3000 bits/sec, 7 packets/sec
  5 minute output rate 0 bits/sec, 1 packets/sec
    30445 packets input, 1839328 bytes, 64 total input drops
    0 drops for unrecognized upper-level protocol
    Received 23564 broadcast packets, 0 multicast packets
      0 runts, 0 giants, 0 throttles, 0 parity
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
  171672 packets output, 8029024 bytes, 0 total output drops
  Output 16 broadcast packets, 0 multicast packets
  0 output errors, 0 underruns, 0 applique, 0 resets
  0 output buffer failures, 0 output buffers swapped out
  1 carrier transitions

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

interface MgmtEth0/RP0/CPU0/0
  description Connected to Lab LAN
```

```
ipv4 address 172.29.52.70 255.255.255.0  
!
```



CHAPTER 5

Configuring Ethernet Interfaces

This module describes the configuration of Ethernet interfaces.

The distributed 10-Gigabit, 25-Gigabit Ethernet, 40-Gigabit, 100-Gigabit Ethernet, architecture and features deliver network scalability and performance, while enabling service providers to offer high-density, high-bandwidth networking solutions designed to interconnect the router with other systems in POPs, including core and edge routers, Layer 2 switches and Layer 3 switches.

Table 4: Feature History Table

Feature Name	Release Information	Feature Description
Introduction of IP MTU	Release 25.1.1	Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*) *This feature is supported on Cisco 8011-4G24Y4H-I routers.
Introduction of IP MTU	Release 24.4.1	Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100, K100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*) *The IP MTU functionality is now supported on: <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 8712-MOD-M • 88-LC1-12TH24FH-E • 88-LC1-36EH • 88-LC1-52Y8H-EM

Feature Name	Release Information	Feature Description
Introduction of IP MTU on Q200-based Systems	Release 7.5.2	<p>You can configure IP MTU for IPv4 and IPv6 on a Layer 3 interface. Depending on your specific network requirements, this ability to specify IP MTU settings helps optimize router data transmission.</p> <p>Use the <code>show ipv4/ipv6 interfaces</code> command to view the IP MTU configurations.</p>



Tip You can programmatically configure and manage the Ethernet interfaces using `openconfig-if-ethernet.yang` and `openconfig-interfaces.yang` OpenConfig data models. To get started with using data models, see the *Programmability Configuration Guide for Cisco 8000 Series Routers*.

- [Prerequisites for Configuring Ethernet Interfaces, on page 16](#)
- [Information About Configuring Ethernet, on page 17](#)
- [Setting the carrier delay on physical interfaces, on page 57](#)
- [How to Configure Ethernet, on page 60](#)
- [Viewing Interface Counters Report, on page 79](#)
- [How to Configure Interfaces in Breakout Mode, on page 82](#)
- [Ethernet Interface Route Statistics, on page 84](#)

Prerequisites for Configuring Ethernet Interfaces

Before configuring Ethernet interfaces, ensure that you meet the following conditions:

- Access to Cisco 8200 series routers or Cisco 8800 series routers with at least one of the supported line cards installed.
- Know the interface IP address.
- Starting with Cisco IOS XR Release 24.1.1, the `openconfig-if-ip.yang` open config model supports a new leaf, `oc-if:interfaces/oc-if:interface/oc-if:subinterfaces/oc-if:subinterface/ipv4/addresses/address[ip]/config/type`. When using the `openconfig-if-ip.yang` model to update or revise the IP address for an interface, you must specify the IP address as either primary or secondary. You can have only one IP address as primary, but you can have multiple addresses configured as secondary.
- Ensure to specify the generalized interface name with the standard notation of `rack/slot/module/port`.



Note An ACL-dependent feature refers to a capability in network systems that relies on Access Control Lists (ACLs) for its operation. These features include both global such as Lawful Intercept (LI), BGP Flow Specification (BGPFS) and interface-level configurations, such as Quality of Service with ACL (QoS-ACL), Security ACL, SPAN ACL, QoS Policy Propagation via BGP (QPPB), Policy Based Routing (PBR), Peering QoS, and L2 ACL for packets with L3 payload.

An interface, whether physical or virtual, supports the configuration of up to four ACL dependent features, such as ACLs, QoS with ACL, BGP Flow Specification, SPAN ACL, and Lawful Intercept. To add a new feature, such as Policy-Based Routing, you must first remove one of the existing features and then configure the new feature.

Information About Configuring Ethernet

This section provides the following information:

Cisco 8000 Modular Line Cards

The current release of the Cisco 8800 Series Routers support the following line cards:

- 36-port QSFP56-DD 400 GbE Line Card - This line card provides 14.4 Tbps via 36 QSFP56-DD ports. It also supports 100G, 2x100G, and 400G modules. If 36 of 2x100G modules are used, the line card can have 72 HundredGigE interfaces.
- 48-port QSFP28 100 GbE Line Card - This line card provides 4.8 Tbps with MACsec support on all ports. It also supports QSFP+ optics for 40G compatibility.

The 8800 Series line cards utilize multiple #ChipName forwarding ASICs to achieve high performance and bandwidth with line rate forwarding.

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 36-port Line Card or a 48-port 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 5: 100-Gigabit Ethernet Line Card Default Configuration Values

Parameter	Configuration File Entry	Default Value
Flow control	flow-control	egress off ingress off

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 and Auto-negotiation

Auto-negotiation is an IEEE 802.3 Ethernet standard that

- enables a router to connect with its peer by automatically determining and negotiating to the highest common supported transmission parameters between them
- exchanges parameters, such as mode, speed, and flow control, between the two connected routers, and
- supports auto-negotiation on 1 G copper ports and 1 G SFP copper transceivers that allow an SFP port to connect to a copper Ethernet network at 1 Gbps using a standard RJ45 connector.

Table 6: Feature History Table

Feature Name	Release Information	Feature Description
Network interface speed support with autonegotiation	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100]) (select variants only*)</p> <p>*The auto-negotiation feature is now supported on 8011-4G24Y4H-I routers.</p>

Feature Name	Release Information	Feature Description
Network interface speed support with autonegotiation	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100]) (select variants only*)</p> <p>You can ensure compatibility between new and legacy routers by enabling auto-negotiation on the Gigabit interfaces of the connected routers. This process allows the routers to exchange key parameters such as mode, speed, and flow control, enabling them to agree on common transmission settings. As a result, the link achieves optimal bandwidth utilization.</p> <p>*This feature is supported on Cisco 8712-MOD-M routers.</p>

Auto-negotiation parameters

These parameters are crucial for Ethernet transmission because they directly define how data is sent and received over the link, impacting performance, efficiency, and reliability.

- Speed—1G copper ports and 1G SFP copper transceivers support auto-negotiation, allowing them to establish the highest common speed and duplex mode with connected routers
- Duplex—Full
- Pause—Receive part (RX) and Transmit part (TX)

Speed configuration and autonegotiation

- Speed can be manually set or configured to auto-negotiate with the remote interface.
- In auto-negotiation mode, the interface automatically determines whether to operate at 10 Mbps, 100 Mbps or 1 Gbps based on the capabilities of the connected device.
- Other settings such as full duplex and pause are also negotiated during this process.
- Auto-negotiation is an optional function of the Fast Ethernet standard that enables devices to automatically exchange information about speed and duplex abilities.
- It is especially useful for ports where devices with different capabilities are connected and disconnected regularly.
- Auto-negotiation is enabled, by default, on integrated copper ports and copper transceivers on most devices.

Configuring network interface speed

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

- Configure with **speed** option
- Configure with **negotiation auto** option
- Configure with **speed** and **negotiation auto** options



Note Cisco recommends configuring network interface speed in autonegotiation mode.

Configure with speed option

When you configure a 1 Gbps network interface using the **speed** command, the interface is forced to operate at the specified value.

Procedure

Step 1 Run the **show interfaces GigabitEthernet**< *interface-name*> command to check the existing speed.

Example:

```
#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, 1000 Mb/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
```

Observe that the speed is 1000 Mbps.

Step 2 Set the configuration that forces 1 Gbps interface speed to 100 Mbps.

Example:

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

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

If the remote end is not also configured for 100 Mbps, the link will go down.

Step 3 Run the **show controller GigE** and **show interface GigE** commands to verify if the speed is configured to 100 Mbps and autonegotiation is disabled.

Example

```
#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          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 disabled.

Operational values:
  Speed: 100 Mbps          /*Gig interface speed is set to 100 Mbps */
  Duplex: Full Duplex
  Flowcontrol: None
```

Configure with negotiation auto option

```

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, 100 Mb/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
carrier transitions

```

The show output confirms that GigabitEthernet0/0/0/31 is up and its line protocol is up. This occurs when both ends of the link are operating at 100 Mbps. If the remote end is not also configured for 100 Mbps, the link will go down.

Configure with negotiation auto option

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 10 Mbps, 100 Mbps, or 1 Gbps to be negotiated with the peer.

Procedure

Step 1 Set the configuration for interface speed to auto-negotiate. The router sets its interface speed to match the highest common speed supported by the peer router.

Example:

```

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

```

From Cisco IOS XR Software Release 7.11.x onwards, autonegotiation is not enabled by default. Use the **negotiation auto** command to enable autonegotiation.

Step 2 Run 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

#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
  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
```

Configure with the speed and negotiation auto options

```

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

```

Configure with the speed and negotiation auto options

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

Procedure

Step 1 Set the configuration for Gigabit interface speed to 100 Mbps and autonegotiate other parameters. The interface speed at remote end is set to 100 Mbps.

Example:

```

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

```

The interface does not attempt to negotiate the speed with its peer. It will simply try to establish a link at 100 Mbps. The auto-negotiation process might still occur for other parameters like duplex and flow control, but the speed itself is fixed by the **speed** command.

Step 2 Use the **show controller GigabitEthernet** and **show interface GigabitEthernet** command to verify if the link is up, speed is forced to 100 Mbps and auto-negotiation is enabled:

Example:

```

#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, 100 Mb/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 100 Mbps.

#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: 100 Mbps /* autonegotiation is enabled and speed is forced to 100 Mbps*/

Operational values:

```

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

```

QoS with auto-negotiated speeds

QoS with auto-negotiated speeds is a congestion management feature that

- manages congestion on interfaces that automatically adjust their speed—such as 10 Mbps or 100 Mbps—to connect with legacy devices
- addresses reduced bandwidth and increased congestion potential when operating at lower speeds, and
- prioritizes critical applications, controls bandwidth allocation, and minimizes packet loss or delay, even when connecting to legacy devices or working in mixed-speed environments.

Table 7: Feature History Table

Feature Name	Release Information	Feature Description
QoS with auto-negotiated speeds	Release 25.3.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100]) (select variants only*)</p> <p>You can now maintain optimal network efficiency by applying QoS on Gigabit Ethernet interfaces that operate at lower speeds through auto-negotiation. Auto-negotiation detects and matches the highest common speed between connected routers, allowing seamless interoperability. With this enhancement, you can use QoS on interfaces connecting to legacy devices that support only 10 Mbps or 100 Mbps speeds. This capability improves traffic management, leading to enhanced network performance and reliability across diverse deployment scenarios.</p> <p>*This feature is supported on 8011-4G24Y4H-1.</p>

Benefits of QoS with auto-negotiated speeds

- **Maximized bandwidth utilization:** When an interface auto-negotiates to a higher speed, the QoS policy dynamically scales up its allocations and limits. This ensures that the newly available bandwidth is fully leveraged by all traffic classes according to their defined priorities, preventing under-utilization of network resources.
- **Optimized resource allocation at lower speeds:** If an interface auto-negotiates down to a lower speed, QoS mechanisms, such as shapers and bandwidth allocations, automatically adjust downward, preventing enforcement of rates exceeding physical link capacity. This minimizes packet drops, control plane overhead, and inconsistent state conditions.

Supported interfaces for QoS with auto-negotiation

QoS with auto-negotiation is supported on

- Gigabit Ethernet interfaces that need to connect to legacy devices operating at 10 Mbps or 100 Mbps speeds.
- bundle interfaces that include Gigabit Ethernet interfaces operating at auto-negotiated speeds of 10 Mbps, 100 Mbps, or 1 Gbps.

Restrictions for QoS support with auto-negotiated speeds

- Absolute rate values such as fixed, specific numerical bandwidth or rate limits cannot be configured with on bundle interfaces.
- Hierarchical QoS (HQoS) policies are not supported on 10 or 100 M port on main and sub-interfaces of 8011-4G24Y4H-I, 8011-32Y8L2H2FH, and 8011-32Y8L2H2FH (ASIC: A100) systems.

Enable QoS with auto-negotiated speeds

The purpose of this task is to enable QoS with auto-negotiated speeds using relative values on Cisco IOS XR routers.

Before you begin

Confirm that the interface you intend to configure supports auto-negotiation and QoS.

Procedure

Step 1 Configure the interface to allow auto-negotiation.

Example:

```
Router(config)#interface GigabitEthernet0/0/0/28
Router(config)#negotiation auto
```

Step 2 Define the class-maps for traffic classification.

Example:

```
Router(config)#class-map match-any voice_traffic
Router(config-cmap)#match dscp ef
Router(config-cmap)#end-class-map
Router(config)#class-map match-any data_traffic
Router(config-cmap)#match dscp af21
Router(config-cmap)#end-class-map
```

Step 3 Define the policy map using bandwidth percentages.

Example:

```
Router(config)#policy-map adaptive_qos_policy
Router(config-pmap)#class voice_traffic
Router(config-pmap)#priority level 1
Router(config-pmap)#bandwidth percent 20 ! Allocates 20% of the *negotiated* link speed
Router(config-pmap)#class data_traffic
Router(config-pmap)#bandwidth percent 50 ! Allocates 50% of the *negotiated* link speed
Router(config-pmap)#class class-default
Router(config-pmap)#end-policy-map
```

We recommend using the percentile of port speed for interfaces with auto-negotiated speeds, as this allows the policy map to automatically adjust to any changes in speed.

Step 4 Apply the policy to the auto-negotiating interface.

Example:

```
Router(config)#interface GigabitEthernet0/0/0/28
Router(config)#service-policy output adaptive_qos_policy
```

Step 5 Execute the **show qos inconsistency** command to identify any conflict between a configured QoS policy and the interface's current operational state.

Example:

```
Router# show qos inconsistency detail location 0/rp0/cpu0
Fri Mar 28 14:53:36.542 UTC
Inconsistency Type: Policer conform rate greater than parent reference rate
=====
Interface                               Direction      Policy Name
-----
GigabitEthernet0/0/0/28                 input         p1-ing-negative

Inconsistency Type: Shape bandwidth rate greater than parent reference rate
=====
Interface                               Direction      Policy Name
-----
GigabitEthernet0/0/0/28                 output        p1-negative
```

Layer 2 VPN on Ethernet Interfaces

Layer 2 Virtual Private Network (L2VPN) connections emulate the behavior of a LAN across an L2 switched, IP or MPLS-enabled IP network, allowing Ethernet devices to communicate with each other as if they were connected to a common LAN segment.

The L2VPN feature enables service providers (SPs) to provide Layer 2 services to geographically disparate customer sites. Typically, an SP uses an access network to connect the customer to the core network. On the router, this access network is typically Ethernet.

Traffic from the customer travels over this link to the edge of the SP core network. The traffic then tunnels through an L2VPN over the SP core network to another edge router. The edge router sends the traffic down another attachment circuit (AC) to the customer's remote site.

On the router, an AC is an interface that is attached to an L2VPN component, such as a bridge domain.

The L2VPN feature enables users to implement different types of end-to-end services.

Switching takes place through local switching where traffic arriving on one AC is immediately sent out of another AC without passing through a pseudowire.

Keep the following in mind when configuring L2VPN on an Ethernet interface:

- L2VPN links support QoS (Quality of Service) and MTU (maximum transmission unit) configuration.
- If your network requires that packets are transported transparently, you may need to modify the packet's destination MAC (Media Access Control) address at the edge of the Service Provider (SP) network. This prevents the packet from being consumed by the devices in the SP network.

Use the **show interfaces** command to display AC information.

To attach Layer 2 service policies, such as QoS, to the Ethernet interface, refer to the appropriate Cisco IOS XR software configuration guide.

Gigabit Ethernet Protocol Standards Overview

The Gigabit Ethernet interfaces support the following protocol standards:

These standards are further described in the sections that follow.

IEEE 802.3 Physical Ethernet Infrastructure

The IEEE 802.3 protocol standards define the physical layer and MAC sublayer of the data link layer of wired Ethernet. IEEE 802.3 uses Carrier Sense Multiple Access with Collision Detection (CSMA/CD) access at various speeds over various physical media. The IEEE 802.3 standard covers 10 Mbps Ethernet. Extensions to the IEEE 802.3 standard specify implementations for 40-Gigabit Ethernet and 100-Gigabit Ethernet.

IEEE 802.3ae 10-Gbps Ethernet

Under the International Standards Organization's Open Systems Interconnection (OSI) model, Ethernet is fundamentally a Layer 2 protocol. 10-Gigabit Ethernet uses the IEEE 802.3 Ethernet MAC protocol, the IEEE 802.3 Ethernet frame format, and the minimum and maximum IEEE 802.3 frame size. 10-Gbps Ethernet conforms to the IEEE 802.3ae protocol standards.

Just as 1000BASE-X and 1000BASE-T (Gigabit Ethernet) remained true to the Ethernet model, 10-Gigabit Ethernet continues the natural evolution of Ethernet in speed and distance. Because it is a full-duplex only and fiber-only technology, it does not need the carrier-sensing multiple-access with the CSMA/CD protocol that defines slower, half-duplex Ethernet technologies. In every other respect, 10-Gigabit Ethernet remains true to the original Ethernet model.

IEEE 802.3ba 100 Gbps Ethernet

IEEE 802.3ba is supported on the Cisco 1-Port 100-Gigabit Ethernet PLIM beginning in Cisco IOS XR 7.0.11.

MAC Address

A MAC address is a unique 6-byte address that identifies the interface at Layer 2.

Ethernet MTU

The Ethernet maximum transmission unit (MTU) is the size of the largest frame, minus the 4-byte frame check sequence (FCS), that the system transmits 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 unfragmented packet that the system can send. 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, the system fragments that packet as necessary. This process ensures that the originating device does not send an IP packet that is too large.

The system automatically enables the jumbo frame support 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).

Limitations for Ethernet MTU

- Cisco IOSXR software does not support Ethernet MTU on BVI interfaces.
- Cisco IOSXR software supports **IPv4 mtu** and **IPv6 mtu** commands on BVI interfaces for Layer 3 to Layer 2 direction.
- Cisco IOSXR software supports packet fragmentation for traffic routed over BVI interfaces for Layer 3 to Layer 2 direction.

IP MTU

In IP protocol, Maximum Transmission Unit (MTU) refers to the maximum size of an IP packet that the system transmits without fragmentation over a given medium. The size of an IP packet includes IP headers but excludes headers from the data link layer, also known as the Ethernet headers. The default IP MTU on all router interfaces is 1500 bytes, when IP is enabled by using the IP address configuration commands. However, you can configure the IP MTU to different value as well.

Starting Cisco IOS XR Release 7.5.2, the system supports IP MTU (IPv4 and IPv6) on Q200-based systems on the following Cisco 8000 Series router and line card:

- 8201-32FH
- 88-LC0-36FH-M

How is Ethernet MTU different from IP MTU?

Ethernet MTU defines the maximum packet size that an interface supports, while IP MTU defines the MTU size of an IP packet.

How is IP MTU calculated?

The following scenarios provide information on how the system calculates IP MTU size, when:

- Ethernet MTU is not configured, the system sets:
 - Default value as 1514 bytes on a physical or bundle main interface
 - 1518 bytes for single tag VLAN interface
 - 1522 bytes for double tag VLAN (QinQ) interface



Note In this case, IP MTU value will be 1500 bytes for IPv4 and IPv6 MTUs.

- Ethernet MTU is configured as X bytes, the system sets:

- X bytes on a physical or bundle main interface
- X+4 bytes for single tag VLAN interface
- X+8 bytes for double tag VLAN (QinQ) interface



Note In this case, IP MTU is X-14 bytes for IPv4 and IPv6 MTUs.

The following are some of the important guidelines for IP MTU size:

- When no Ethernet MTU and IP MTU is configured, the default value is 1500B.
- When Ethernet MTU and no IP MTU is configured, IP MTU value in the hardware is Ethernet MTU-14B.
- IP MTU value can't be more than Ethernet MTU value. For example, if the Ethernet MTU value is 3000 bytes and you configure IP MTU value as 5000 bytes. The system sets the IP MTU value as 2986 (3000-14)

What is Maximum Receive Unit (MRU) and how is it different from MTU?

MRU is the largest packet size that an interface can receive. This is an ingress parameter. Usually, MRU equals MTU. However, you can't configure an MRU value. The Ethernet MTU, also known as a Layer2 (L2) value that you configure on a physical interface is also applied as the MRU of that physical interface.

The following table lists Ethernet MTU, IPv4, and IPv6 MTU support across various platforms and their limitations as applicable:

Table 8: IP MTU Support Across Platforms

ASIC	Ethernet MTU Check	IPv4 and IPv6 MTU
Q100-based	Ethernet MTU check on an egress interface is not supported.	Supported and the system derives IP MTU value from Ethernet MTU.
Q200-based	Ethernet MTU check on an egress interface is not supported.	Supported Note IPv4 and IPv6 can have their own separate MTU values.
P100-based	Ethernet MTU check on an egress interface is not supported.	Supported Note IPv4 and IPv6 can have their own separate MTU values.

IP MTU Checks

Cisco IOS XR supports MRU checks, ethernet MTU checks, and IP MTU checks. These checks decide if an IP payload packets needs fragmentation or not. Ethernet MTU and IP MTU check is applied on all egress traffic for each packet that flows in the system. In the forwarding plane, also known as dataplane, the IP packet

length is compared against IP MTU value applied on the L3 forwarding interface. Full packet length is compared against ethernet MTU value applied on the physical port.

The following table describes IP MTU check to each forwarding flow on different systems.

Table 9: IP MTU Check

ASIC	IP MTU Check
Q100-based	These interfaces implement IP MTU checks and provide fragmentation for all IP payload packets. A single value for both IPv4 and IPv6 is supported.
Q200-based	These interfaces implement IP MTU checks and provide fragmentation for all IP payload packets.
Mixed System with Q100 and Q200-based line cards	Interfaces with Q100-based system provides its behavior and interfaces with Q200-based systems provides its behavior as described in this table.
P100-based	These interfaces implement IP MTU checks and provide fragmentation for all IP payload packets.

IP MTU Configuration Guidelines

An Ethernet interface has a default of 1514 bytes. IP MTU is always derived from the default Ethernet MTU. If Ethernet MTU is configured, IP MTU is derived from the previously configured Ethernet MTU.

The following are the configuration guidelines for IP MTU across various platforms:

Guidelines for Q200-based Systems

- If you don't define any Ethernet MTU configuration, the system assigns the default value of Ethernet packet size of 1514 bytes.
- For IPv4 and IPv6 MTUs, 1500 bytes is used, which is derived by subtracting 14 bytes of Ethernet header size from its default value.
- On any interface, if an Ethernet MTU is configured, the new configuration takes higher precedence than the default interface configuration. However, in such a scenario, the new configuration doesn't get applied on the subinterfaces. The rest of the interfaces continue to work with the default Ethernet MTU configurations.
- For the double tag (QinQ) packets, MRU and Ethernet MTU value are derived by adding 8 bytes to the configured value.
- IPv4 and IPv6 MTU values are derived by subtracting 14 bytes of the Ethernet header size out of the configured value.
- If an IPv4 or IPv6 MTU is configured, it is applied to IP MTU value. MRU or MTU values remain unaffected by this configuration.
- Configuration restrictions apply to validate that the IP MTU value is smaller than Ethernet MTU value that is configured on an interface.



Note P100-based systems follow the same guidelines as Q200-based systems.

The following sample table explains how the system calculates the Ethernet MTU, IPv4, and IPv6 MTU values when configured on various interfaces that are Q200-based systems.

Table 10: Interfaces and Configurations

Interface Type	Default Configurations when, no Ethernet or IP MTU configuration is applied	Ethernet MTU Configurations for example, MTU = 1614	IPv4/IPv6 Configurations for example, IPv4 IPv6 MTU = 1550, Ethernet MTU = 1614
Any Main Interface	NA	Configuration that is applied on any main interface and its subinterfaces is effective. Configuration that is applied on any subinterface is not effective.	NA
Physical Port	MRU = 1514 + 8	MRU = 1614 + 8	MRU = 1614 + 8
L3 Main Interface (Physical or bundle)	Ethernet MTU = 1514 + 8 IPv4/IPv6 MTU = 1500	Ethernet MTU = 1614 + 8 IPv4/IPv6 MTU = 1614 - 14	Ethernet MTU = 1614 + 8 IPv4/IPv6 MTU = 1550
L3 Sub-interface (Physical or bundle)	Ethernet MTU = 1514 + 8 IPv4/IPv6 MTU = 1500	Note Configuration not applicable on subinterfaces. Takes main interface configuration. Ethernet MTU = Not applicable IPv4/IPv6 MTU = 1614 - 14	Ethernet MTU = 1614 + 8 IPv4/IPv6 MTU = 1550
SVI/BVI	IPv4/IPv6 MTU = 1500	Ethernet MTU = Not applicable IPv4/IPv6 MTU = 1600	IPv4/IPv6 MTU = 1550

Interface Type	Default Configurations <i>when, no Ethernet or IP MTU configuration is applied</i>	Ethernet MTU Configurations <i>for example, MTU = 1614</i>	IPv4/IPv6 Configurations <i>for example, IPv4 IPv6 MTU = 1550, Ethernet MTU = 1614</i>
Ethernet Main Interface	Ethernet MTU = 1514 + 8	Ethernet MTU = 1614 + 8 IPv4/IPv6 MTU = Not applicable	Not applicable Ethernet MTU = 1614 + 8
Ethernet Sub-interface	Ethernet MTU = 1514 + 8	Ethernet MTU = 1614 + 8 IPv4/IPv6 MTU = Not applicable	Not applicable Ethernet MTU = 1614 + 8

Guidelines for Q100-based Systems

- If you don't define any MTU configuration in the system, the system assigns default value of 1514 bytes to the Ethernet packets.
- An MRU of 1522 bytes is set to allow for any double tag packets on an interface.
- Ethernet MTU check on egress interface is not supported.
- For IPv4 and IPv6 MTUs, 1500 bytes is used, which is derived by subtracting 14 bytes of Ethernet header size from its default value.
- If you configure an IP MTU (IPv4/IPv6 MTU) on any interface, the configuration does not take effect.

The following table lists the MTU configurations and MTU calculations on various interfaces for Q100-based systems:

Table 11: Interfaces and Configurations

Interface Type	Default Configurations	Interface MTU Configurations <i>where, MTU = 1614</i>
Any Main Interface	NA	Configuration applied on any main interface and its subinterfaces is effective. Configuration applied on any sub-interface is not effective.
Physical Port	MRU = 1514 + 8	MRU = 1614 + 8
L3 Main Interface (Physical or bundle)	Ethernet MTU check on egress interface is not supported. IPv4/IPv6 MTU = 1522 applied on full packet size	Ethernet MTU check on egress interface is not supported. IPv4/IPv6 MTU = 1622 applied on full packet size

Interface Type	Default Configurations	Interface MTU Configurations <i>where, MTU = 1614</i>
L3 Subinterface (Physical or bundle)	Ethernet MTU check on egress interface is not supported. IPv4/IPv6 MTU = 1522 applied on full packet size	Note Configuration not applicable on sub-interfaces. Takes main interface configuration. Ethernet MTU = Not applicable IPv4/IPv6 MTU = 1622 applied on full packet size
SVI and BVI	Ethernet MTU is not applicable IPv4/IPv6 MTU = 1522 applied on full packet size	
Ethernet Main Interface	Ethernet MTU check on egress interface is not supported. IPv4/IPv6 MTU is not applicable	
Ethernet Subinterface	Ethernet MTU check on egress interface is not supported. IPv4/IPv6 MTU is not applicable	
GRE tunnel	Ethernet MTU check on egress interface is not supported. IPv4/IPv6 MTU = 1522 applied on full packet size	

IP MTU Limitations and Feature Support

The following are the limitations and feature support for IP MTU:

- MPLS MTU is not supported.
- GRE/IPinIP MTU is not supported.



Note GRE interface uses the configured L3 MTU. If an egress packet is above the configured L3 MTU value, the packet is discarded.

- Multicast MDT MTU is not supported.
- Ethernet MTU configuration on sub-interfaces is not supported.
- Ethernet MTU configuration on BVI interfaces is not supported.



Important Although CLIs for the unsupported features might be available on your router, they are not functional.

IP MTU Scale

The following are the scale support and limitations for IP MTU configurations:

- Ethernet MTU configuration is allowed on every physical or bundle interface without any scale limits.
- IPv4/IPv6 MTU configuration is allowed on every L3 forwarding interface. However, the system applies it as an IP MTU profile with a combination of <IPv4 MTU and IPv6 MTU> set. Each profile takes a unique set of these two MTUs, and supports eight such unique MTUs. The IP MTU profiles are stored identically across all NPUs of a given line card.
- When system runs out of resources (OOR), the system generates syslog messages while the continuing to use the previously configured IP MTU values.

Configure IP MTU

To configure IPv4, or IPv6 MTU, you must be in the configuration mode on L3 interface.

To configure IP MTU, use the following configuration steps:

```
Router#config t
Router#interface HundredGigE0/0/0/33
Router:abc(config)#
Router:abc(config-if)#ipv4 mtu 2000
Router:abc(config-subif)#commit
Router:abc(config-if)#end
```

Running Configuration

```
Router:abc#
Router:abc#sh ipv4 int HundredGigE0/0/0/33
Thu Apr 28 16:54:10.820 UTC
HundredGigE0/0/0/33 is Shutdown, ipv4 protocol is Down
  Vrf is default (vrfid 0x60000000)
  Internet address is 10.10.10.91/29
  MTU is 9642 (2000 is available to IP)
```

Verification

```
Router:abc#sh int HundredGigE0/0/0/33
Thu Apr 28 16:57:23.390 UTC
HundredGigE0/0/0/33 is administratively down, line protocol is administratively down
  Interface state transitions: 0
  Hardware is HundredGigE, address is e41f.7bde.123c (bia e41f.7bde.123c)
  Internet address is 194.19.242.94/29
  MTU 9642 bytes, BW 100000000 Kbit (Max: 100000000 Kbit)
    reliability 255/255, txload 0/255, rxload 0/255
  Encapsulation ARPA,
  Full-duplex, 100000Mb/s, 100GBASE-AOC, link type is force-up
  output flow control is off, input flow control is off
  Carrier delay (up) is 10 msec
  loopback not set,
  ARP type ARPA, ARP timeout 04:00:00
  Last input never, output never
  Last clearing of "show interface" counters never
  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 runts, 0 giants, 0 throttles, 0 parity
 0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
 0 packets output, 0 bytes, 0 total output drops
Output 0 broadcast packets, 0 multicast packets
 0 output errors, 0 underruns, 0 applique, 0 resets
 0 output buffer failures, 0 output buffers swapped out
 0 carrier transitions

```

```
Router:abc#
```

Configure IPv4 MTU

To configure IPv4 MTU, use the following configuration steps:

```

Router#config t
Router#interface HundredGigE0/0/0/33
Router:abc(config)#
Router:abc(config-if)#ipv4 mtu 2000
Router:abc(config-subif)#commit
Router:abc(config-if)#end

```

Running Configuration

```

Router:abc#sh run int HundredGigE0/0/0/33
Thu Apr 28 16:22:06.796 UTC
interface HundredGigE0/0/0/33
mtu 9642
ipv4 mtu 2000
ipv4 address 10.10.10.1 255.255.255.248
ipv6 address 10::1:10:9/112
load-interval 30
!

```

Verification

```

Router:abc#sh ipv4 int HundredGigE0/0/0/33
Thu Apr 28 16:54:10.820 UTC
HundredGigE0/0/0/33 is Shutdown, ipv4 protocol is Down
  Vrf is default (vrfid 0x60000000)
  Internet address is 10.10.10.1/29
  MTU is 9642 (2000 is available to IP)

```

Configure IPv6 MTU

To configure IPv6 MTU, use the following configuration steps:

```

Router#config t
Router#interface HundredGigE0/0/0/33
Router:abc(config)#
Router:abc(config-if)#ipv6 mtu 3000

```

```
Router:abc(config-subif)#commit
Router:abc(config-if)#end
```

Running Configuration

```
Router:abc#sh run int HundredGigE0/0/0/33
Thu Apr 28 16:23:09.141 UTC
interface HundredGigE0/0/0/33
mtu 9642
ipv4 mtu 2000
ipv4 address 10.10.10.1 255.255.255.248
ipv6 mtu 3000
ipv6 address 10::10:10:9/112
load-interval 30
!

Router:abc#
```

Verification

```
Router:abc#sh ipv6 int HundredGigE0/0/0/33
Thu Apr 28 16:54:41.222 UTC
HundredGigE0/0/0/33 is Shutdown, ipv6 protocol is Down, Vrfid is default (0x60000000)
  IPv6 is enabled, link-local address is fe80::e61f:7bff:fede:123c [TENTATIVE]
  Global unicast address(es):
    194::19:242:94, subnet is 10::10:10:0/112 [TENTATIVE]
  Joined group address(es): ff02::2 ff02::1
  MTU is 9642 (3000 is available to IPv6)
```

Flow Control on Ethernet Interfaces

The flow control that the system uses on 10-Gigabit Ethernet interfaces consists of periodically sending flow control pause frames. It is fundamentally different from the usual full and half-duplex flow control that is used on standard management interfaces. You can activate or deactivate flow control for ingress traffic only. The system automatically implements flow control for egress traffic.

802.1Q VLAN

A VLAN is a group of devices on one or more LANs that the system configures so that they can communicate as if they are attached to the same wire, when in fact they are located on several different LAN segments. Because VLANs are based on logical instead of physical connections, it is flexible for user and host management, bandwidth allocation, and resource optimization.

The IEEE's 802.1Q protocol standard addresses the problem of breaking large networks into smaller parts so broadcast and multicast traffic does not consume more bandwidth than necessary. The standard also helps to provide a higher level of security between segments of internal networks.

The 802.1Q specification establishes a standard method for inserting VLAN membership information into Ethernet frames.

Interfaces and subinterfaces on the router

Table 12: Feature History Table

Feature Name	Release Information	Feature Description
Interfaces and subinterfaces	Release 25.1.1	Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*) *This feature is supported on Cisco 8011-4G24Y4H-I routers.
Interfaces and subinterfaces	Release 24.4.1	Introduced in this release on: Fixed Systems (8700 [ASIC: K100])(select variants only*) This feature that allows for the configuration of main trunk interfaces that are either physical or bundle interfaces, automatic creation of physical interfaces, and creation of logical subinterfaces with unique IDs for efficient network segmentation and management is now supported on the following hardware. *This feature is now supported on the Cisco 8712-MOD-M routers.
Interfaces and subinterfaces	Release 24.3.1	Introduced in this release on: Fixed Systems (8200 [ASIC: Q200, P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: Q100, Q200, P100])(select variants only*) *The support for Interfaces and subinterfaces is now extended to: <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-52Y8H-EM • 88-LC1-36EH • 88-LC1-12TH24FH-E
Interfaces and subinterfaces	Release 24.2.11	Introduced in this release on: Modular Systems (8800 [LC ASIC: P100]) (select variants only*) The interfaces are initially configured as main trunk interfaces, which can be either physical or bundle types, with physical interfaces being automatically created by the system. The benefit of this setup is the flexibility it provides, allowing users to create logical subinterfaces under trunk interfaces with unique identifiers, enabling efficient network segmentation and management. *This feature is now supported on routers with the Cisco 88-LC1-36EH line cards.

In Cisco IOS XR, interfaces are, by default, main interfaces. A main interface is also known as a trunk interface, which you must not confuse with the word trunk in the context of VLAN trunking.

There are two types of trunk interfaces:

- Physical
- Bundle

On the router, the system automatically creates the physical interfaces when the router recognizes a card and its physical interfaces. However, the system does not automatically create bundle interfaces but you must create them at the time of configuration.

The following configuration samples are examples of the trunk interfaces that you can create:

- interface HundredGigE 0/5/0/0
- interface bundle-ether 1

A subinterface is a logical interface that the system create under a trunk interface.

To create a subinterface, you must first identify a trunk interface under which to place it. In case of bundle interfaces, if a trunk interface does not exist, you must create a bundle interface before creating any subinterfaces under it.

You can then assign a subinterface number to the subinterface that you want to create. The subinterface number must be a positive integer from zero to some high value. For a given trunk interface, each subinterface under it must have a unique value.

Subinterface numbers do not need to be contiguous or in numeric order. For example, the following subinterfaces numbers are valid under one trunk interface:

1001, 0, 97, 96, 100000

Subinterfaces can never have the same subinterface number under one trunk.

In the following example, the card in slot 5 has trunk interface, HundredGigE 0/5/0/0. A subinterface, HundredGigE 0/5/0/0.0, is created under it.

```
RP/0/RSP0/CPU0:router# conf
Mon Sep 21 11:12:11.722 EDT
RP/0/RSP0/CPU0:router(config)# interface HundredGigE0/5/0/0.0
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 100
RP/0/RSP0/CPU0:router(config-subif)# commit

RP/0/RSP0/CPU0:Sep 21 11:12:34.819 : config[65794]: %MGBL-CONFIG-6-DB_COMMIT : Configuration
  committed by user 'root'. Use 'show configuration commit changes 1000000152' to view the
  changes.

RP/0/RSP0/CPU0:router(config-subif)# end

RP/0/RSP0/CPU0:Sep 21 11:12:35.633 : config[65794]: %MGBL-SYS-5-CONFIG_I : Configured from
  console by root
RP/0/RSP0/CPU0:router#
```

The **show run** command displays the trunk interface first, then the subinterfaces in ascending numerical order.

```
RP/0/RSP0/CPU0:router# show run | begin HundredGigE 0/5/0/0
Mon Sep 21 11:15:42.654 EDT
Building configuration...
interface HundredGigE 0/5/0/0
  shutdown
!
interface HundredGigE 0/5/0/0.0
  encapsulation dot1q 100
!
```

```
interface HundredGigE 0/5/0/1
 shutdown
!
```

When a subinterface is first created, the router recognizes it as an interface that, with few exceptions, is interchangeable with a trunk interface. After the new subinterface is configured further, the **show interface** command can display it along with its unique counters:

The following example shows the display output for the trunk interface, HundredGigE 0/5/0/0, followed by the display output for the subinterface HundredGigE 0/5/0/0.0.

```
RP/0/RSP0/CPU0:router# show interface HundredGigE 0/5/0/0
Mon Sep 21 11:12:51.068 EDT
HundredGigE0/5/0/0 is administratively down, line protocol is administratively down.
Interface state transitions: 0
Hardware is HundredGigE, address is 0024.f71b.0ca8 (bia 0024.f71b.0ca8)
Internet address is Unknown
MTU 1514 bytes, BW 1000000 Kbit
    reliability 255/255, txload 0/255, rxload 0/255
Encapsulation 802.1Q Virtual LAN,
Full-duplex, 1000Mb/s, SXFD, link type is force-up
output flow control is off, input flow control is off
loopback not set,
ARP type ARPA, ARP timeout 04:00:00
Last input never, output never
Last clearing of "show interface" counters never
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
    0 packets input, 0 bytes, 0 total input drops
    0 drops for unrecognized upper-level protocol
Received 0 broadcast packets, 0 multicast packets
    0 runts, 0 giants, 0 throttles, 0 parity
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
0 packets output, 0 bytes, 0 total output drops
Output 0 broadcast packets, 0 multicast packets
0 output errors, 0 underruns, 0 applique, 0 resets
0 output buffer failures, 0 output buffers swapped out
0 carrier transitions
```

```
RP/0/RSP0/CPU0:router# show interface HundredGigE0/5/0/0.0
Mon Sep 21 11:12:55.657 EDT
HundredGigE0/5/0/0.0 is administratively down, line protocol is administratively down.
Interface state transitions: 0
Hardware is VLAN sub-interface(s), address is 0024.f71b.0ca8
Internet address is Unknown
MTU 1518 bytes, BW 1000000 Kbit
    reliability 255/255, txload 0/255, rxload 0/255
Encapsulation 802.1Q Virtual LAN, VLAN Id 100, loopback not set,
ARP type ARPA, ARP timeout 04:00:00
Last input never, output never
Last clearing of "show interface" counters never
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
    0 packets input, 0 bytes, 0 total input drops
    0 drops for unrecognized upper-level protocol
Received 0 broadcast packets, 0 multicast packets
0 packets output, 0 bytes, 0 total output drops
Output 0 broadcast packets, 0 multicast packets
```

This example shows two interfaces being created at the same time: first, the bundle trunk interface, then a subinterface attached to the trunk:

```

RP/0/RSP0/CPU0:router# conf
Mon Sep 21 10:57:31.736 EDT
RP/0/RSP0/CPU0:router(config)# interface Bundle-Ether1
RP/0/RSP0/CPU0:router(config-if)# no shut
RP/0/RSP0/CPU0:router(config-if)# interface bundle-Ether1.0
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 100
RP/0/RSP0/CPU0:router(config-subif)# commit
RP/0/RSP0/CPU0:Sep 21 10:58:15.305 : config[65794]: %MGBL-CONFIG-6-DB_COMMIT : C
onfiguration committed by user 'root'. Use 'show configuration commit changes 10
00000149' to view the changes.
RP/0/RSP0/CPU0:router# show run | begin Bundle-Ether1
Mon Sep 21 10:59:31.317 EDT
Building configuration..
interface Bundle-Ether1
!
interface Bundle-Ether1.0
  encapsulation dot1q 100
!

```

You delete a subinterface using the **no interface** command.

```

RP/0/RSP0/CPU0:router#
RP/0/RSP0/CPU0:router# show run | begin HundredGigE 0/5/0/0
Mon Sep 21 11:42:27.100 EDT
Building configuration...
interface HundredGigE 0/5/0/0
  negotiation auto
!
interface HundredGigE 0/5/0/0.0
  encapsulation dot1q 100
!
interface HundredGigE 0/5/0/1
  shutdown
!
RP/0/RSP0/CPU0:router# conf
Mon Sep 21 11:42:32.374 EDT
RP/0/RSP0/CPU0:router(config)# no interface HundredGigE 0/5/0/0.0
RP/0/RSP0/CPU0:router(config)# commit
RP/0/RSP0/CPU0:Sep 21 11:42:47.237 : config[65794]: %MGBL-CONFIG-6-DB_COMMIT : Configuration
committed by user 'root'. Use 'show configuration commit changes 1000000159' to view the
changes.
RP/0/RSP0/CPU0:router(config)# end
RP/0/RSP0/CPU0:Sep 21 11:42:50.278 : config[65794]: %MGBL-SYS-5-CONFIG_I : Configured from
console by root
RP/0/RSP0/CPU0:router# show run | begin HundredGigE 0/5/0/0
Mon Sep 21 11:42:57.262 EDT
Building configuration...
interface HundredGigE 0/5/0/0
  negotiation auto
!
interface HundredGigE 0/5/0/1
  shutdown
!

```

Layer 2, Layer 3, and EFPs

On the router, a trunk interface can be either a Layer 2 or Layer 3 interface. A Layer 2 interface is configured using the **interface** command with the **l2transport** keyword. When the **l2transport** keyword is not used, the interface is a Layer 3 interface. Subinterfaces are configured as Layer 2 or Layer 3 subinterface in the same way.

A Layer 3 trunk interface or subinterface is a routed interface and can be assigned an IP address. Traffic sent on that interface is routed.

A Layer 2 trunk interface or subinterface is a switched interface and cannot be assigned an IP address. A Layer 2 interface must be connected to an L2VPN component. Once it is connected, it is called an access connection.

Subinterfaces can only be created under a Layer 3 trunk interface. Subinterfaces cannot be created under a Layer 2 trunk interface.

A Layer 3 trunk interface can have any combination of Layer 2 and Layer 3 interfaces.

The following example shows an attempt to configure a subinterface under an Layer 2 trunk and the commit errors that occur. It also shows an attempt to change the Layer 2 trunk interface to an Layer 3 interface and the errors that occur because the interface already had an IP address assigned to it.

```
RP/0/RP0/CPU0:router# config
Mon Sep 21 12:05:33.142 EDT
RP/0/RP0/CPU0:router(config)# interface HundredGigE0/5/0/0
RP/0/RP0/CPU0:router(config-if)# ipv4 address 10.0.0.1/24
RP/0/RP0/CPU0:router(config-if)# commit
RP/0/RP0/CPU0:Sep 21 12:05:57.824 : config[65794]: %MGBL-CONFIG-6-DB_COMMIT : Configuration
  committed by user 'root'. Use 'show configuration commit changes 1000000160' to view the
  changes.
RP/0/RP0/CPU0:router(config-if)# end
RP/0/RP0/CPU0:Sep 21 12:06:01.890 : config[65794]: %MGBL-SYS-5-CONFIG_I : Configured from
  console by root
RP/0/RP0/CPU0:router# show run | begin HundredGigE0/5/0/0
Mon Sep 21 12:06:19.535 EDT
Building configuration...
interface HundredGigE0/5/0/0
  ipv4 address 10.0.0.1 255.255.255.0
  negotiation auto
!
interface HundredGigE0/5/0/1
  shutdown
!
RP/0/RP0/CPU0:router#
RP/0/RP0/CPU0:router#
RP/0/RP0/CPU0:router# conf
Mon Sep 21 12:08:07.426 EDT
RP/0/RP0/CPU0:router(config)# interface HundredGigE0/5/0/0 l2transport
RP/0/RP0/CPU0:router(config-if-l2)# commit

% Failed to commit one or more configuration items during a pseudo-atomic operation. All
  changes made have been reverted. Please issue 'show configuration failed' from this session
  to view the errors
RP/0/RP0/CPU0:router(config-if-l2)# no ipv4 address
RP/0/RP0/CPU0:router(config-if)# commit
RP/0/RP0/CPU0:Sep 21 12:08:33.686 : config[65794]: %MGBL-CONFIG-6-DB_COMMIT : Configuration
  committed by user 'root'. Use 'show configuration commit changes 1000000161' to view the
  changes.
RP/0/RP0/CPU0:router(config-if)# end
RP/0/RP0/CPU0:Sep 21 12:08:38.726 : config[65794]: %MGBL-SYS-5-CONFIG_I : Configured from
  console by root
RP/0/RP0/CPU0:router#
RP/0/RP0/CPU0:router# show run interface HundredGigE0/5/0/0
Mon Sep 21 12:09:02.471 EDT
interface HundredGigE0/5/0/0
  negotiation auto
  l2transport
!
```

```

!
RP/0/RP0/CPU0:router#
RP/0/RP0/CPU0:router# conf
Mon Sep 21 12:09:08.658 EDT
RP/0/RP0/CPU0:router(config)# interface HundredGigE0/5/0/0.0
                                     ^
RP/0/RP0/CPU0:router(config)# interface HundredGigE0/5/0/0.0
RP/0/RP0/CPU0:router(config-subif)# commit

% Failed to commit one or more configuration items during a pseudo-atomic operation. All
changes made have been reverted. Please issue 'show configuration failed' from this session
to view the errors
RP/0/RP0/CPU0:router(config-subif)#
RP/0/RP0/CPU0:router(config-subif)# interface HundredGigE0/5/0/0
RP/0/RP0/CPU0:router(config-if)# no l2transport
RP/0/RP0/CPU0:router(config-if)# interface HundredGigE0/5/0/0.0
RP/0/RP0/CPU0:router(config-subif)# encapsulation dot1q 99
RP/0/RP0/CPU0:router(config-subif)# ipv4 address 11.0.0.1/24
RP/0/RP0/CPU0:router(config-subif)# interface HundredGigE0/5/0/0.1 l2transport
RP/0/RP0/CPU0:router(config-subif)# encapsulation dot1q 700
RP/0/RP0/CPU0:router(config-subif)# commit
RP/0/RP0/CPU0:Sep 21 12:11:45.896 : config[65794]: %MGBL-CONFIG-6-DB_COMMIT : Configuration
committed by user 'root'. Use 'show configuration commit changes 1000000162' to view the
changes.
RP/0/RP0/CPU0:router(config-subif)# end
RP/0/RP0/CPU0:Sep 21 12:11:50.133 : config[65794]: %MGBL-SYS-5-CONFIG_I : Configured from
console by root
RP/0/RP0/CPU0:router#
RP/0/RP0/CPU0:router# show run | b HundredGigE0/5/0/0
Mon Sep 21 12:12:00.248 EDT
Building configuration...
interface HundredGigE0/5/0/0
  negotiation auto
!
interface HundredGigE0/5/0/0.0
  ipv4 address 11.0.0.1 255.255.255.0
  encapsulation dot1q 99
!
interface HundredGigE0/5/0/0.1 l2transport
  encapsulation dot1q 700
!
interface HundredGigE0/5/0/1
  shutdown
!

```

All subinterfaces must have unique encapsulation statements, so that the router can send incoming packets and frames to the correct subinterface. If a subinterface does not have an encapsulation statement, the router will not send any traffic to it.

In Cisco IOS XR, an Ethernet Flow Point (EFP) is implemented as a Layer 2 subinterface, and consequently, a Layer 2 subinterface is often called an EFP.

A Layer 2 trunk interface can be used as an access connection. However, a Layer 2 trunk interface is not an EFP because an EFP, by definition, is a substream of an overall stream of traffic.

Cisco IOS XR also has other restrictions on what can be configured as a Layer 2 or Layer 3 interface. Certain configuration blocks only accept Layer 3 and not Layer 2. For example, OSPF only accepts Layer 3 trunks and subinterface. Refer to the appropriate Cisco IOS XR configuration guide for other restrictions.

Untagged L2 subinterface

Table 13: Feature History Table

Feature Name	Release Information	Feature Description
Untagged L2 subinterface	Release 25.4.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8011-12G12X4Y-A • 8011-12G12X4Y-D
Untagged L2 subinterface	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100])</p> <p>This feature support is now extended to the 8011-4G24Y4H-I. routers.</p>
Untagged L2 subinterface	Release 24.4.1	<p>Introduced in this release on: Fixed Systems(8700)</p> <p>This feature support is now extended to the Cisco 8712-MOD-M routers.</p>
Untagged L2 subinterface	Release 24.3.1	<p>Introduced in this release on: Fixed Systems(8200, 8700)(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature support is now extended to:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-52Y8H-EM • 88-LC1-12TH24FH-E

Untagged L2 subinterface	Release 24.2.11	<p>Introduced in this release on: Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>You can now use untagged L2 subinterfaces to effectively manage and process traffic from customer edge (CE) devices that do not employ VLAN tagging. This capability allows you to apply services to untagged packets, which would not have been possible if the packets were to be logically received on the main interface. As a result, you can now push a dot1q or other supported Layer 2 encapsulation on the received frame.</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 88-LC1-36EH <p>This feature introduces the encapsulation untagged command.</p>
--------------------------	-----------------	--

Untagged L2 subinterface

An untagged L2 subinterface is a subinterface on a network device that isn't linked to a VLAN tag. The VLAN tags divide network traffic into smaller, distinct networks, which helps improve security and manageability in a bigger network setup.

Challenges in Processing Traffic without Untagged L2 Subinterfaces

You cannot perform the following actions on an untagged frame received on the main interface:

- Apply commands available only to l2transport subinterfaces, such as popping or pushing a dot1q or dot1ad header.
- Attach a service policy because you cannot simultaneously attach a service policy to the main interface and subinterfaces of that interface.

Untagged L2 Subinterface Offerings

The untagged subinterface takes a higher priority than the main interface on service mapping. When both interfaces are created, the untagged traffic is mapped to the untagged subinterface.

Untagged Traffic Mapped to an Untagged Subinterface

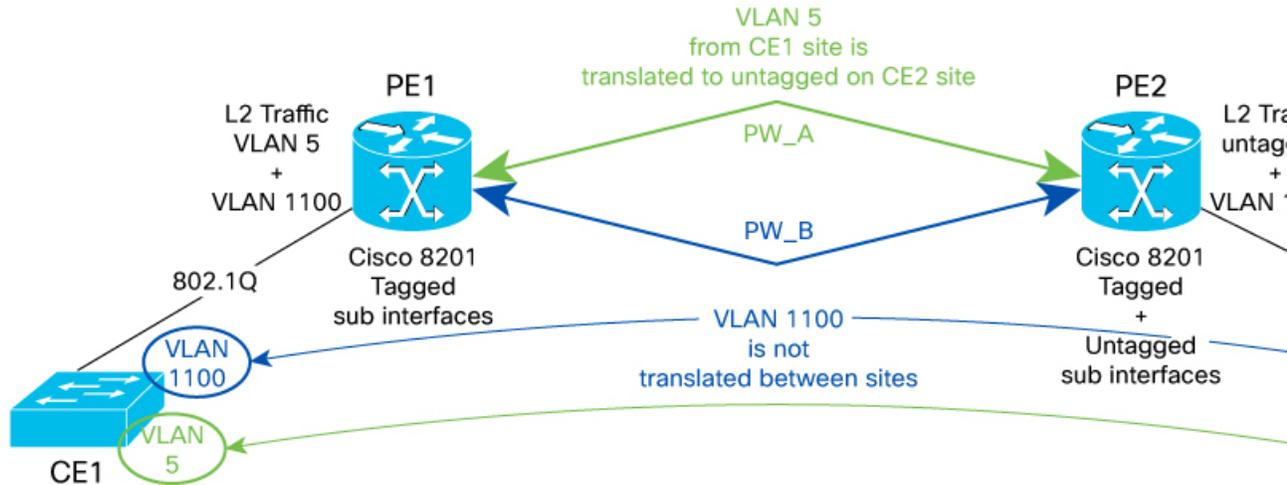
Traffic using an unsupported VLAN tag format is considered untagged traffic and mapped to an untagged subinterface. The following VLAN tag formats are supported:

- Single tag format - dot1q
- Double tag format - outer dot1ad paired with inner dot1q tag, or outer dot1q tag paired with inner dot1q tag
- Three or more tags - outermost dot1ad tag followed by dot1q tag, or outermost dot1q tag followed by dot1q tag

Benefits of Untagged L2 Subinterfaces

- Manages untagged and tagged traffic on the same port to different services.
- Allows rewriting of different VLANs on untagged subinterfaces for both tagged and untagged traffic.

How does the Untagged L2 subinterface Work?



Let's understand the feature using this sample topology, in which the P2P L2VPN service is deployed between the two CE devices.

In this topology,

- VLAN 1100 is a common VLAN that is shared between CE1 and CE2.
PE nodes do not require VLAN translation. However, if a VLAN ID is not common on CE1 and CE2, PE must perform VLAN translation.
- For instance, VLAN 5 is known only to CE1, and for the same service, CE1 uses VLAN 5, while CE2 uses no VLAN. In this situation, PE1 performs VLAN translation.
- When traffic is sent from CE1 to CE2, PE1 removes the VLAN 5 tag on the traffic from CE1 and then sends untagged traffic to PE2 through the pseudowire (PW). PE2 disposes the PW header and forwards untagged traffic to CE2.
- When a traffic is sent from CE2 to CE1, PE2 maps untagged traffic from CE2 to an L2 subinterface and sends the traffic to PE1 through the PW. PE1 disposes of the PW header and adds the VLAN 5 tag before switching the traffic to CE1.

Configure Untagged L2 Subinterface

To manage traffic efficiently, configure untagged L2 subinterface.

Procedure

- Step 1** Create an untagged L2 subinterface with the **encapsulation untagged** keyword under an L2 transport main interface.

This feature is only applicable to L2 transport main and subinterfaces and does not support any other encapsulation types on the same L2 subinterface.

Example:

```
Router# configure
Router(config)# interface HundredGigE0/0/16.1500 l2transport
Router(config-subif)# encapsulation untagged
Router(config-subif)# commit
```

Step 2 Verify that the untagged encapsulation is configured on the L2 subinterface.

Example:

```
Router# show interfaces HundredGigE0/0/16.1500
HundredGigE0/0/16.1500 is up, line protocol is up
Interface state transitions: 1
Hardware is VLAN sub-interface(s), address is 0029.c201.3f0c
Internet address is 40.40.50.1/24
MTU 1522 bytes, BW 100000000 Kbit (Max: 100000000 Kbit)
reliability 255/255, txload 0/255, rxload 0/255
Encapsulation Untagged Virtual LAN,
Last link flapped 00:01:25
ARP type ARPA, ARP timeout 04:00:00
Last input never, output never
Last clearing of "show interface" counters never
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
0 packets input, 0 bytes, 0 total input drops
0 drops for unrecognized upper-level protocol
Received 0 broadcast packets, 0 multicast packets
0 packets output, 0 bytes, 0 total output drops
Output 0 broadcast packets, 0 multicast packets
```

Enhanced Performance Monitoring for Layer 2 Subinterfaces (EFPs)

Beginning in Cisco IOS XR Release 7.2.12, the router adds support for basic counters for performance monitoring on Layer 2 subinterfaces. This section provides a summary of the new support for Layer 2 interface counters.

The **interface basic-counters** keyword has been added to support a new entity for performance statistics collection and display on Layer 2 interfaces in the following commands:

- **performance-mgmt statistics interface basic-counters**
- **performance-mgmt threshold interface basic-counters**
- **performance-mgmt apply statistics interface basic-counters**
- **performance-mgmt apply threshold interface basic-counters**
- **performance-mgmt apply monitor interface basic-counters**
- show performance-mgmt monitor interface basic-counters
- show performance-mgmt statistics interface basic-counters

The **performance-mgmt threshold interface basic-counters** command supports the following attribute values for Layer 2 statistics, which also appear in the **show performance-mgmt statistics interface basic-counters** and **show performance-mgmt monitor interface basic-counters** command:

Attribute	Description
InOctets	Bytes received (64-bit)
InPackets	Packets received (64-bit)
InputQueueDrops	Input queue drops (64-bit)
InputTotalDrops	Inbound correct packets discarded (64-bit)
InputTotalErrors	Inbound incorrect packets discarded (64-bit)
OutOctets	Bytes sent (64-bit)
OutPackets	Packets sent (64-bit)
OutputQueueDrops	Output queue drops (64-bit)
OutputTotalDrops	Outband correct packets discarded (64-bit)
OutputTotalErrors	Outband incorrect packets discarded (64-bit)

Other Performance Management Enhancements

The following additional performance management enhancements are included in Cisco IOS XR Release 7.0.11:

- You can retain performance management history statistics across a process restart or route processor (RP) failover using the new **history-persistent** keyword option for the **performance-mgmt statistics interface** command.
- You can save performance management statistics to a local file using the **performance-mgmt resources dump local** command.
- You can filter performance management instances by defining a regular expression group (**performance-mgmt regular-expression** command), which includes multiple regular expression indices that specify strings to match. You apply a defined regular expression group to one or more statistics or threshold templates in the **performance-mgmt statistics interface** or **performance-mgmt thresholds interface** commands.

Frequency Synchronization and SyncE

Cisco IOS XR Software supports SyncE-capable Ethernet on the router. Frequency Synchronization enables you to distribute the precision clock signals around the network. The system injects a highly accurate timing signal into the router in the network. The timing signals use an external timing technology, such as Cesium atomic clocks, or GPS, and then pass the signals to the physical interfaces of the router. Peer routers can then recover this precision frequency from the line, and also transfer it around the network. This feature is traditionally applicable to SONET or SDH networks, but is now available on Ethernet for Cisco 8000 Series

Router with Synchronous Ethernet capability. For more information, see *Cisco 8000 Series Router System Management Configuration Guide*.

LLDP

Table 14: Feature History Table

Feature Name	Release Information	Feature Description
LLDP Snooping	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8712-MOD-M • 8011-4G24Y4H-I
LLDP Snooping	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-12TH24FH-E • 88-LC1-36EH • 88-LC1-52Y8H-EM

Feature Name	Release Information	Feature Description
LLDP Snooping	Release 7.3.3	<p>With this release, you can further leverage the Link Layer Discovery Protocol (LLDP) information for directly attached devices or equipment in an L2 (Layer 2) network via LLDP snoop. In order to utilize the LLDP snoop functionality, the neighbouring devices must exchange the LLDP packets with the L2 network.</p> <p>With the help of the LLDP snoop functionality, you can identify the cabling and modeling failures and isolate faults.</p> <p>To enable LLDP snoop, enable the LLDP command on an interface while the outgoing (TX) traffic is disabled.</p>

The Cisco Discovery Protocol (CDP) is a device discovery protocol that runs over Layer 2 (the Data Link layer) on all Cisco-manufactured devices (routers, bridges, access servers, and switches). CDP allows network management applications to automatically discover and learn about other Cisco devices connected to the network.

To support non-Cisco devices and to allow for interoperability between other devices, the router also supports the IEEE 802.1AB Link Layer Discovery Protocol (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.

LLDP supports a set of attributes that it uses to learn information about neighbor devices. These attributes have a defined format known as a Type-Length-Value (TLV). LLDP supported devices can use TLVs to receive and send information to their neighbors. Details such as configuration information, device capabilities, and device identity can be advertised using this protocol.

In addition to the mandatory TLVs (Chassis ID, Port ID, End of LLDPDU, and Time-to-Live), the router also supports the following basic management TLVs, which are optional:

- Port Description
- System Name
- System Description
- System Capabilities
- Management Address

These optional TLVs are automatically sent when LLDP is active, but you can disable them as needed using the **lldp tlv-select <Optional TLV> disable** command.



Note MACsec encrypts LLDP packets by default. You can enable exceptions in the MACsec policy using the **allow lldp-in-clear** command to retain the LLDP packets unencrypted with MACsec. For more information, see *MACsec Policy Exceptions for Link Layer Discovery Protocol Packets* section in the *Configuring MACsec* chapter of the *System Security Configuration Guide for Cisco 8000 Series Routers*.



Note For LLDP to work on any bundle member, enable LLDP on the bundle main interface either globally or on the interface itself. You can then choose to disable LLDP transmission on bundle main interface by using the `lldp transmit disable` command.

You can also control LLDP transmit or receive on each bundle member interface as desired.

LLDP Frame Format

LLDP frames use the IEEE 802.3 format, which consists of the following fields:

- Destination address (6 bytes)—Uses a multicast address of 01-80-C2-00-00-0E.
- Source address (6 bytes)—MAC address of the sending device or port.
- LLDP Ethertype (2 bytes)—Uses 88-CC.
- LLDP PDU (1500 bytes)—LLDP payload consisting of TLVs.
- FCS (4 bytes)—Cyclic Redundancy Check (CRC) for error checking.

LLDP TLV Format

LLDP TLVs carry the information about neighboring devices within the LLDP PDU using the following basic format:

- TLV Header (16 bits), which includes the following fields:
 - TLV Type (7 bits)
 - TLV Information String Length (9 bits)
- TLV Information String (0 to 511 bytes)

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 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.
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 8000 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
```

LLDP Operation

LLDP is a one-way protocol. The basic operation of LLDP consists of a device enabled for transmit of LLDP information sending periodic advertisements of information in LLDP frames to a receiving device.

Devices are identified using a combination of the Chassis ID and Port ID TLVs to create an MSAP (MAC Service Access Point). The receiving device saves the information about a neighbor in a remote lldp cache for a certain amount of time as specified in the TTL TLV received from the neighbor, before aging and removing the information.

LLDP supports the following additional operational characteristics:

- LLDP can operate independently in transmit or receive modes. On global lldp enablement, the default mode is to operate in both transmit and receive modes.
- LLDP operates as a slow protocol with transmission speeds not greater than one frame per five seconds.
- LLDP packets are sent when the following occurs:
 - The packet update frequency specified by the **lldp timer** command is reached. The default is 30 seconds.
 - When a change in the values of the managed objects occurs from the local system's LLDP MIB.
 - When LLDP is activated on an interface (3 frames are sent upon activation similar to CDP).
- When an LLDP frame is received, the LLDP remote services and PTOPO MIBs are updated with the information in the TLVs.
- LLDP supports the following actions on these TLV characteristics:
 - Interprets a neighbor TTL value of 0 as a request to automatically purge the information of the transmitting device. These shutdown LLDPDUs are typically sent prior to a port becoming inoperable.
 - An LLDP frame with a malformed mandatory TLV is dropped.
 - A TLV with an invalid value is ignored.
 - A copy of an unknown organizationally-specific TLV is maintained if the TTL is non-zero, for later access through network management.

Supported LLDP Functions

The router supports the following LLDP functions:

- IPv4 and IPv6 management addresses—In general, both IPv4 and IPv6 addresses will be advertised if they are available, and preference is given to the address that is configured on the transmitting interface.

If the transmitting interface does not have a configured address, then the system populates the TLV with an address from another interface. The advertised LLDP IP address is implemented according to the following priority order of IP addresses for interfaces on the router:

- Locally configured address on the transmitting interface
- MgmtEth0/RSP0RP0/CPU0/0
- MgmtEth0/RSP0RP0/CPU0/1
- MgmtEth0/RSP1RP1/CPU0/0
- MgmtEth0/RSP1RP1/CPU0/1
- Loopback interfaces

There are some differences between IPv4 and IPv6 address management in LLDP:

- For IPv4, as long as the IPv4 address is configured on an interface, it can be used as an LLDP management address.
- For IPv6, after the IPv6 address is configured on an interface, the interface status must be Up and pass the DAD (Duplicate Address Detection) process before it can be used as an LLDP management address.
- LLDP is supported for the nearest physically attached, non-tunneled neighbors.
- LLDP is supported for Ethernet interfaces, L3 subinterfaces, bundle interfaces, and L3 bundle subinterfaces.
- LLDP snoop is supported on L2 interfaces, when the incoming (RX) traffic is enabled and outgoing (TX) traffic is disabled.

Unsupported LLDP Functions

The following LLDP functions are not supported on the router:

- LLDP-MED organizationally unique extension—However, interoperability still exists between other devices that do support this extension.
- Tunneled neighbors, or neighbors more than one hop away.
- LLDP TLVs cannot be disabled on a per-interface basis; However, certain optional TLVs can be disabled globally.
- LLDP SNMP trap `lldpRemTablesChange`.

Setting the carrier delay on physical interfaces

Table 15: Feature History Table

Feature Name	Release Information	Feature Description
Default carrier delay value on physical interfaces	Release 25.4.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100])(select variants only*)</p> <p>*This feature is now supported on Cisco 8711-48Z-M routers.</p>
Default carrier delay value on physical interfaces	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on Cisco 8011-4G24Y4H-I routers.</p>
Default carrier delay value on physical interfaces	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100, K100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>You can now set the carrier-delay up default value on the listed hardware that provides sufficient time to establish a stable hardware link state. If you have not set the default timer, the default carrier delay automatically delays the hardware link-up notifications by 200 ms.</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 8712-MOD-M • 88-LC1-12TH24FH-E • 88-LC1-36EH+A8:B12 • 88-LC1-52Y8H-EM

Feature Name	Release Information	Feature Description
Default carrier delay value on physical interfaces	Release 24.3.1	<p>Introduced in this release on: Fixed Systems(8200, 8700)(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*).</p> <p>We have introduced the carrier-delay up default value to ensure enough time to establish a stable hardware link state. If you haven't configured the timer, the default carrier delay automatically delays the hardware link-up notifications by 200 ms.</p> <p>Previously, we recommended that you set the carrier delay-up timer to 10 ms.</p> <p>If you want to change the delay of the interface state change notification, you can use the carrier-delay command to set a different value.</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-12TH24FH-E • 88-LC1-52Y8H-EM
Default carrier delay value on physical interfaces	Release 24.2.11	<p>Introduced in this release on: ; Modular Systems (8800 [LC ASIC: P100])(select variants only*).</p> <p>We have introduced the carrier-delay up default value to ensure enough time to establish a stable hardware link state. If you haven't configured the timer, the default carrier delay automatically delays the hardware link-up notifications by 200 ms.</p> <p>Previously, we recommended that you set the carrier delay-up timer to 10 ms.</p> <p>If you want to change the delay of the interface state change notification, you can use the carrier-delay command to set a different value.</p> <p>*This feature is now supported on 88-LC1-36EH.</p>

Feature Name	Release Information	Feature Description
Setting the carrier delay on physical interfaces	Release 7.5.4	<p>You can configure the Ethernet interfaces to delay the processing of hardware link-down and link-up notifications. With this functionality, the interface state remains stable for the configured delay duration, even if the hardware link state fluctuates. This prevents interface flapping and improves network reliability.</p> <p>Use the following CLI command in interface configuration mode to configure the delay time:</p> <p>carrier-delay</p>

Hardware links take time to stabilize after a state change and may experience link flaps. Link flap is a condition where a physical interface frequently fluctuates between an up and a down state.

During link flaps, the network reestablishes and updates routing paths after a disruption, which leads to resource exhaustion on routers. To overcome the problem, we recommend waiting until the link state is stable before taking action.

The carrier delay introduces a delay in processing interface link-state notifications in the router to provide enough time for the interface link to stabilize.

When there is a change in the link state, the carrier-delay timer starts. If the link state goes up, the **carrier-delay up** timer starts. Similarly, when the link state goes down, the **carrier-delay down** timer starts. During this delay period, the Ethernet interface state remains unchanged even if the link is physically restored. Setting a delay timer ensures the link state is established before the interface becomes operational again and avoids unnecessary interface state changes and associated traffic rerouting.

Guidelines and Restrictions for Setting the Carrier Delay on Physical Interfaces

The following usage guidelines and restrictions are applicable for setting the carrier delay on physical interfaces:

- You can configure carrier-delay for only link-up, only link-down, or both link-up and link-down notifications.
- If the **carrier-delay down milliseconds** command is configured on a physical link that fails and cannot be recovered, link down detection time increases, and it may take longer for the routing protocols to reroute the traffic around the failed link.
- Loss of Signal (LOS) is not supported on carrier delay.
- From Release 24.2.11, the default value of carrier-delay up parameter is changed to 200 ms. To restore your original configuration, you need to configure the parameter explicitly using **carrier-delay** command.
- If not configured, the carrier-delay up parameter defaults to 200 ms and the carrier-delay down parameter to 0 ms. When carrier-delay down is not configured, the higher-layer protocols are notified immediately when a physical link state changes.

- The **carrier-delay** command overwrites the previous configuration every time you execute the command. For example, if you already have the **carrier-delay up** configured and later configure the **carrier-delay down**, the carrier-delay down overwrites the previously configured carrier-delay up. However, you can configure the carrier delay up and down concurrently using the **carrier-delay up (milliseconds) down (milliseconds)** command.

Configure the Carrier-delay Timer

Default Configuration Example

In this example, one interface is brought up to check the default value of link-up notification delay.

```
Router#configure
Router(config)#interface HundredGigE 0/0/0/0
Router(config-if)#no shutdown
Router(config-if)#commit
```

Run the **show interfaces** command to check if the carrier-delay configuration for the interface defaults to 200 ms.

```
Router#show interfaces HundredGigE 0/0/0/0 | include Carrier
Fri Mar 31 07:25:05.273 UTC
Carrier delay (up) is 200 msec
```

Configuration Example

In this example, link-up and link-down notifications are configured to be delayed by 1000 ms and 150 ms using **carrier-delay** command.

```
Router#configure
Router(config)#interface HundredGigE 0/0/0/0
Router(config-if)#carrier-delay up 1000 down 150
Router(config-if)#commit
```

Running Configuration

```
interface HundredGigE0/0/0/0
carrier-delay up 1000 down 150
!
```

Verification

Run the **show interfaces** command to see the current state of the carrier-delay configuration for an interface.

```
Router#show interfaces HundredGigE 0/0/0/0 | include Carrier
Fri Mar 31 07:25:05.273 UTC
Carrier delay (up) is 1000 msec, Carrier delay (down) is 150 msec
```

How to Configure Ethernet

This section provides the following configuration procedures:

Configuring LLDP



Note LLDP is not supported on the FP-X line cards.

This section includes the following configuration topics for LLDP:

LLDP Default Configuration

This table shows the values of the LLDP default configuration on the router. To change the default settings, use the LLDP global configuration and LLDP interface configuration commands.

LLDP Function	Default
LLDP global state	Disabled
LLDP holdtime (before discarding)	120 seconds
LLDP timer (packet update frequency)	30 seconds
LLDP reinitialization delay	2 seconds
LLDP TLV selection	All TLVs are enabled for sending and receiving.
LLDP interface state	Enabled for both transmit and receive operation when LLDP is globally enabled.

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:

```
RP/0/RSP0/CPU0:ios(config)# int HundredGigE 0/2/0/0
RP/0/RSP0/CPU0:ios(config-if)# no sh
RP/0/RSP0/CPU0:ios(config-if)#commit
RP/0/RSP0/CPU0:ios(config-if)#lldp ?
RP/0/RSP0/CPU0:ios(config-if)#lldp enable
RP/0/RSP0/CPU0:ios(config-if)#commit
```

Running configuration

```
RP/0/RSP0/CPU0:ios#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 HundredGigE0/1/0/0
 shutdown
!
interface HundredGigE0/1/0/1
 shutdown
!
interface HundredGigE0/1/0/2
 shutdown
!
```

```

interface HundredGigE0/2/0/0
 Shutdown
!
interface HundredGigE0/2/0/1
 shutdown
!
interface HundredGigE0/2/0/2
 shutdown
!
end

```

Verification

Verifying the config

=====

```

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

```

```

HundredGigE0/2/0/0:
  Tx: enabled
  Rx: enabled
  Tx state: IDLE
  Rx state: WAIT FOR FRAME
RP/0/RSP0/CPU0:ios#

```

```

RP/0/RSP0/CPU0:ios# 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 neighborhood seen for the same.

```

```
Total entries displayed: 1
```

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

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. For more information about how to selectively disable LLDP receive or transmit operations for an interface, see the *Disabling LLDP Receive and Transmit Operation for an Interface* section.



Note For LLDP to work on any bundle member, enable LLDP on the bundle main interface either globally or on the interface itself. You can then choose to disable LLDP transmission on bundle main interface by using the `lldp transmit disable` command.

You can also control LLDP transmit or receive on each bundle member interface as desired.

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

Running configuration

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

```
RP/0/RP0/CPU0:turin-5#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:turin-5#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

Configuring Global LLDP Operational Characteristics

When you enable LLDP globally on the router using the **lldp** command, these defaults are used for the protocol.

To modify the global LLDP operational characteristics such as the LLDP neighbor information holdtime, initialization delay, or packet rate, complete the following steps:

Procedure

Step 1 Example:

```
/CPU0:router# configure
```

Enters global configuration mode.

Step 2 **lldp holdtime** *seconds*

Example:

```
RP/0//CPU0:router(config)#lldp holdtime 60
```

(Optional) Specifies the length of time that information from an LLDP packet should be held by the receiving device before aging and removing it.

Step 3 **lldp reinit** *seconds*

Example:

```
RP/0//CPU0:router(config)# lldp reinit 4
```

(Optional) Specifies the length of time to delay initialization of LLDP on an interface.

Step 4 **lldp timer** *seconds*

Example:

```
RP/0//CPU0:router(config)#lldp reinit 60
```

(Optional) Specifies the LLDP packet rate.

Step 5 **end** or **commit**

Example:

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

or

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

Disabling Transmission of Optional LLDP TLVs

Certain TLVs are classified as mandatory in LLDP packets, such as the Chassis ID, Port ID, and Time to Live (TTL) TLVs. These TLVs must be present in every LLDP packet. You can suppress transmission of certain other optional TLVs in LLDP packets.

To disable transmission of optional LLDP TLVs, complete the following steps:

Procedure

Step 1 **configure**

Example:

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

Enters global configuration mode.

Step 2 **lldp tlv-select *tlv-name* disable**

Example:

```
RP/0/RSP0/CPU0:router(config)# lldp tlv-select system-capabilities disable
```

(Optional) Specifies that transmission of the selected TLV in LLDP packets is disabled. The *tlv-name* can be one of the following LLDP TLV types:

- **management-address**
- **port-description**
- **system-capabilities**
- **system-description**
- **system-name**

Step 3 **end or commit**

Example:

```
RP/0/RSP0/CPU0:router(config)# end
```

or

```
RP/0/RSP0RSP0/CPU0:router(config)# 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.

Disabling LLDP Receive and Transmit Operation for an Interface

When you enable LLDP globally on the router, all supported interfaces are automatically enabled for LLDP receive and transmit operation. You can override this default by disabling these operations for a particular interface.

To disable LLDP receive and transmit operations for an interface, complete the following steps:

Procedure

Step 1 **configure**

Example:

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

Enters global configuration mode.

Step 2 **interface HundredGigE 0/2/0/0**

Example:

```
RP/0/RSP0RP0/CPU0:router(config)#interface HundredGigE 0/2/0/0
```

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

- HundredGigE
- TenGigE

Step 3 **lldp**

Example:

```
RP/0/RSP0/CPU0:router(config-if)#lldp
```

(Optional) Enters LLDP configuration mode for the specified interface.

Step 4 **receive disable**

Example:

```
RP/0/RSP0/CPU0:router(config-lldp)#receive disable
```

(Optional) Disables LLDP receive operations on the interface.

Step 5 **transmit disable**

Example:

```
RP/0/RSP0/CPU0:router(config-lldp)#transmit disable
```

(Optional) Disables LLDP transmit operations on the interface.

Step 6 **end or commit**

Example:

```
RP/0/RSP0/CPU0:router(config)# end
```

or

```
RP/0/RSP0/CPU0:router(config)# 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 the LLDP Configuration

This section describes how you can verify the LLDP configuration both globally and for a particular interface.

Verifying the LLDP Global Configuration

To verify the LLDP global configuration status and operational characteristics, use the **show lldp** command as shown in the following example:

```
RP/0/RSP0/CPU0:router# show lldp
Wed Apr 13 06:16:45.510 DST
Global LLDP information:
  Status: ACTIVE
  LLDP advertisements are sent every 30 seconds
  LLDP hold time advertised is 120 seconds
  LLDP interface reinitialisation delay is 2 seconds
```

If LLDP is not enabled globally, the following output appears when you run the **show lldp** command:

```
RP/0/RSP0/CPU0:router# show lldp
Wed Apr 13 06:42:48.221 DST
% LLDP is not enabled
```

Verifying the LLDP Interface Configuration

To verify the LLDP interface status and configuration, use the **show lldp interface** command as shown in the following example:

```
RP/0/RSP0/CPU0:router# show lldp interface HundredGigE 0/1/0/7
Wed Apr 13 13:22:30.501 DST

HundredGigE0/1/0/7:
  Tx: enabled
  Rx: enabled
  Tx state: IDLE
  Rx state: WAIT FOR FRAME
```

To monitor and maintain LLDP on the system or get information about LLDP neighbors, use one of the following commands:

Command	Description
clear lldp counters	Resets LLDP traffic counters or LLDP neighbor information.
show lldp entry	Displays detailed information about LLDP neighbors.
show lldp errors	Displays LLDP error and overflow statistics.
show lldp neighbors	Displays information about LLDP neighbors.
show lldp traffic	Displays statistics for LLDP traffic.

To collect or clear LLDP interface statistics, you can use the following commands:

Command	Description
show lldp traffic interface <i>interface_name</i>	Displays LLDP traffic statistics for the specified interface.

Command	Description
<code>clear lldp counters interface <i>interface_name</i></code>	Clears LLDP traffic statistics for the specified interface. Global statistics remains intact. Similarly, clearing global statistics does not impact the interface statistics.

Examples for LLDP Interface Statistics

This example shows interface statistics for **gigabitEthernet0/0/0/0**:

```
Router#show lldp traffic interface gigabitEthernet0/0/0/0
```

This example clears the interface statistics for **gigabitEthernet0/0/0/0**.

```
Router#show lldp traffic interface gigabitEthernet0/0/0/0
```

Running Configuration

```
Router#show lldp traffic interface gigabitEthernet 0/2/0/8
Wed Aug 24 17:38:11.829 IST
```

```
LLDP Interface statistics:
  Total frames out: 28786
  Total frames in: 38417
  Total frames received in error: 0
  Total frames out error: 0
  Total frames discarded: 0
  Total TLVs discarded: 0
  Total TLVs unrecognized: 0
```

Configuring LLDP Snoop

If you have LLDP enabled on all Ethernet interfaces, the system enables Link Layer Discovery Protocol (LLDP) snoop on all L2 interfaces by default. You can use LLDP snooping to troubleshoot problems at the client ports.



Note LLDP snoop is enabled only when LLDP RX is enabled and LLDP TX (transmit) is disabled either on interface or global LLDP configuration.

To enable LLDP snoop on an L2 interface, perform the following steps:

```
RP/0/RSP0/CPU0:ios# configure
RP/0/RSP0/CPU0:ios(config)# interface FourHundredGigE 0/0/0/5
RP/0/RSP0/CPU0:ios(config-if)#lldp
RP/0/RSP0/CPU0:ios(config-if)#enable
RP/0/RSP0/CPU0:ios(config-if)#transmit disable
RP/0/RSP0/CPU0:ios(config-if)#commit
```

Running Configuration

```

RP/0/RP0/CPU0:router#show run
Fri Jan 21 17:45:17.529 UTC
Building configuration...
!! IOS XR Configuration 7.7.1.06I
!! Last configuration change at Fri Jan 21 17:20:27 2022 by cisco
!
hostname router1
logging console disable
username xxxx
  group root-lr
  group cisco-support
  secret 10
$6$JELNK0oJaZZN7K0.$8YmyRWkq3D92i.lJc5QsDdkq4kUjU.g9U7sYIIAV1QVnSBemng5q.5EyYv6xSL9niDxRmKaFEATs9BkitDqpr.
!
line console
  exec-timeout 0 0
  absolute-timeout 0
  session-timeout 0
!
line default
  exec-timeout 0 0
  absolute-timeout 0
  session-timeout 0
!
vty-pool default 0 99 line-template default
call-home
  service active
  contact smart-licensing
  profile CiscoTAC-1
    active
  destination transport-method email disable
  destination transport-method http
!
!
interface MgmtEth0/RP0/CPU0/0
  shutdown
!
interface FourHundredGigE0/0/0/0
  lldp
    enable
    transmit disable
!
  l2transport
!
!
interface FourHundredGigE0/0/0/1
  shutdown
!
interface FourHundredGigE0/0/0/2
  shutdown
!
interface FourHundredGigE0/0/0/3
  shutdown
!
interface FourHundredGigE0/0/0/4
  shutdown
!
interface FourHundredGigE0/0/0/5
  lldp
    enable
    transmit disable
!

```

```
l2transport
!
!
interface FourHundredGigE0/0/0/6
 shutdown
!
interface FourHundredGigE0/0/0/7
 shutdown
!
interface FourHundredGigE0/0/0/8
 shutdown
!
interface FourHundredGigE0/0/0/9
 shutdown
!
interface FourHundredGigE0/0/0/10
 shutdown
!
interface FourHundredGigE0/0/0/11
 shutdown
!
interface FourHundredGigE0/0/0/12
 shutdown
!
interface FourHundredGigE0/0/0/13
 shutdown
!
interface FourHundredGigE0/0/0/14
 shutdown
!
interface FourHundredGigE0/0/0/15
 shutdown
!
interface FourHundredGigE0/0/0/16
 shutdown
!
interface FourHundredGigE0/0/0/17
 shutdown
!
interface FourHundredGigE0/0/0/18
 shutdown
!
interface FourHundredGigE0/0/0/19
 shutdown
!
interface FourHundredGigE0/0/0/20
 shutdown
!
interface FourHundredGigE0/0/0/21
 shutdown
!
interface FourHundredGigE0/0/0/22
 shutdown
!
interface FourHundredGigE0/0/0/23
 shutdown
!
interface HundredGigE0/0/0/24
 shutdown
!
interface HundredGigE0/0/0/25
 shutdown
!
interface HundredGigE0/0/0/26
```

```

    shutdown
    !
interface HundredGigE0/0/0/27
  shutdown
  !
interface HundredGigE0/0/0/28
  shutdown
  !
interface HundredGigE0/0/0/29
  shutdown
  !
interface HundredGigE0/0/0/30
  shutdown
  !
interface HundredGigE0/0/0/31
  shutdown
  !
interface HundredGigE0/0/0/32
  shutdown
  !
interface HundredGigE0/0/0/33
  shutdown
  !
interface HundredGigE0/0/0/34
  shutdown
  !
interface HundredGigE0/0/0/35
  shutdown
  !
l2vpn
  bridge group bg1
  bridge-domain bd1
    interface FourHundredGigE0/0/0/0
    !
    interface FourHundredGigE0/0/0/5
    !
  !
  !
end

RP/0/RP0/CPU0:router#

```

Verification

```

router0 <---> router1 <---> router2
          0/0/0/0          0/0/0/0/5

```

```

RP/0/RP0/CPU0:router0#config
Fri Jan 21 17:16:41.713 UTC
RP/0/RP0/CPU0:router0(config)#lldp
RP/0/RP0/CPU0:router0(config-lldp)#exit
RP/0/RP0/CPU0:router0(config)#int hu 0/0/0/0
RP/0/RP0/CPU0:router0(config-if)#no shut
RP/0/RP0/CPU0:router0(config-if)#end
Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:yes

```

```

RP/0/RP0/CPU0:router1#config
Fri Jan 21 17:17:41.459 UTC
RP/0/RP0/CPU0:router1(config)#int FourHundredGigE 0/0/0/0
RP/0/RP0/CPU0:router1(config-if)#no shut
RP/0/RP0/CPU0:router1(config-if)#l2transport
RP/0/RP0/CPU0:router1(config-if-l2)#exit
RP/0/RP0/CPU0:router1(config-if)#lldp

```

```

RP/0/RP0/CPU0:router1(config-lldp)#enable
RP/0/RP0/CPU0:router1(config-lldp)#transmit disable
RP/0/RP0/CPU0:router1(config-lldp)#exit
RP/0/RP0/CPU0:router1(config-if)#exit
RP/0/RP0/CPU0:router1(config)#int FourHundredGigE 0/0/0/5
RP/0/RP0/CPU0:router1(config-if)#no shut
RP/0/RP0/CPU0:router1(config-if)#l2transport
RP/0/RP0/CPU0:router1(config-if-l2)#exit
RP/0/RP0/CPU0:router1(config-if)#lldp
RP/0/RP0/CPU0:router1(config-lldp)#enable
RP/0/RP0/CPU0:router1(config-lldp)#transmit disable
RP/0/RP0/CPU0:router1(config-lldp)#exit
RP/0/RP0/CPU0:router1(config-if)#exit
RP/0/RP0/CPU0:router1(config)#l2vpn bridge group bg1
RP/0/RP0/CPU0:router1(config-l2vpn-bg)#bridge-domain bd1
RP/0/RP0/CPU0:router1(config-l2vpn-bg-bd)#interface FourHundredGigE 0/0/0/0
RP/0/RP0/CPU0:router1(config-l2vpn-bg-bd-ac)#exit
RP/0/RP0/CPU0:router1(config-l2vpn-bg-bd)#interface FourHundredGigE 0/0/0/5
RP/0/RP0/CPU0:router1(config-l2vpn-bg-bd-ac)#end
Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:yes

```

```

RP/0/RP0/CPU0:router0#config
Fri Jan 21 17:16:41.713 UTC
RP/0/RP0/CPU0:router0(config)#lldp
RP/0/RP0/CPU0:router0(config-lldp)#exit
RP/0/RP0/CPU0:router0(config)#int hu 0/0/0/0
RP/0/RP0/CPU0:router0(config-if)#no shut
RP/0/RP0/CPU0:router0(config-if)#end
Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:yes

```

```

RP/0/RP0/CPU0:router0#sh lldp neighbors
Fri Jan 21 17:21:15.857 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
router2	HundredGigE0/0/0/0	120	R	
FourHundredGigE0/0/0/5				

Total entries displayed: 1

```
RP/0/RP0/CPU0:router0#
```

```

RP/0/RP0/CPU0:router0#sh lldp neighbors
Fri Jan 21 17:21:15.857 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
router2	HundredGigE0/0/0/0	120	R	
FourHundredGigE0/0/0/5				

Total entries displayed: 1

```
RP/0/RP0/CPU0:router0#
```

```

RP/0/RP0/CPU0:router2#sh lldp neighbors
Fri Jan 21 17:21:20.998 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
router0            FourHundredGigE0/0/0/5  120       R           
```

Total entries displayed: 1

RP/0/RP0/CPU0:router2#

```
RP/0/RP0/CPU0:router1#show controllers npu stats traps-all instance all location all | inc
LLDP
Fri Jan 21 17:24:07.964 UTC
LLDP
3975      IFG      1520     0         0         0         0         0         0         0
LLDP_SNOOP
3862      NPU      N/A      16        0         0         0         0         0         0
RP/0/RP0/CPU0:router1#
```

Configuration Examples for Ethernet

This section provides the following configuration examples:

Configuring an Ethernet Interface: Example

The following example shows how to configure an interface for a 10-Gigabit Ethernet modular services card:

```
RP/0//CPU0:router# configure
RP/0//CPU0:router(config)# interface TenGigE 0/0/0/1
RP/0//CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224
RP/0//CPU0:router(config-if)# flow-control ingress
RP/0//CPU0:router(config-if)# mtu 1448
RP/0//CPU0:router(config-if)# mac-address 0001.2468.ABCD
RP/0//CPU0:router(config-if)# no shutdown
RP/0//CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes
```

```
RP/0//CPU0:router# show interfaces TenGigE 0/0/0/1
```

```
TenGigE0/0/0/1 is down, line protocol is down
Hardware is TenGigE, address is 0001.2468.abcd (bia 0001.81a1.6b23)
Internet address is 172.18.189.38/27
MTU 1448 bytes, BW 10000000 Kbit
    reliability 0/255, txload Unknown, rxload Unknown
Encapsulation ARPA,
    Full-duplex, 10000Mb/s, LR
    output flow control is on, input flow control is on
Encapsulation ARPA,
ARP type ARPA, ARP timeout 01:00:00
Last clearing of "show interface" counters never
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
    0 packets input, 0 bytes, 0 total input drops
    0 drops for unrecognized upper-level protocol
    Received 0 broadcast packets, 0 multicast packets
    0 runts, 0 giants, 0 throttles, 0 parity
```

```

0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
0 packets output, 0 bytes, 0 total output drops
Output 0 broadcast packets, 0 multicast packets
0 output errors, 0 underruns, 0 applique, 0 resets
0 output buffer failures, 0 output buffers swapped out
0 carrier transitions

```

Configuring LLDP: Examples

The following example shows how to enable LLDP globally on the router and modify the default LLDP operational characteristics:

```

RP/0//CPU0:router# configure
RP/0//CPU0:router(config)# lldp
RP/0//CPU0:router(config)# lldp holdtime 60
RP/0//CPU0:router(config)# lldp reinit 4
RP/0//CPU0:router(config)# lldp timer 60
RP/0//CPU0:router(config)# commit

```

The following example shows how to disable a specific Gigabit Ethernet interface for LLDP transmission:

```

RP/0//CPU0:router# configure
RP/0//CPU0:router(config)# interface HundredGigE 0/2/0/0
RP/0//CPU0:router(config-if)# lldp
RP/0//CPU0:router(config-lldp)# transmit disable

```

Where to Go Next

When you have configured an Ethernet interface, you can configure individual VLAN subinterfaces on that Ethernet interface.

For information about modifying Ethernet management interfaces for the shelf controller (SC), route processor (RP), and distributed RP, see the Advanced Configuration and Modification of the Management Ethernet Interface later in this document.

For information about IPv6 see the Implementing Access Lists and Prefix Lists on Cisco IOS XR Software module in the Cisco IOS XR IP Addresses and Services Configuration Guide.

Configuring a Layer 2 VPN AC: Example

The following example indicates how to configure a Layer 2 VPN AC on an Ethernet interface:

```

RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface TenGigE 0/0/0/2
RP/0/RSP0/CPU0:router(config-if)# l2transport
RP/0/RSP0/CPU0:router(config-if-l2)# l2protocol tunnel
RP/0/RSP0/CPU0:router(config-if-l2)# commit

```

Configuring Physical Ethernet Interfaces

Use this procedure to create a basic Ethernet interface configuration.

Procedure

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 [TenGigE FortyGigE HundredGigE FourHundredGigE] 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 **show interfaces [TenGigE FortyGigE HundredGigE FourHundredGigE] 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:

- 10GigE
- 40GigE
- 100GigE

Note

- The example indicates a 100-Gigabit Ethernet interface in the line card in slot 1.
- 400GigE

The examples of *interface-path-id* ranges are:

- **TenGigE** — 0/0/0/0 - 0/0/0/31
- **FortyGigE** — 0/0/1/0 - 0/0/1/1
- **HundredGigE** — 0/0/1/0 - 0/0/1/1

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 **flow-control** {**bidirectional**| **egress** | **ingress**}**Example:**

```
RP/0/RP0/CPU0:router(config-if)# flow control ingress
```

(Optional) Enables the sending and processing of flow control pause frames.

- **egress**—Enables the sending of flow control pause frames in egress.
- **ingress**—Enables the processing of received pause frames on ingress.
- **bidirectional**—Enables the sending of flow control pause frames in egress and the processing of received pause frames on ingress.

Step 7 **mtu** *bytes***Example:**

```
RP/0/RP0/CPU0:router(config-if)# mtu 1448
```

(Optional) Sets the MTU value for the interface.

- The default is 1514 bytes for normal frames and 1518 bytes for 802.1Q tagged frames.
- The range for 100-Gigabit Ethernet mtu values is 64 bytes to 65535 bytes.

Step 8 **no shutdown****Example:**

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

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

Step 9 **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 10 `show interfaces [TenGigE FortyGigE HundredGigE FourHundredGigE] 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 1448

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
!

```

Viewing Interface Counters Report

The Interface Counters report summarizes the statistics for all interfaces configured on the router.

The report displays the interfaces configured, the input and output rate, the total number of packets transmitted and received, the time interval, the current status of each interface, and the packet counts for input and output broadcast, multicast, and errored packets.

The **show interfaces** command, displays statistics per interface with many lines of data. The traffic rate displays the average number of packets received per second over the load interval. The load interval is configurable on the physical and bundle main interface. The report displays the load on the interface for a longer duration of time and does not show a spike in the traffic rate. This rate is the exponentially weighted average with a time constant of the load interval.



Note For the average to be within two percent of the instantaneous rate of a uniform stream of traffic, four times the load interval must pass.

For more information about the use of **show interfaces** command, see *Interface and Hardware Component Command Reference for Cisco 8000 Series Routers*.

Instant Display of Traffic Rates for all the Physical Interfaces

Table 16: Feature History Table

Feature Name	Release Information	Feature Description
Instant display of traffic rates on all physical interfaces	Release 7.5.4	<p>You can now display a snapshot of the traffic throughput and traffic rate on all physical interfaces over the last few seconds. We have introduced a show command to view the counters and rate information for the interfaces.</p> <p>The feature introduces these:</p> <ul style="list-style-type: none"> • CLI: show interfaces counter rates physical • YANG Data Model: Cisco-IOS-XR-infra-statsd-oper.yang Cisco-IOS-XR-infra-statsd-oper.yang (see GitHub under the 754 folder.)

The new **show** command displays a snapshot of statistics for all the interfaces at a given instant for your quick reference. Here, the display is in a tabular format for easy analysis.

Run the **show interfaces counter rates physical** command to view statistics of all physical interfaces.

View the statistics

```
Router#show interfaces counters rates physical
```

InterfaceName	Intvl	InMbps	InBW%	InKpps	OutMbps	OutBW%	OutKpps
GigabitEthernet0/2/0/0	0:05	0.0	0.0%	0.0	0.0	0.0%	0.0
GigabitEthernet0/2/0/1	0:05	0.0	0.0%	0.0	0.0	0.0%	0.0
GigabitEthernet0/2/0/2	0:05	0.0	0.0%	0.0	0.0	0.0%	0.0
GigabitEthernet0/2/0/3	0:05	235.0	22.0%	23.5	87.0	9.5%	7.2
GigabitEthernet0/3/0/0	0:05	88.0	9.3%	7.0	100.0	10.0%	10.5
GigabitEthernet0/3/0/1	0:05	0.0	0.0%	0.0	0.0	0.0%	0.0

The statistics for each physical interface is calculated for the time interval of 5 sec. Hence, the input and output rate (in Mbps and Kpps) is the real-time statistics.



Note The traffic rate displayed is the real-time link utilization of the time interval. The time interval is determined by the system and may vary based on the system processing load. The time interval increases during events where the system is handling, for example, performing routing updates.

Display of traffic rates for bundle interfaces

Table 17: Feature History Table

Feature name	Release information	Description
Display of traffic rates for bundle interfaces	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8712-MOD-M • 8011-4G24Y4H-I
Display of traffic rates for bundle interfaces	Release 24.4.1	<p>Routers can now display a snapshot of the traffic throughput and traffic rate on all bundle interfaces over the last few seconds, facilitating easy analysis. These statistics are presented in a tabular format for quick reference.</p> <p>The feature introduces these changes:</p> <p>CLI:</p> <ul style="list-style-type: none"> • show interfaces counters rates bundle <p>YANG Data Models:</p> <ul style="list-style-type: none"> • The existing Xpath <code>interface-rate</code> in <code>Cisco-IOS-XR-infra-statsd-oper.yang</code> is extended to retrieve bundle interface statistics. <p>(see GitHub, YANG Data Models Navigator)</p>

The **show interfaces counters rates bundle** command displays a snapshot of statistics for all the bundle interfaces at a given instant for your quick reference. The statistics display is in a tabular format facilitating easy analysis.

Guidelines and restrictions for display of traffic rates for bundle interfaces

The displayed traffic rate represents the real-time link utilization for a given time interval. This interval is system-determined and may vary depending on system processing load. During events such as routing updates, the time interval may increase.

View bundle interface statistics

Run the **show interfaces counter rates bundle** command to view the statistics of all bundle interfaces.

```

Router#show interfaces counters rates bundle
Wed Aug 14 19:54:04.842 EDT
InterfaceName          Intval      InMbps      InBW%       InKpps      OutMbps     OutBW%
OutKpps
Bundle-Ether111        0:03        38454.9     40.0%       8064.8      0.0         0.0%
0.0
Bundle-Ether112        0:03         0.0         0.0%        0.0         0.0         0.0%
0.0
Bundle-Ether1201       0:03         0.0         0.0%        0.0         9741.6      3.4%
2002.2
Bundle-Ether1202       0:03         0.0         0.0%        0.0         9703.0      5.0%
1995.4
Bundle-Ether1203       0:03         0.0         0.0%        0.0         9841.9      5.1%
2023.0
Bundle-Ether1204       0:03         0.0         0.0%        0.0         10011.3     10.4%
2058.5
Bundle-Ether1205       0:03         0.0         0.0%        0.0         0.0         0.0%
0.0
Bundle-Ether1206       0:03         0.0         0.0%        0.0         0.0         0.0%
0.0
Bundle-Ether1207       0:03         0.0         0.0%        0.0         0.0         0.0%
0.0
Bundle-Ether1208       0:03         0.0         0.0%        0.0         0.0         0.0%
0.0
Bundle-Ether1209       0:03         0.0         0.0%        0.0         0.0         0.0%
0.0
Bundle-Ether1210       0:03         0.0         0.0%        0.0         0.0         0.0%
0.0
Bundle-Ether1301       0:03        38698.3     13.4%       8062.7      0.0         0.0%
0.0
Bundle-Ether1302       0:03         0.0         0.0%        0.0         0.0         0.0%
0.0
Bundle-Ether1303       0:03         0.0         0.0%        0.0         38741.7     28.8%
8070.8

```

How to Configure Interfaces in Breakout Mode

Information About Breakout

The router supports transmission of traffic in the breakout mode. The breakout mode enables a 40 Gigabit Ethernet port to be split into four independent and logical 10 Gigabit Ethernet ports. The 4x10 breakout mode is supported on the following types of 40G modules:

- QSFP-4x10-LR-S
- QSFP-40G-SR4

Guidelines and Restrictions for Breakout Mode

- If you're using a Q100-based Cisco 8200 Series Router and want to set up a 4x10G breakout configuration, you need to use even numbered ports from 24 to 35. These include ports 24, 26, 28, 30, 32, and 34. Once you do this, the system automatically disables the odd numbered ports in this range - ports 25, 27, 29, 31, 33, and 35.
- Use the *hw-module port-range* command to set the port range for the breakout configuration in the global configuration.

- To remove the global *hw-module port-range* configuration, you must first remove the 'breakout 4x10' configuration under the controller.
- On the Cisco 8812 router, the 88-LC0-34H14FH line card can have only three ports per Interface Group (IFG) when operating in 4x10 GbE or 4x25 GbE breakout configuration mode. This helps prevent the QOS-DPA_QOSEA-2-TMPORT_PROG_ERROR issue, which creates partial interfaces during configuration mode.

Configure Breakout in a Port

Configuring breakout in a port:

```
RP/0/RP0/CPU0:uut# configure
Fri Oct 11 23:58:47.165 UTC
RP/0/RP0/CPU0:uut(config)# controller optics 0/1/0/28
RP/0/RP0/CPU0:uut(config-Optics)# breakout 4x10
RP/0/RP0/CPU0:uut(config-Optics)# commit
Fri Oct 11 23:59:51.261 UTC
RP/0/RP0/CPU0:uut(config-Optics)# end
RP/0/RP0/CPU0:uut#
```

Remove the Breakout Configuration

Removing the breakout configuration:

```
RP/0/RP0/CPU0:uut# configure
Sat Oct 12 00:01:38.673 UTC
RP/0/RP0/CPU0:uut(config)# controller optics 0/1/0/28
RP/0/RP0/CPU0:uut(config-Optics)# no breakout 4x10
RP/0/RP0/CPU0:uut(config-Optics)# commit
Sat Oct 12 00:01:55.864 UTC
RP/0/RP0/CPU0:uut(config-Optics)# end
```

Verify a Breakout Configuration

Verifying a breakout configuration:

```
RP/0/RP0/CPU0:uut# show running-config controller optics 0/1/0/28
Sat Oct 12 00:11:33.962 UTC
controller Optics0/1/0/28
breakout 4x10
!
```

```
RP/0/RP0/CPU0:uut# show int br location 0/1/CPU0 | i Te
Sat Oct 12 00:11:38.609 UTC
      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
```

Ethernet Interface Route Statistics

Table 18: Feature History Table

Feature Name	Release Information	Feature Description
Ethernet Interface Route Statistics	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8712-MOD-M • 8011-4G24Y4H-I
Ethernet Interface Route Statistics	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-12TH24FH-E • 88-LC1-36EH • 88-LC1-52Y8H-EM
Ethernet Interface Route Statistics	Release 7.3.4	<p>You can view statistics on the number of packets and bytes sent and received in unicast, multicast, and broadcast routes.</p> <p>These statistics help you to monitor the network performance and measure your bandwidth.</p>

Ethernet interface route statistics provide the following information about the unicast, multicast, and broadcast routes:

- The number of packets or bytes received and transmitted.
- The total number of packets or bytes passing through the Ethernet interface.

These statistics are available on Cisco 8000 routers that are built with Cisco Silicon One Q100 and Q200 processors and all Network Processor Unit (NPU) 2.0 devices.

Ethernet interface route statistics may be useful for monitoring the network devices and their traffic. For example, if you are not able to connect to the internet or use some cloud-based applications, these route statistics can help you understand the problems in the network and where they occur.

Viewing the Interface Statistics

Use the `show interface` and the `show controller interface` commands to view these Ethernet interface route statistics. The following is a sample showing both commands.

```
Router#show interfaces HundredGigE 0/0/0/0
```

```
<Timestamp>
```

```
HundredGigE0/0/0/0 is up, line protocol is up
```

```
Interface state transitions: 93
```

```
Hardware is HundredGigE, address is acbc.d975.0500 (bia acbc.d975.0500)
```

```
Internet address is 100.0.1.1/24
```

```
MTU 1514 bytes, BW 100000000 Kbit (Max: 100000000 Kbit)
```

```
reliability 255/255, txload 0/255, rxload 0/255
```

```
Encapsulation ARPA,
```

```
Full-duplex, 100000Mb/s, 100GBASE-SR4, 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 01:03:01
```

```
ARP type ARPA, ARP timeout 04:00:00
```

```
Last input 3d09h, output 01:02:41
```

```
Last clearing of "show interface" counters never
```

```
5 minute input rate 0 bits/sec, 0 packets/sec
```

```
5 minute output rate 0 bits/sec, 0 packets/sec
```

```

2959131434 packets input, 757537646912 bytes, 0 total input drops

0 drops for unrecognized upper-level protocol

Received 0 broadcast packets, 0 multicast packets

    0 runts, 0 giants, 0 throttles, 0 parity

0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort

2958525230 packets output, 757382468319 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

93 carrier transitions

```

```
Router#show controllers HundredGigE0/0/0/0 stats
```

```
<Timestamp>
```

```
Statistics for interface HundredGigE0/0/0/0 (cached values):
```

```
Ingress:
```

```

Input total bytes           = 757537646912

Input good bytes             = 757537646912

Input total packets        = 2959131434
Input 802.1Q frames          = 0
Input pause frames           = 0
Input pkts 64 bytes          = 1
Input pkts 65-127 bytes      = 0
Input pkts 128-255 bytes     = 0
Input pkts 256-511 bytes     = 2959131433
Input pkts 512-1023 bytes    = 0
Input pkts 1024-1518 bytes   = 0
Input pkts 1519-Max bytes    = 0

Input good pkts              = 2959131434

```

```

Input unicast pkts           = 0
Input multicast pkts        = 0
Input broadcast pkts        = 0

Input drop overrun            = 0
Input drop abort              = 0
Input drop invalid VLAN      = 0
Input drop invalid DMAC      = 0
Input drop invalid encap     = 0
Input drop other              = 0

Input error giant             = 0
Input error runt              = 0
Input error jabbers           = 0
Input error fragments         = 0
Input error CRC               = 0
Input error collisions        = 0
Input error symbol            = 0
Input error other             = 0

Input MIB giant               = 0
Input MIB jabber              = 0
Input MIB CRC                  = 0

```

Egress:

```

Output total bytes           = 757382468319
Output good bytes             = 757382468319

Output total packets        = 2958525230
Output 802.1Q frames          = 0
Output pause frames           = 0
Output pkts 64 bytes          = 41
Output pkts 65-127 bytes     = 296
Output pkts 128-255 bytes    = 746
Output pkts 256-511 bytes    = 2958524147
Output pkts 512-1023 bytes   = 0
Output pkts 1024-1518 bytes  = 0
Output pkts 1519-Max bytes   = 0

Output good pkts              = 2958525230
Output unicast pkts          = 0
Output multicast pkts       = 0
Output broadcast pkts       = 0

Output drop underrun         = 0
Output drop abort            = 0
Output drop other            = 0

Output error other            = 0

```




CHAPTER 6

Configuring Ethernet OAM

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

Table 19: Feature Information Table

Release	Modification
Release 7.3.1	Support for Ethernet Link OAM was introduced.

- [Information About Configuring Ethernet OAM, on page 89](#)
- [Configuration Examples for Ethernet OAM, on page 93](#)
- [Ethernet CFM, on page 96](#)
- [Unidirectional link detection protocol, on page 116](#)
- [How to Configure Ethernet OAM, on page 121](#)
- [CFM Over Bundles, on page 143](#)
- [Ethernet SLA Statistics Measurement in a Profile, on page 144](#)
- [Ethernet frame delay measurement for L2VPN services, on page 148](#)
- [Minimum delay bin, on page 153](#)
- [Link loss forwarding, on page 156](#)

Information About Configuring Ethernet OAM

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

Ethernet Link OAM

Table 20: Feature History Table

Feature Name	Release Information	Feature Description
Ethernet Link OAM on Physical Interface— (802.3ah) Link Monitoring	Release 25.4.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100], 8700 [ASIC: K100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8011-12G12X4Y-A • 8011-12G12X4Y-D • 8711-48Z-M
Ethernet Link OAM support	Release 25.2.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC:K100]) (select variants only*)</p> <p>This release supports:</p> <ul style="list-style-type: none"> • Ethernet Link OAM on physical interface— (802.3ah) link monitoring. <p>*This feature is now supported on Cisco 8712-MOD-M routers.</p>
Ethernet Link OAM on Physical Interface— (802.3ah) Link Monitoring	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is now supported on Cisco 8011-4G24Y4H-I routers.</p>

Feature Name	Release Information	Feature Description
Ethernet Link OAM on Physical Interface— (802.3ah) Link Monitoring	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100, K100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>Ethernet link OAM operates on a single, physical link and it can be configured to monitor either side or both sides of that link.</p> <p>Ethernet OAM supports link monitoring:</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 8712-MOD-M • 88-LC1-12TH24FH-E • 88-LC1-36EH • 88-LC1-52Y8H-EM
Ethernet Link OAM	Release 7.3.1	<p>This feature allows service providers to monitor the quality of the connections on a MAN or WAN. Service providers can monitor specific events, and take actions on 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.</p>

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, and take actions on 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.

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 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.

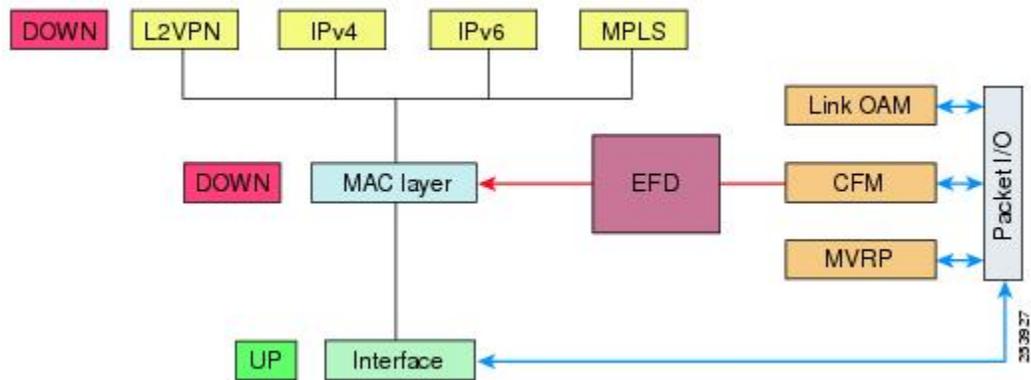
EFD changes this to allow EOAM to act as the line protocol for Ethernet interfaces. This allows EOAM to control the interface state so that if a EOAM defect (such as AIS or loss of continuity) is detected with an expected peer MEP, the interface can be shut down. This not only stops traffic flow, but 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 EOAM frames continue to flow. This allows EOAM to detect when the problem has been resolved, and thus bring the interface backup automatically.

This figure shows EOAM 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 EOAM detects there is no longer any error, it can signal to EFD and all protocols will once again go active.

Figure 1: EOAM 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.

Link Monitoring

Link monitoring enables an OAM peer to monitor faults that cause the quality of a link to deteriorate over time. When link monitoring is enabled, an OAM peer can be configured to take action when the configured thresholds are exceeded.

Remote Loopback

Remote loopback enables one side of a link to put the remote side of the link into loopback mode for testing. When remote loopback is enabled, all packets initiated by the primary side of the link are looped back to the primary side, unaltered by the remote side. In remote loopback mode, the remote side is not allowed to inject any data into the packets.

Configuration Examples for Ethernet OAM

This section provides the following configuration examples:

Configuring Ethernet OAM Features on an Individual Interface: Example

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

```
configure
interface TenGigE 0/1/0/0
  ethernet oam
  connection timeout 30
  mib-retrieval
  link-monitor
    frame window milliseconds 60000
    frame threshold low 10000000 high 60000000
    frame-period window milliseconds 60000
    frame-period threshold ppm low 100 high 120000
    frame-seconds window milliseconds 900000
    frame-seconds threshold low 3 high 900
    symbol-period window milliseconds 60000
    symbol-period threshold ppm low 1000000 high 1000000
  exit
  require-remote
  mode active
  mib-retrieval
  exit
  action
    critical-event error-disable-interface
    dying-gasp error-disable-interface
    capabilities-conflict error-disable-interface
    wiring-conflict error-disable-interface
    discovery-timeout error-disable-interface
    session-down error-disable-interface
  commit
```

Configuring an Ethernet OAM Profile Globally: Example

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

```
configure
ethernet oam profile Profile_1
  connection timeout 30
  mib-retrieval
  link-monitor
    frame window milliseconds 60000
    frame threshold low 10000000 high 60000000
    frame-period window milliseconds 60000
    frame-period threshold ppm low 100 high 1000000
    frame-seconds window milliseconds 900000
    frame-seconds threshold low 3 high 900
    symbol-period window milliseconds 60000
    symbol-period threshold ppm low 100000 high 1000000
  exit
  require-remote
  mode active
  mib-retrieval
  exit
  action
    critical-event error-disable-interface
    dying-gasp error-disable-interface
    capabilities-conflict error-disable-interface
    wiring-conflict error-disable-interface
    discovery-timeout error-disable-interface
```

```
session-down 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
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

commit

configure
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

commit
```

Recovering from error-disable: Example

You can recover an error-disabled interface due to session-down using one of these methods:

- Manually clear the error-disable using the **clear** command.

```
Router# configure
Router(config)# ethernet oam profile Profile_1
Router(config-eoam)# action
Router(config-eoam-action)# clear session-down error-disable-interface
```

- Disable and then re-enable the network link using administrative shutdown commands to reset the connection.

```
Router# configure
Router(config)# interface TenGigE 0/1/0/0
Router(config-if)# shutdown
Router(config-if)# commit
Router(config-if)# no shutdown
Router(config-if)# commit
```

- Configure an auto-recovery timer for this error-disable reason.

```
Router# configure
Router(config)# error-disable recovery cause link-oam-session-down interval 30
Router(config)# 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
snmp-server traps ethernet oam events
```

Ethernet CFM

Table 21: Feature History Table

Feature name	Release	Description
Y.1731 support on CFM	Release 25.4.1	Introduced in this release on: Fixed Systems (8700 [ASIC: K100])(select variants only*) *This feature is supported on Cisco 8711-48Z-M routers.
Y.1731 support on CFM	Release 25.3.1	Introduced in this release on: Fixed Systems (8010 [ASIC: A100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*) This feature is now supported on: <ul style="list-style-type: none"> • 8011-4G24Y4H-I • 8711-32FH-M • 88-LC1-52Y8H-EM • 88-LC1-12TH24FH-E

Feature name	Release	Description
Increase in number of CFM sessions	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])</p> <p>This feature is now supported on:</p> <ul style="list-style-type: none"> • 8712-MOD-M • 8011-4G24Y4H-I
Increase in number of CFM sessions	Release 24.4.1	<p>Introduced in this release on: Fixed Systems(8200, 8700); Centralized Systems (8600); Modular Systems (8800 [LC ASIC: Q100, Q200, P100])</p> <p>The number of supported Connectivity Fault Management (CFM) sessions is now increased to 500. This enhancement improves fault detection, network visibility, scalability, and troubleshooting, which are crucial for managing high-performance networks.</p>
CFM on bundle member link for connectivity check	Release 25.4.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100], 8700 [ASIC: K100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8011-12G12X4Y-A • 8011-12G12X4Y-D • 8711-48Z-M
CFM on bundle member link for connectivity check	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100])</p> <p>This feature is now supported on Cisco 8011-4G24Y4H-I routers.</p>
CFM on bundle member link for connectivity check	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100]) (select variants only*)</p> <p>This feature introduces support for Connectivity Fault Management (CFM) on bundle members.</p> <p>* This feature is supported on Cisco 8712-MOD-M routers.</p>

Feature name	Release	Description
Up MEP and down MEP support in CFM	Release 25.4.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100], 8700 [ASIC: K100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8011-12G12X4Y-A • 8011-12G12X4Y-D • 8711-48Z-M
Up MEP and down MEP support in CFM	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100, K100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>This feature introduces Maintenance End Points (MEP) entities that you can configure in a domain.</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 8712-MOD-M • 88-LC1-12TH24FH-E • 88-LC1-36EH • 88-LC1-52Y8H-EM
Monitoring Layer 3 connectivity using down MEP on L3 interfaces	Release 25.4.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8011-12G12X4Y-A • 8011-12G12X4Y-D
Monitoring Layer 3 connectivity using down MEP on L3 interfaces	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100]) (select variants only*)</p> <p>*This feature is now supported on Cisco 8011-4G24Y4H-I routers.</p>

Feature name	Release	Description
Monitoring Layer 3 connectivity using down MEP on L3 interfaces	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100, K100]) (select variants only*); Modular Systems (8800 [LC ASIC: P100])</p> <p>This enhancement expands network diagnostics to L3 interfaces at L2 network termination, simplifying the management and maintenance of multilayer networks.</p> <p>* This feature is supported on Cisco 8712-MOD-M routers.</p>
Monitoring Layer 3 connectivity using down MEP on L3 interfaces	Release 24.2.11	<p>This enhancement expands network diagnostics to L3 interfaces at L2 network termination, simplifying the management and maintenance of multilayer networks. Without impacting the underlying L2 infrastructure, this feature uses CFM packets to verify the connection of L3 paths.</p> <p>Previously, CFM Down MEP support was limited to L2 interfaces associated with cross-connect or bundle members.</p> <p>This feature is supported on both physical main and subinterfaces, bundle main and subinterfaces.</p>

Feature name	Release	Description
CFM on bundle member link for connectivity check	Release 7.3.15	<p>This feature introduces support for Connectivity Fault Management (CFM) on bundle members. Earlier, network administrators managed networks by using the fault, configuration, account, performance, security model. CFM is one of a suite of the Ethernet OAM protocols, which uses a combination of keepalive packets and MAC-based pings, and traceroutes to detect faults in a network.</p> <p>With the CFM feature, you:</p> <ul style="list-style-type: none"> • reduce operating expenses for service operators by reducing network faults and errors • provide end-to-end maintenance of networks
Up MEP and down MEP support in CFM	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])</p> <p>This feature is now supported on:</p> <ul style="list-style-type: none"> • 8712-MOD-M • 8011-4G24Y4H-I
Up MEP and down MEP support in CFM	Release 7.3.15	<p>This feature introduces Maintenance End Points (MEP) entities that you can configure in a domain.</p> <p>MEPs send either CFM frames from the interface where they are configured or CFM frames that are received on other interfaces.</p> <p>MEPs allow you to perform fault management and carry out performance checks.</p>

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.

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—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.

Limitations and restrictions

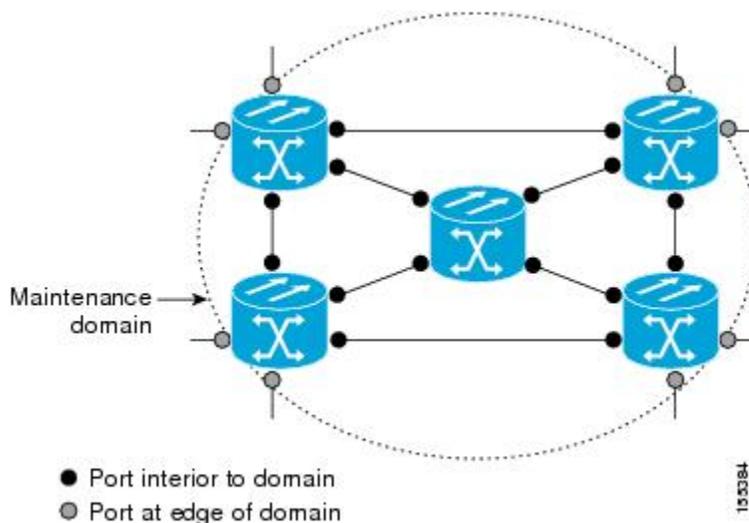
- The system supports only cross-connect.
- MIPs are not supported.
- Supports timer of 1s, 10s, 1m, 10m.
- Supports timer of 100ms, 1s, 10s, 1m, 10m for bundle members.
- Multiple MEPs of different directions are not supported on the same interface or Xconnect.
- CFM is not supported on L2 subinterfaces with default encapsulation.
- When configuring CFM down MEP on an interface, ensure that the interface is included in an L2VPN.

Maintenance Domains

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

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 2: 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.

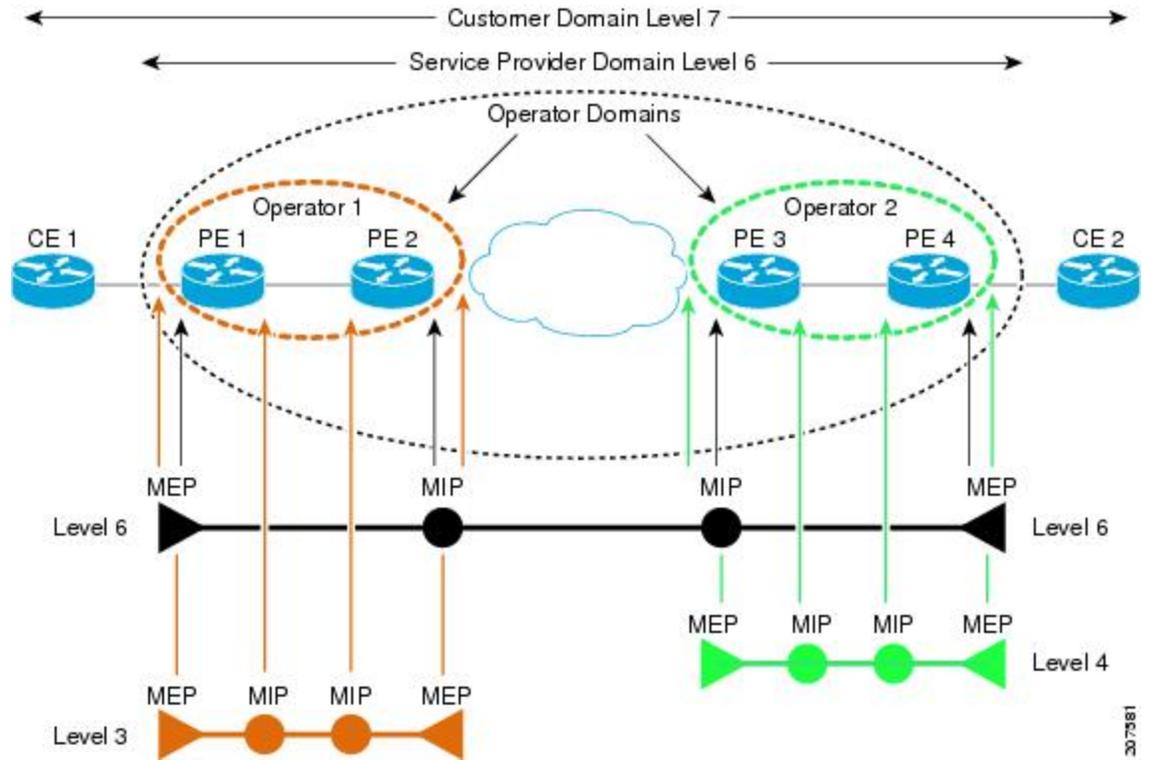
Each organization uses a different CFM maintenance domain.

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



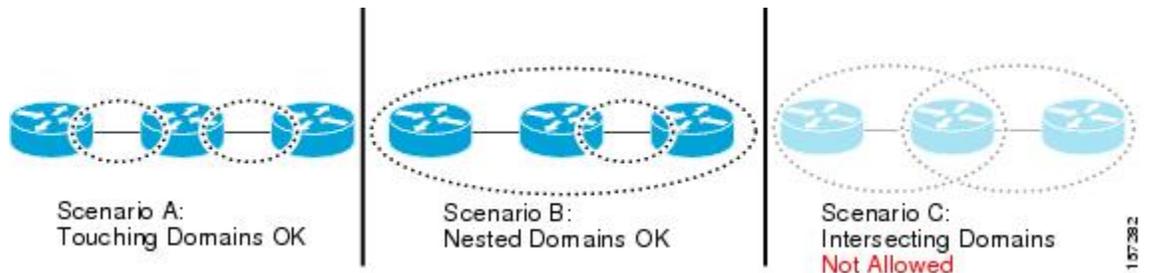
Note In CFM diagrams, the conventions are that triangles represent MEPs, pointing in the direction that the MEP sends CFM frames, and circles represent MIPs.

Figure 3: 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.



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 following type(s) of MP(s):

- 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.

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,

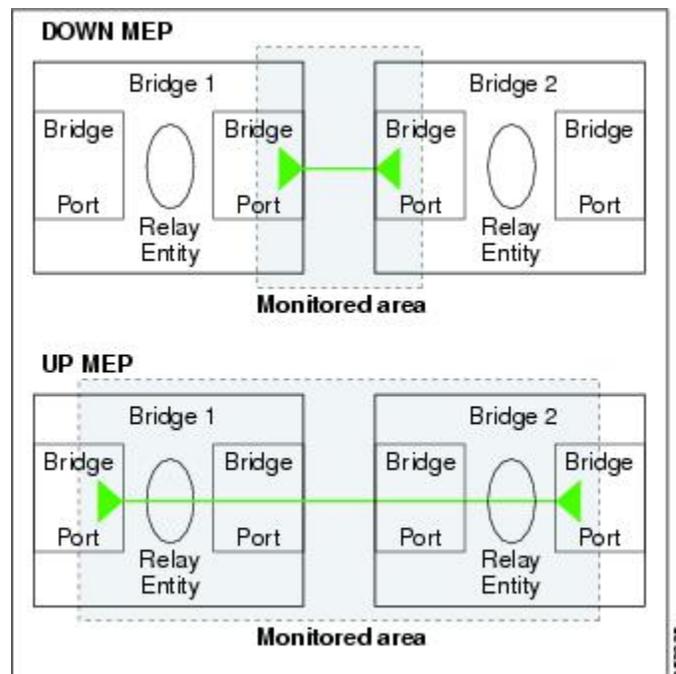
AIS packets are only sent when there is a MIP configured on the same interface as the MEP and at the level of the MIP.



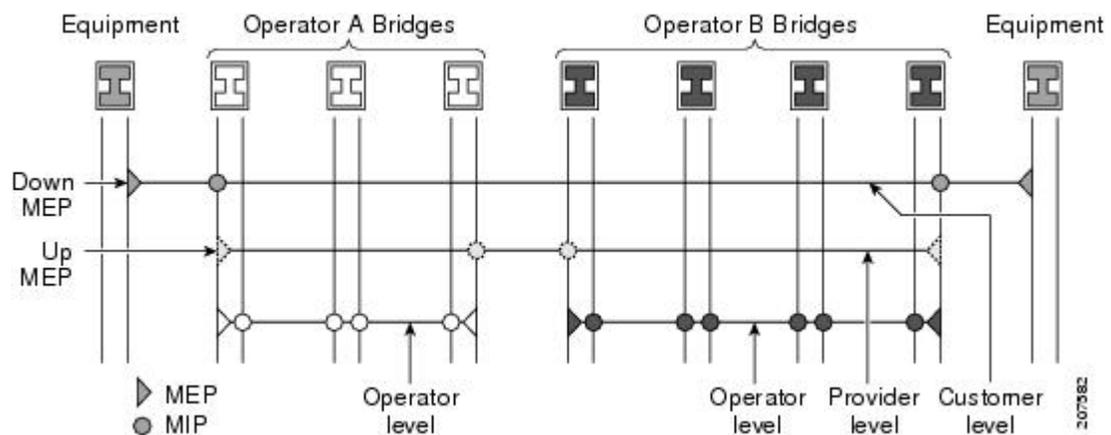
- Note**
- The terms *Down MEP* and *Up MEP* are defined in the IEEE 802.1ag and ITU-T Y.1731 standards, and refer to the direction that CFM frames are sent from the MEP. The terms should not be confused with the operational status of the MEP.
 - The router only supports the “Down MEP level < Up MEP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

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

Figure 4: Monitored Areas for Down and Up MEPs



This figure shows maintenance points at different levels. Because domains are allowed to nest but not intersect, a MEP at a low level often corresponds with a MEP at a higher level.



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.



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)

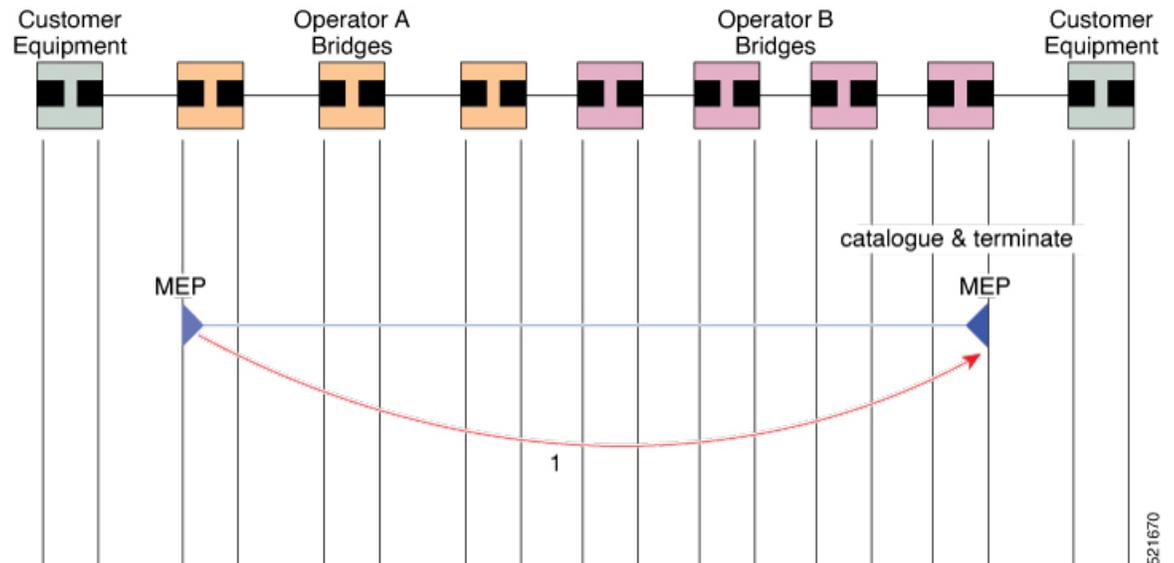
Continuity Check Messages (CCMs) are “heartbeat” messages exchanged periodically between all the MEPs in a service. Each MEP sends out multicast CCMs, and receives CCMs from all the other MEPs in the service—these are referred to as *peer MEPs*. This allows each MEP to discover its peer MEPs, and to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

Table 22: Feature History Table

Feature Name	Release Information	Feature Description
Continuity Check (IEEE 802.1ag and ITU-T Y.1731)	Release 25.3.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>This feature is now supported on:</p> <ul style="list-style-type: none"> • 8011-4G24Y4H-I • 8711-32FH-M • 88-LC1-52Y8H-EM • 88-LC1-12TH24FH-E
Continuity Check (IEEE 802.1ag and ITU-T Y.1731)	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100])</p> <p>Maintenance Endpoints (MEPs) capable of supporting Ethernet OAM send Continuity Check Messages (CCMs) at regular intervals to verify connectivity. If one of them stops receiving CCMs from another endpoint, it infers that there is a connectivity issue or fault in the path.</p> <p>This feature is now supported on:</p> <ul style="list-style-type: none"> • 8011-4G24Y4H-I

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 \(IEEE 802.1ag and ITU-T Y.1731\)”](#) section.

Figure 5: Continuity Check Message Flow



All the MEPs in a service must transmit CCMs at the same interval. IEEE 802.1ag defines the following possible intervals that can be used:

- 100 ms (only supported on bundle members)
- 1 s
- 10 s
- 1 minute
- 10 minutes

A MEP detects a loss of connectivity with one of its peer MEPs when some number of CCMs are 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.

With the exception of bundle members, CFM is supported only on interfaces that have Layer 2 transport feature enabled.

CCM messages carry a variety of information that allows different defects to be detected in the service. This information includes:

- A configured identifier for the domain of the transmitting MEP. This is referred to as the Maintenance Domain Identifier (MDID).
- A configured identifier for the service of the transmitting MEP. This is referred to as the Short MA Name (SMAN). Together, the MDID and the SMAN make up the Maintenance Association Identifier (MAID). The MAID must be configured identically on every MEP in the service.
- These are restrictions on the type of MAID that are supported for sessions with time interval of less than 1 minute. The MAID supports two types of formats on 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

Maintenance Association Identifier (MAID) comprises of the Maintenance Domain Identifier (MDID) and Short MA Name (SMAN). MDID only supports **null** value and SMAN only supports ITU Carrier Code (ICC) or a numerical. No other values are supported.

- An example for configuring domain ID null is: **ethernet cfm domain SMB level 3 id null**
- An example for configuring SMAN is: **ethernet cfm domain SMB level 3 id null service 901234AB xconnect group 99999 p2p 99999 id number 1**
- A configured numeric identifier for the MEP (the MEP ID). Each MEP in the service must be configured with a different MEP ID.
- Dynamic Remote MEPs are not supported for MEPs with less than 1 min interval. You must configure MEP CrossCheck for all such MEPs.
- Sequence numbering is not supported for MEPs with less than 1 minute interval.
- In a Remote Defect Indication (RDI), each MEP includes this 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.
- The interval at which CCMs are being transmitted.
- 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 1 minute intervals.
- The status of the interface where the MEP is operating, for example, whether the interface is up, down, STP blocked, and so on.



Note The status of the interface (up/down) should not be confused with the direction of any MEPs on the interface (Up MEPs/Down MEPs).

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 sending CCMs.
- Level mismatch: A MEP has received a CCM carrying a lower maintenance level than the MEPs own level.
- Loop: A CCM is received with the source MAC address equal to the MAC address of the interface where the MEP is operating.
- Configuration error: A CCM is received with the same MEP ID as the MEP ID configured for the receiving MEP.
- Cross-connect: A CCM is received with a MAID that does not match the locally configured MAID. This generally indicates a VLAN misconfiguration within the network, such that CCMs from one service are leaking into a different service.
- 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.



Note This defect does not cause the MEP to include a remote defect indication in the CCMs that it is sending.

Out-of-sequence CCMs can also be detected by monitoring the sequence number in the received CCMs from each peer MEP. However, this is not considered a CCM defect.

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.

Table 23: Feature History Table

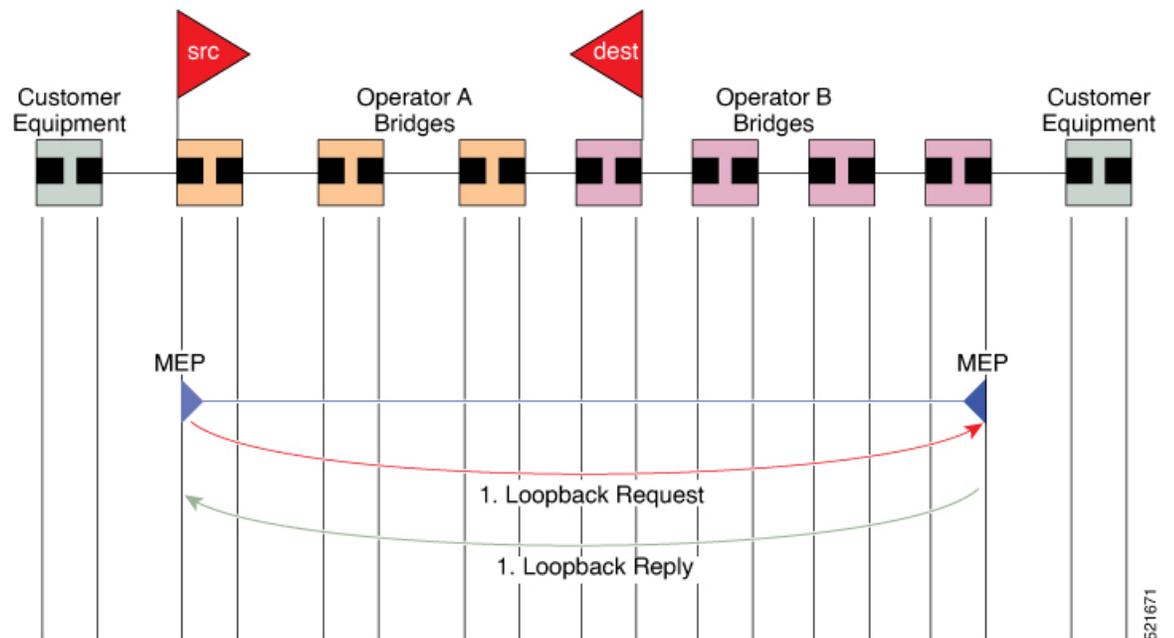
Feature Name	Release Information	Feature Description
Loopback (IEEE 802.1ag and ITU-T Y.1731)	Release 25.3.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>This feature is now supported on:</p> <ul style="list-style-type: none"> • 8011-4G24Y4H-I • 8711-32FH-M • 88-LC1-52Y8H-EM • 88-LC1-12TH24FH-E

Feature Name	Release Information	Feature Description
Loopback (IEEE 802.1ag and ITU-T Y.1731)	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100])</p> <p>You can test connectivity between a local Maintenance End Point (MEP) and a remote Maintenance Point (MP) by sending Loopback Messages (LBM) and Loopback Replies (LBR). The connectivity verification facilitates effective network troubleshooting and maintenance without requiring a detailed hop-by-hop path analysis.</p> <p>This feature is now supported on:</p> <ul style="list-style-type: none"> • 8011-4G24Y4H-I

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 MEP.

Figure 6: 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.

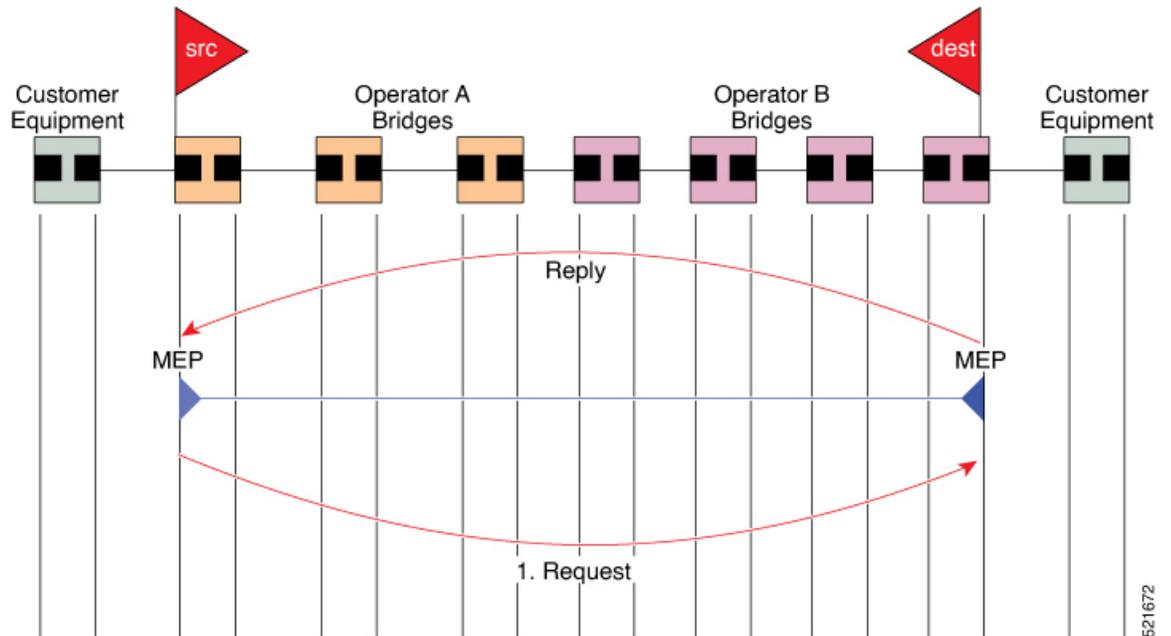
Table 24: Feature History Table

Feature Name	Release Information	Feature Description
Linktrace (IEEE 802.1ag and ITU-T Y.1731)	Release 25.3.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>This feature is now supported on:</p> <ul style="list-style-type: none"> • 8011-4G24Y4H-I • 8711-32FH-M • 88-LC1-52Y8H-EM • 88-LC1-12TH24FH-E
Linktrace (IEEE 802.1ag and ITU-T Y.1731)	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100])</p> <p>You can trace the path to a specific unicast destination MAC address on a network using the Linktrace Messages (LTM) and Linktrace Replies (LTR) diagnostic tools. These tools perform a detailed hop-by-hop path analysis to identify connectivity problems.</p> <p>This feature is now supported on:</p> <ul style="list-style-type: none"> • 8011-4G24Y4H-I

At the request of the operator, a local MEP sends an Linktrace Messages (LTM). Each hop where there is a maintenance point sends an Linktrace Replies (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 MEPs.

Figure 7: 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.

CFM hardware offload

Table 25: Feature History Table

Feature Name	Release Information	Feature Description
CFM hardware offload	Release 25.4.1	Introduced in this release on: Fixed Systems (8700 [ASIC: K100])(select variants only*) *This feature is supported on Cisco 8711-48Z-M routers.
CFM hardware offload	Release 25.3.1	Introduced in this release on: Fixed Systems (8010 [ASIC: A100]) (select variants only*) This feature enables faster fault detection in Connectivity Fault Management (CFM) sessions. It introduces hardware offload capabilities for 3.3 ms, 10 ms, and 100 ms CCM intervals, where the NPU now directly processes these packets. *This feature is supported on 8011-4G24Y4H-I.

Depending on where the continuity check messages (CCMs) are processed, offload is categorized into the following types:

- Software offload—When CCMs are processed by the line card CPU, offload type is known as software offload. Software offload is supported only on bundle interface.
- Hardware offload—When CCMs are processed by network processing unit (NPU), offload type is known as hardware offload.

- Non-offload—When CCMs are processed by route processor (RP), offload type is known as non-offload.

CCM intervals are the intervals in which CCMs are sent and received. If the CCMs are not received within the configured interval, the CFM MEP goes down. The following table shows the supported CCM timers for the offload types:

Interface Type	Offload Type	Supported CCM Timers
Physical interfaces and subinterfaces	Non-offload	<ul style="list-style-type: none"> • 1 sec • 10 sec • 1 min • 10 min
	Hardware offload	<ul style="list-style-type: none"> • 3.3 ms • 10 ms • 100 ms
Bundle members	Non-offload	<ul style="list-style-type: none"> • 1 sec • 10 sec • 1 min • 10 min
	Hardware offload	<ul style="list-style-type: none"> • 3.3 ms • 10 ms • 100 ms
Bundle interfaces and subinterfaces	Non-offload	<ul style="list-style-type: none"> • 1 sec • 10 sec • 1 min • 10 min
	Hardware offload	<ul style="list-style-type: none"> • 3.3 ms • 10 ms • 100 ms

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 Ethernet OAM protocol that detects link error conditions such as miswiring or unidirectional link failure, which are not detected at the physical link layer. This protocol is specifically useful for identifying potential wiring errors caused when using unbundled fiber links, that can lead to a mismatch between the transmitting and receiving connections of a port.

Table 26: Feature History Table

Feature Name	Release	Description
Unidirectional link detection protocol support on physical Ethernet interfaces	Release 25.4.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8011-12G12X4Y-A • 8011-12G12X4Y-D
Unidirectional link detection protocol support on physical Ethernet interfaces	Release 25.2.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC:K100]) (select variants only*)</p> <p>This release supports:</p> <ul style="list-style-type: none"> • Unidirectional Link Detection Protocol (UDLD) on the physical ethernet interfaces. This feature helps detect faults and miswiring conditions with unbundled fiber links and enables each device to understand its own connections as well as those of its neighbors. <p>*This feature is now supported on Cisco 8712-MOD-M routers.</p>
Unidirectional link detection protocol support on physical Ethernet interfaces	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100]) (select variants only*)</p> <p>* This feature is supported on:</p> <ul style="list-style-type: none"> • 8011-4G24Y4H-I

Feature Name	Release	Description
Unidirectional link detection protocol support on physical Ethernet interfaces	Release 24.4.1	<p>Introduced in this release on: Fixed Systems(8200, 8700)(select variants only*); Modular Systems (8800 [LC ASIC: P100]).</p> <p>The Unidirectional Link Detection Protocol (UDLD) is now supported on the Physical Ethernet interfaces on the Cisco Silicon One P100 ASIC-based Systems. This feature helps detect faults and miswiring conditions with unbundled fiber links and enables each device to understand its own connections as well as those of its neighbors.</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 8712-MOD-M <p>This feature introduces these changes:</p> <p>CLI:</p> <ul style="list-style-type: none"> • clear ethernet udld statistics • ethernet udld reset interface • show ethernet udld interfaces • show ethernet udld statistics

UDLD operation

UDLD exchanges protocol packets between the neighboring devices. UDLD works if both devices on the link support UDLD and have it enabled on respective ports.

UDLD sends an initial PROBE message on the ports where it is configured. When it receives a PROBE message, UDLD sends periodic ECHO (hello) messages. Both the messages help identify the sender and its port, and also contain some information about the operating parameters of the protocol on that port. The messages also contain the device and port identifiers on the port for any neighbor devices that the local device has connected with on the port. Similarly, each device gets to know where it is connected and where its neighbors are connected. This helps in detecting faults and miswiring conditions.

The protocol employs a mechanism where information from neighbors that is not periodically refreshed is eventually timed out for fault detection.

The protocol uses a FLUSH message to indicate when UDLD is disabled on a port. This causes the peers to remove the local device from their neighbor cache to prevent a time out.

If a problem is detected, UDLD disables the affected interface and notifies the user to avoid further network problems beyond traffic loss. Example: Loops which are not detected or prevented by Spanning Tree Protocol (STP).

Types of fault detection

UDLD can detect these types of faults:

- **Transmit faults** — These are transmission failures from the local port to the peer device which also includes the faults caused by physical link failure or packet path issues on the local or peer device. These failures can lead to serious network issues such as loops which occur specifically when a link is unidirectional.
- **Miswiring faults** — These are instances that occur when using unbundled fibers to connect fiber optic ports. In such instances, 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).
- **Loopback faults** — In these instances, 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 uses 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 namely **Normal** mode and **Aggressive** mode, which determine the behavior on a heartbeat timeout. For more information about these modes, see [UDLD modes of operation, on page 118](#).

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.

UDLD aging mechanism

Aging of UDLD information occurs in a `Receive Fault` condition when the port that runs UDLD does not receive UDLD packets from the neighbor port for a duration of the hold time. The hold time for the port is dictated by the remote port and is dependent on the message interval at the remote end. The shorter the message interval, the shorter is the hold time and faster the detection of the fault. The hold time is three times the message interval in Cisco IOS XR Software.

UDLD information can age out due to the high error rate on the port caused by a physical issue or duplex mismatch. Packet drops due to age out does not mean that the link is unidirectional. UDLD in normal mode does not disable such link.

It is important to choose the right message interval to ensure proper detection time. The message interval should be fast enough to detect the unidirectional link before the forwarding loop is created. The default message interval is 60 seconds. The detection time is approximately equal to three times the message interval. Therefore, when using default UDLD timers, UDLD does not timeout the link faster than the STP aging time.

UDLD state machines

UDLD uses two types of finite state machines (FSMs), generally referred as state machines. The Main FSM deals with all the phases of operation of the protocol while the Detection FSM handles only the phases that determine the status of a port.

Main FSM

The Main FSM can be in one of these states:

- **Init:** Protocol is initializing.
- **UDLD inactive:** Port is down or UDLD is disabled.
- **Linkup:** Port is up and running, and UDLD is in the process of detecting a neighbor.
- **Detection:** A hello message from a new neighbor has been received and the Detection FSM is running to determine the status of the port.
- **Advertisement:** The Detection FSM has run and concluded that the port is operating correctly, periodic hello messages are being sent and the hello messages from neighbors are monitored.
- **Port shutdown:** The Detection FSM detected a fault, or all neighbors were timed out in Aggressive mode, and the port has been disabled as a result.

Detection FSM

The Detection FSM can be in one of these states:

- **Unknown:** Detection has not yet been performed or UDLD has been disabled.
- **Unidirectional detected:** A unidirectional link condition has been detected because a neighbor does not see the local device, the port will be disabled.
- **Tx/Rx loop:** A loopback condition has been detected by receiving a type, length, and value (TLV) message with the ports own identifiers, the port will be disabled.
- **Neighbor mismatch:** A miswiring condition has been detected in which a neighbor can identify other devices than the devices the local device can see and the port will be disabled.
- **Bidirectional detected:** UDLD hello messages are exchanged successfully in both the directions, the port is operating correctly.

Limitations

- UDLD on Cisco 8000 Series Routers does not work if the peer UDLD configuration has custom MAC address; Peer must have either Cisco MAC address or IEEE Slow Proto MAC address.
- Use only these MAC Addresses to establish a successful connection and communication with the Cisco 8000 Series Routers.
 - cisco-l2cp (0x01000cccccc) - Cisco proprietary MAC Address which can also be used by all other Cisco protocols.
 - ieee-slow-protocols (0x0180c200002) - IEEE Slow Protocol MAC Address.
- UDLD is not tunneled through L2VPN like other slow protocols.
- UDLD must not be enabled on a Switched Port Analyzer (SPAN) source or a destination port.
- The UDLD protocol is not supported on the subinterfaces and bundle interfaces.

Configure UDLD

SUMMARY STEPS

1. **configure**
2. **interface** [**GigabitEthernet** | **TenGigE**] *interface-path-id*
3. **ethernet udld**
4. **mode** {**normal** |**aggressive**}
5. **message-time**
6. **logging disable**
7. **end**

DETAILED STEPS

Procedure

Step 1 **configure**

Example:

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

Enters global configuration mode.

Step 2 **interface** [**GigabitEthernet** | **TenGigE**] *interface-path-id*

Example:

```
RP/0/RSP0/CPU0:router(config)# interface
TenGigE 0/1/0/0
```

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

Note

The example indicates an 8-port 10-Gigabit Ethernet interface in modular services card slot 1.

Step 3 **ethernet udld****Example:**

```
RP/0/RSP0/CPU0:router(config-if)# ethernet udld
```

Enables ethernet UDLD functionality and enters interface Ethernet UDLD configuration mode.

Step 4 **mode {normal |aggressive}****Example:**

```
RP/0/RSP0/CPU0:router(config-if-udld)# mode normal
```

(Optional) Specifies the mode of operation for UDLD. The options are normal and aggressive.

Step 5 **message-time****Example:**

```
RP/0/RSP0/CPU0:router(config-if-udld)# message-time 70
```

(Optional) Specifies the message time (in seconds) to use for the UDLD protocol. The value ranges from 7 to 90 seconds.

Step 6 **logging disable****Example:**

```
RP/0/RSP0/CPU0:router(config-if-udld)# logging disable
```

(Optional) Suppresses the operational UDLD syslog messages.

Step 7 **end****Example:**

```
RP/0/RSP0/CPU0:router(config-if-udld)# end
```

Ends the configuration session and exits to the EXEC mode.

How to Configure Ethernet OAM

This section provides these configuration procedures:

Configuring Ethernet 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.

Procedure

	Command or Action	Purpose
Step 1	configure Example: <pre>RP/0/RP0/CPU0:router# configure terminal</pre>	Enters global configuration mode.
Step 2	ethernet oam profile <i>profile-name</i> Example: <pre>RP/0/RP0/CPU0:router(config)# ethernet oam profile Profile_1</pre>	Creates a new Ethernet Operations, Administration and Maintenance (OAM) profile and enters Ethernet OAM configuration mode.
Step 3	link-monitor Example: <pre>RP/0/RP0/CPU0:router(config-eoam)# link-monitor</pre>	Enters the Ethernet OAM link monitor configuration mode.
Step 4	symbol-period window { <i>milliseconds window</i> <i>symbols window</i> [<i>thousand</i> <i>million</i> <i>billion</i>] } Example: <pre>RP/0/RP0/CPU0:router(config-eoam-lm)# symbol-period window 60000</pre>	(Optional) Configures the window size for an Ethernet OAM symbol-period error event. If specified in milliseconds, the range is 1000 to 60000. If not specified as a multiple of 1 second, the actual window used will be rounded up to the nearest second, with thresholds scaled accordingly. If specified in symbols, the range is interface speed dependent (must be between the maximum number of symbols that could be received in 1 second and the maximum number of symbols that could be received in 1 minute). Again the actual window used is rounded up to the nearest second, with thresholds scaled accordingly. The default value is 1000 milliseconds.
Step 5	symbol-period threshold { <i>ppm</i>[<i>low threshold</i>] [<i>high threshold</i>] <i>symbols</i> [<i>low threshold</i> [<i>thousand</i> <i>million</i> <i>billion</i>]] [<i>high threshold</i> [<i>thousand</i> <i>million</i> <i>billion</i>]] } Example: <pre>RP/0/RP0/CPU0:router(config-eoam-lm)# symbol-period threshold ppm low 100 high 1000000</pre>	(Optional) Configures the thresholds that trigger an Ethernet OAM symbol-period error event, in symbols or ppm (errors per million symbols). When using this command at least one of the high and low thresholds must be specified. If the low threshold is not specified, the default value is used. If the high threshold is not specified, no action is performed in response to an event. The high threshold must not be smaller than the low threshold.

	Command or Action	Purpose
		<p>If specified in ppm, the range (for both thresholds) is 1 to 1000000. If specified in symbols, the range (for both thresholds) is 1 to the maximum window size in symbols, see symbol-period window.</p> <p>The default low threshold is 1 symbol.</p>
Step 6	<p>frame window milliseconds <i>window</i></p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-eoam-lm)# frame window milliseconds 60</pre>	<p>(Optional) Configures the frame window size (in milliseconds) of an OAM frame error event.</p> <p>The range is from 1000 to 60000.</p> <p>The default value is 1000.</p>
Step 7	<p>frame threshold [<i>low threshold</i>] [high threshold]</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-eoam-lm)# frame threshold low 10000000 high 60000000</pre>	<p>(Optional) Configures the thresholds (in symbols) that triggers an Ethernet OAM frame error event. When using this command at least one of the high and low thresholds must be specified. If the low threshold is not specified, the default value is used. If the high threshold is not specified, no action is performed in response to an event. The high threshold must not be smaller than the low threshold.</p> <p>The range is from 1 to 60000000.</p> <p>The default low threshold is 1.</p>
Step 8	<p>frame-period window { milliseconds <i>window</i> frames <i>window</i> [thousand million billion] }</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-eoam-lm)# frame-period window milliseconds 60000</pre>	<p>(Optional) Configures the window size for an Ethernet OAM frame-period error event.</p> <p>The range is from 100 to 60000, if defined in milliseconds. If the window is defined as say, 200ms, and the interface could receive at most say 10000 minimum size frames in 200ms, then the actual window size used will be the time taken to receive 10000 frames, rounded up to the nearest second. The thresholds will be scaled accordingly.</p> <p>If specified in frames, the range is interface speed dependent, but must be between the number of minimum size frames that could be received in 100ms and the number of minimum size frames that could be received in 1 minute. If the window is defined as 20000 frames, the actual window size used will be the time taken to receive 20000 frames, rounded up to the nearest second. The thresholds will be scaled accordingly.</p> <p>The default value is 1000 milliseconds.</p>
Step 9	<p>frame-period threshold { ppm [<i>low threshold</i>] [high threshold] frames [<i>low threshold</i> [thousand million billion]] [high threshold [thousand million billion]] }</p> <p>Example:</p>	<p>(Optional) Configures the thresholds (either in frames or in ppm - errors per million frames) that trigger an Ethernet OAM frame-period error event. When using this command at least one of the high and low thresholds must be specified. If the low threshold is not specified, the default value is used. If the high threshold is not specified, no</p>

	Command or Action	Purpose
	<pre>RP/0/RP0/CPU0:router(config-eoam-lm)# frame-period threshold ppm low 100 high 1000000</pre>	<p>action is performed in response to an event. The high threshold must not be smaller than the low threshold.</p> <p>The range for both thresholds is from 1 to 1000000 if specified in ppm. If specified in frames, the range is from 1 to the maximum frame-period window size in frames, see frame-period window.</p> <p>The default low threshold is 1 ppm.</p>
Step 10	<p>frame-seconds window milliseconds <i>window</i></p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-eoam-lm)# frame-seconds window milliseconds 900000</pre>	<p>(Optional) Configures the window size (in milliseconds) for the OAM frame-seconds error event.</p> <p>The range is 10000 to 900000.</p> <p>The default value is 6000.</p>
Step 11	<p>frame-seconds threshold [<i>low threshold</i>] [<i>high threshold</i>]</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-eoam-lm)# frame-seconds threshold low 3 threshold high 900</pre>	<p>(Optional) Configures the thresholds (in seconds) that trigger a frame-seconds error event. When using this command at least one of the high and low thresholds must be specified. If the low threshold is not specified, the default value is used. If the high threshold is not specified, no action is performed in response to an event. The high threshold must not be smaller than the low threshold.</p> <p>The range is 1 to 900</p> <p>The default value is 1.</p>
Step 12	<p>exit</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-eoam-lm)# exit</pre>	Exits back to Ethernet OAM mode.
Step 13	<p>mib-retrieval</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-eoam)# mib-retrieval</pre>	Enables MIB retrieval in an Ethernet OAM profile or on an Ethernet OAM interface.
Step 14	<p>connection timeout <i><timeout></i></p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-eoam)# connection timeout 30</pre>	<p>Configures the connection timeout period for an Ethernet OAM session, as a multiple of the hello interval.</p> <p>The range is 2 to 30.</p> <p>The default value is 5.</p>
Step 15	<p>hello-interval <i>1s</i></p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-eoam)# hello-interval 1s</pre>	Configures the time interval between hello packets for an Ethernet OAM session. The default is 1 second (1s).

	Command or Action	Purpose
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.
Step 18	require-remote mib-retrieval Example: RP/0/RP0/CPU0:router(config-eoam)# require-remote mib-retrieval	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 log} Example: RP/0/RP0/CPU0:router(config-eoam)# action capabilities-conflict efd	Specifies the action that is taken on an interface when a capabilities-conflict event occurs. The default action is to create a syslog entry.
Step 20	action critical-event {disable error-disable-interface log} Example: RP/0/RP0/CPU0:router(config-eoam)# action critical-event error-disable-interface	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.
Step 21	action discovery-timeout {disable efd error-disable-interface log} Example: RP/0/RP0/CPU0:router(config-eoam)# action discovery-timeout efd	Specifies the action that is taken on an interface when a connection timeout occurs. The default action is to create a syslog entry.
Step 22	action dying-gasp {disable error-disable-interface log} Example: RP/0/RP0/CPU0:router(config-eoam)# action dying-gasp error-disable-interface	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.
Step 23	action high-threshold {disable error-disable-interface log} Example:	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.

	Command or Action	Purpose
	<pre>RP/0/RP0/CPU0:router(config-eoam)# action high-threshold error-disable-interface</pre>	
Step 24	<p>action session-down {disable efd error-disable-interface log}</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-eoam)# action session-down efd</pre>	Specifies the action that is taken on an interface when an Ethernet OAM session goes down.
Step 25	<p>action session-up {disable log}</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-eoam)# action session-up disable</pre>	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.
Step 26	<p>action uni-directional link-fault {disable efd error-disable-interface log}</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 In Cisco IOS XR Release 4.x, this command replaces the action link-fault command.</p>
Step 27	<p>action wiring-conflict {disable efd error-disable-interface log}</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-eoam)# action session-down efd</pre>	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.
Step 28	<p>uni-directional link-fault detection</p> <p>Example:</p> <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	<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 30	<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.

Attaching an Ethernet OAM Profile to an Interface

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

Procedure

	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: <pre>RP/0/RP0/CPU0:router(config-if)# ethernet oam</pre>	Enables Ethernet OAM and enters interface Ethernet OAM configuration mode.
Step 4	profile profile-name Example: <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	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 6	end Example: <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” section](#).

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

Procedure

	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 <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: 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-eoam)# 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.

	Command or Action	Purpose
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
```

Configuring Ethernet CFM

To configure Ethernet CFM, perform the following tasks:



Note CFM is not supported for the following:

- L3 Interfaces and Sub-Interfaces
- Bridge Domain, Release 7.3.1 and earlier
- VPLS, Release 7.3.1 and earlier

Configuring a CFM Maintenance Domain

To configure a CFM maintenance domain, perform 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. **traceroute cache hold-time** *minutes* **size** *entries*
5. **end** or **commit**

DETAILED STEPS

Procedure

	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	Creates and names a container for all domain configurations and enters 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 identifier (MAID) in CFM frames. If the MDID is not specified, the domain name is used as the MDID by default.

	Command or Action	Purpose
Step 4	<p>traceroute cache hold-time <i>minutes</i> size <i>entries</i></p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-cfm)# traceroute cache hold-time 1 size 3000</pre>	<p>(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.</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

From Release 24.4.1, Cisco 8000 routers support 500 CFM sessions.

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* **p2p** *xconnect-name*} [**id** [**icc-based** *icc-string umc-string*] | [**number** *number*]]
- end** or **commit**

DETAILED STEPS

Procedure

	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 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 at a specified maintenance level, and enters CFM domain configuration mode. 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</i> <i>umc-string</i>] [[number <i>number</i>]] Example: RP/0/RP0/CPU0:router(config-cfm-dmn)# service xconnect group X1	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. The id sets the short MA name.
Step 5	end or commit Example: RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# 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. 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

- 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 p2p xconnect-name} [id [icc-based icc-string umc-string] | [number number]*
- continuity-check interval** *time [loss-threshold threshold]*
- continuity-check archive hold-time** *minutes*
- continuity-check loss auto-traceroute**
- end** or **commit**

DETAILED STEPS

Procedure

	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 level level-value [id [null] [dns DNS-name] [mac H.H.H] [string 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>

	Command or Action	Purpose
Step 4	<p>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>]</p> <p>Example:</p> <pre>RP/0/RP0/CPU0:router(config-cfm-dmn)# service xconnect group X1</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 xconnect where up MEPs will be created.</p> <p>The id sets the short MA name.</p>
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>	<p>(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.</p>
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>	<p>(Optional) Configures how long information about peer MEPs is stored after they have timed out.</p>
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>	<p>(Optional) Configures automatic triggering of a traceroute when a MEP is declared down.</p>
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 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

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*] | [**string** *text*] | [**number** *number*] | [**vlan-id** *id-number*] | [**vpn-id** *oui-vpnid*]]
5. **mep crosscheck**
6. **mep-id** *mep-id-number*
7. **end** or **commit**

DETAILED STEPS

Procedure

	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# 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 <i>null</i>] [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> { 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>]]	<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 xconnect where up MEPs will be created.</p> <p>The id sets the short MA name.</p>

	Command or Action	Purpose
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 mep-id-number Example: <pre>RP/0/RP0/CPU0:router(config-cfm-xcheck)# mep-id 10</pre>	Enables cross-check on a MEP. Note <ul style="list-style-type: none"> For non-offloaded and software-offloaded MEPs, use the mep-id mep-id-number [mac-address mac-address] command. For hardware-offloaded MEPs, use the mep-id mep-id-number command. From Release 24.2.11, mac-address mac-address option is obsolete for hardware-offloaded MEPs. 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>	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. 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* {**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

Procedure

	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# <code>ethernet cfm</code>	Enters the Ethernet Connectivity Fault Management (CFM) configuration mode.
Step 3	domain <i>domain-name level 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)# <code>domain Domain_One level 1 id string D1</code>	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 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 p2p 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>]]	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 an xconnect where up MEPs will be created. The id sets the short MA name.
Step 5	maximum-meps <i>number</i> Example: RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# <code>maximum-meps 1000</code>	(Optional) Configures the maximum number (2 to 8190) of MEPs across the network, which limits the number of peer MEPs recorded in the database.

	Command or Action	Purpose
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

- configure**
- interface** {**HundredGigE** | **TenGigE**} *interface-path-id*
- interface** {**HundredGigE** | **TenGigE** | **Bundle-Ether**} *interface-path-id*2*transport*
- ethernet cfm**
- mep domain** *domain-name* **service** *service-name* **mep-id** *id-number*
- cos** *cos*
- end** or **commit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters global configuration mode.
Step 2	interface {HundredGigE TenGigE} interface-path-id 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. • L3 interfaces are only supported for bundle member interfaces. Else, you must enable l2transport.
Step 3	interface {HundredGigE TenGigE Bundle-Ether} interface-path-id.l2transport 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 , TenGigE , or Bundle-Ether and the physical interface or virtual interface followed by the l2transport. L2transport configures the interface as an L2 interface. 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	ethernet cfm Example: RP/0/RP0/CPU0:router(config-if)# ethernet cfm	Enters interface Ethernet CFM configuration mode.
Step 5	mep domain domain-name service service-name mep-id id-number 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 6	cos cos 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

	Command or Action	Purpose
		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 7	end or commit Example: <pre>RP/0/RP0/CPU0:router(config-if-cfm-mep) # commit</pre>	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. • 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* **xconnect** **group** *xconnect-group-name* **p2p** *xconnect-name*
5. **ais transmission** [**interval** {**1s**|**1m**}][**cos** *cos*]
6. **log ais**
7. **end** or **commit**

DETAILED STEPS

Procedure

	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 xconnect group xconnect-group-name p2p xconnect-name Example: RP/0/RP0/CPU0:router(config-cfm-dmn)# service S1 xconnect group XG1 p2p X2	Specifies the service and cross-connect group and name.
Step 5	ais transmission [interval {1s 1m}][cos cos] Example: RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# ais transmission interval 1m cos 7	Configures Alarm Indication Signal (AIS) transmission for a Connectivity Fault Management (CFM) domain service.
Step 6	log ais Example: RP/0/RP0/CPU0:router(config-cfm-dmn-svc)# log ais	Configures AIS logging for a Connectivity Fault Management (CFM) domain service to indicate when AIS or LCK packets are received.
Step 7	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 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

Procedure

	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:	Configures Alarm Indication Signal (AIS) transmission on a Connectivity Fault Management (CFM) interface.

	Command or Action	Purpose
	RP/0/RP0/CPU0:router(config-if-cfm)# ais transmission up interval 1m cos 7	
Step 5	<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.

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.

CFM Over Bundles

CFM over bundle supports the following:

- CFM Maintenance Points — UP MEP, Down MEP, which only includes L2 bundle main and sub-interfaces.
- CCM interval of 3.3 ms, 10 ms, 100 ms, 1 sec, 10 sec, 1 min, and 10 mins.
- RP OIR/VM reload without impacting learnt CFM peer MEPs.
- Process restart without impacting CFM sessions.
- Static MEPs.

Restrictions for Configuration of CFM on Bundles

Following are the restrictions for configuring CFM over bundle member interfaces:

- Only Layer 2 bundle Ethernet interfaces and sub-interfaces are supported, which are part of a L2VPN cross-connect.
- Supports 5000 pps rates of CCM traffic for bundle interfaces.
- Ethernet Connectivity Fault Management (CFM) is not supported with Maintenance association End Points (MEPs) that are configured on default and untagged encapsulated sub-interfaces that are part of a single physical interface.
- Multiple MEPs of different directions are not supported on the same interface or Xconnect.
- CFM does not support fast failover, which may result in session flaps on bundle interfaces. Use offload for virtual interfaces to avoid flaps on faster CCM intervals.

Ethernet SLA Statistics Measurement in a Profile

Table 27: Feature History Table

Feature Name	Release Information	Feature Description
Enhancement to Ethernet SLA Statistics Measurement	Release 7.7.1	<p>You can now configure the size of bins for the delay and jitter measurement in Ethernet SLA statistics with a width value ranging from 1 to 10000000 microseconds. This enhancement provides granularity to store more accurate results of SLA statistics in the aggregate bins.</p> <p>In earlier releases, you could only configure the width value for the delay and jitter measurement in milliseconds.</p> <p>This feature introduces the usec keyword in the aggregate command.</p>

The Ethernet SLA feature supports measurement of one-way and two-way delay and jitter statistics, and one-way FLR statistics.

Ethernet SLA statistics measurement for network performance is performed by sending packets and storing data metrics such as:

- Round-trip delay time—The time for a packet to travel from source to destination and back to source again.
- Round-trip jitter—The variance in round-trip delay time (latency).
- One-way delay and jitter—The router also supports measurement of one-way delay or jitter from source to destination, or from destination to source.
- One-way frame loss—The router also supports measurement of one-way frame loss from source to destination, or from destination to source.

In addition to these metrics, these statistics are also kept for SLA probe packets:

- Packet loss count
- Packet corruption event
- Out-of-order event
- Frame Loss Ratio (FLR)

Counters for packet loss, corruption, and, out-of-order packets are kept for each bucket, and in each case, a percentage of the total number of samples for that bucket is reported (for example, 4% packet corruption). For delay, jitter, and loss statistics, the minimum, maximum, mean and standard deviation for the whole bucket are reported, as well as the individual samples or aggregated bins. Also, the overall FLR for the bucket, and individual FLR measurements or aggregated bins are reported for synthetic loss measurement statistics. The packet loss count is the overall number of measurement packets lost in either direction and the one-way FLR measures the loss in each direction separately.

When aggregation is enabled using the **aggregate** command, bins are created to store a count of the samples that fall within a certain value range, which is set by the **width** keyword. Only a counter of the number of results that fall within the range for each bin is stored. This uses less memory than storing individual results. When aggregation is not used, each sample is stored separately, which can provide a more accurate statistics analysis for the operation, but it is highly memory-intensive due to the independent storage of each sample.

A bucket represents a time period during which statistics are collected. All the results received during that time period are recorded in the corresponding bucket. If aggregation is enabled, each bucket has its own set of bins and counters, and only results relating to the measurements initiated during the time period represented by the bucket are included in those counters.

Frame Loss Ratio (FLR) is a primary attribute that can be calculated based on loss measurements. FLR is defined by the ratio of lost packets to sent packets and expressed as a percentage value. FLR is measured in each direction (source to destination and destination to source) separately. Availability is an attribute that is typically measured over a long period of time, such as weeks or months. The intent is to measure the proportion of time when there was prolonged high loss.

To configure one-way delay or jitter measurements, you must first configure the **profile (SLA)** command using the **type cfm-delay-measurement** form of the command.

For valid one-way delay results, you must have both local and remote devices time synchronized. In order to do this, you must select sources for frequency and time-of-day (ToD).

Frequency selection can be between any source of frequency available to the router, such as: BITS, GPS, SyncE, or PTP. The ToD selection is between the source selected for frequency and PTP or DTI. Note that NTP is not sufficient.

Configuration Guidelines



Caution Certain SLA configurations can use a large amount of memory which can affect the performance of other features on the router.

Before you configure Ethernet SLA, consider the following guidelines:

- Aggregation—Use of the **aggregate none** command significantly increases the amount of memory required because each individual measurement is recorded, rather than just counts for each aggregation bin. When you configure aggregation, consider that more bins will require more memory.
- Buckets archive—When you configure the **buckets archive** command, consider that the more history that is kept, the more memory will be used.
- Measuring two statistics (such as both delay and jitter) will use approximately twice as much memory as measuring one.
- Separate statistics are stored for one-way source-to-destination and destination-to-source measurements, which consumes twice as much memory as storing a single set of round-trip statistics.
- You must define the schedule before you configure SLA probe parameters to send probes for a particular profile. It is recommended to set up the profile—probe, statistics, and schedule before any commit.

Restrictions

One-way delay and jitter measurements are not supported by cfm-loopback profile types.

Configure Ethernet SLA Statistics Measurement in a Profile

To configure SLA statistics measurement in a profile, perform these steps:

1. Enter the Ethernet SLA configuration mode, using the **ethernet sla** command in Global Configuration mode.
2. Create an SLA operation profile with the **profile *profile-name* type cfm-delay-measurement** command.
3. Enable the collection of SLA statistics using the **statistics measure {one-way-delay-ds | one-way-delay-sd | one-way-jitter-ds | one-way-jitter-sd | round-trip-delay | round-trip-jitter | one-way-loss-ds | one-way-loss-sd}** command.
4. Configure the size and number of bins into which to aggregate the results of statistics collection. For delay measurements and data loss measurements, the default is that all values are aggregated into 1 bin. For synthetic loss measurements, by default the aggregation is disabled. Use the **aggregate {bins *count* width [*usec*] width | none}** command to configure the bins.
 - For delay and jitter measurements, you can configure a width value from 1 to 10000 milliseconds, if the number of bins is at least 2. To configure the width value in microseconds, use the **usec** option. You can configure the width value from 1 to 10000000 microseconds.

- For data loss and synthetic loss measurements, you can configure a width value from 1 to 100 percentage points, if the number of bins is at least 2.

5. Configure the size of the buckets in which statistics are collected, using the **buckets size number probes** command.
6. Configure the number of buckets to store in memory using the **buckets archive number** command.
7. Save the configuration changes using the **end** or **commit** command.

Configuration Example

This example shows configuration of round-trip-delay statistics measurement in 5 bins each with a range of 123 microseconds:

```
Router(config)# ethernet sla
Router(config-sla)# profile test type cfm-delay-measurement
Router(config-sla-prof)# statistics measure round-trip-delay
Router(config-sla-prof-stat-cfg)# aggregate bins 5 width usec 123
Router(config-sla-prof-stat-cfg)# buckets size 1 probes
Router(config-sla-prof-stat-cfg)# buckets archive 50
Router(config-sla-prof-stat-cfg)# commit
```

This example shows configuration of round-trip-delay statistics measurement in 5 bins each with a range of 10 milliseconds:

```
Router(config)# ethernet sla
Router(config-sla)# profile test type cfm-delay-measurement
Router(config-sla-prof)# statistics measure round-trip-delay
Router(config-sla-prof-stat-cfg)# aggregate bins 5 width 10
Router(config-sla-prof-stat-cfg)# buckets size 1 probes
Router(config-sla-prof-stat-cfg)# buckets archive 50
Router(config-sla-prof-stat-cfg)# commit
```

Verification

This example displays aggregate bins configured with a range of 123 microseconds:

```
Router# show ethernet sla statistics detail
Tue Sep 28 07:59:22.340 PDT
Source: Interface GigabitEthernet0/0/0/2, Domain dom1
Destination: Target MAC Address 0012.0034.0056
=====
Profile 'test', packet type 'cfm-delay-measurement'
Scheduled to run every lmin first at 00:00:31 UTC for 10s

Round Trip Delay
~~~~~
1 probes per bucket

No stateful thresholds.

Bucket started at 07:56:31 PDT Tue 28 September 2021 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: 0.000ms, occurred at 07:56:32 PDT Tue 28 September 2021
  Max: 1.000ms, occurred at 07:56:31 PDT Tue 28 September 2021
  Mean: 0.100ms; StdDev: 0.300ms
```

```

Bins:
Range          Samples  Cum. Count  Mean
-----
  0 to 0.123 ms  9 (90.0%)  9 (90.0%)  0.000ms
0.123 to 0.246 ms  0 (0.0%)  9 (90.0%)  -
0.246 to 0.369 ms  0 (0.0%)  9 (90.0%)  -
0.369 to 0.492 ms  0 (0.0%)  9 (90.0%)  -
> 0.492      ms  1 (10.0%) 10 (100.0%) 1.000ms

```

This example displays aggregate bins configured with a range of 10 milliseconds:

```

Router# show ethernet sla statistics detail
Tue Sep 28 08:00:57.527 PDT
Source: Interface GigabitEthernet0/0/0/2, Domain dom1
Destination: Target MAC Address 0012.0034.0056
=====
Profile 'test', packet type 'cfm-delay-measurement'
Scheduled to run every 1min first at 00:00:31 UTC for 10s

Round Trip Delay
~~~~~
1 probes per bucket

No stateful thresholds.

Bucket started at 08:00:32 PDT Tue 28 September 2021 lasting 10s
  Pkts sent: 9; Lost: 0 (0.0%); Corrupt: 0 (0.0%);
    Misordered: 1 (11.1%); Duplicates: 0 (0.0%)
  Result count: 9
  Min: 0.000ms, occurred at 08:00:32 PDT Tue 28 September 2021
  Max: 0.000ms, occurred at 08:00:32 PDT Tue 28 September 2021
  Mean: 0.000ms; StdDev: 0.000ms

Results suspect due to a probe starting mid-way through a bucket

Bins:
Range          Samples  Cum. Count  Mean
-----
  0 to 10 ms  9 (100.0%)  9 (100.0%)  0.000ms
10 to 20 ms  0 (0.0%)  9 (100.0%)  -
20 to 30 ms  0 (0.0%)  9 (100.0%)  -
30 to 40 ms  0 (0.0%)  9 (100.0%)  -
> 40      ms  0 (0.0%)  9 (100.0%)  -

```

Ethernet frame delay measurement for L2VPN services

Ethernet frame delay measurement complies with the ITU-T Y.1731 standard, which provides comprehensive fault management and performance monitoring recommendations. Delay Measurement Message (DMM) and Delay Measurement Reply (DMR) are used to periodically measure one-way or two-way frame delay and frame delay variation between a pair of point-to-point MEPs. Measurements are made between two MEPs belonging to the same domain and Maintenance Association (MA).

Table 28: Feature History Table

Feature Name	Release Information	Feature Description
Ethernet frame delay measurement for L2VPN services	Release 25.4.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100], 8700 [ASIC: K100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8011-12G12X4Y-A • 8011-12G12X4Y-D • 8711-48Z-M
Ethernet frame delay measurement for L2VPN services	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100, K100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 8712-MOD-M • 88-LC1-12TH24FH-E • 88-LC1-36EH+A8:B12 • 88-LC1-52Y8H-EM
Ethernet frame delay measurement for L2VPN services	Release 7.5.3	<p>You can now monitor L2VPN networks and avoid impact to your customers' operations by accurately measuring frame round-trip delays and jitters between two maintenance endpoints (MEPs).</p> <p>This feature lets you detect end-to-end connectivity, loopback, and link trace on MEPs. It reports service performance to your end customers, helping improve technical and operational tasks such as troubleshooting and billing.</p> <p>This feature introduces the cfm-delay-measurement probe command.</p>

You can measure frame delay in the Layer 2 networks to detect end-to-end connectivity, loopback, and link trace on Maintenance End Points (MEPs) and also report service performance that helps to improve technical and operational tasks such as troubleshooting, billing, and so on. Frame delay is the duration between the time the source node transmits the first bit of a frame and the time the same source node receives the last bit of the frame.

The frame delay measurement uses the following two protocol data units (PDUs):

- Delay Measurement Message (DMM)—DMM is used to measure frame delay and frame delay variation between a pair of point-to-point Maintenance End Points (MEPs).
- Delay Measurement Response (DMR)—DMR is the delay measurement response sent by the destination MEP. When an MEP receives a DMM frame, the responder MEP responds with a DMR frame. The DMR frame carries a reply information and a copy of the timestamp contained in the DMM frame.



Note DMM sessions (using CFM) are not supported with MACsec enabled on the core interface, as this requires pre-encryption timestamping in the interface group.

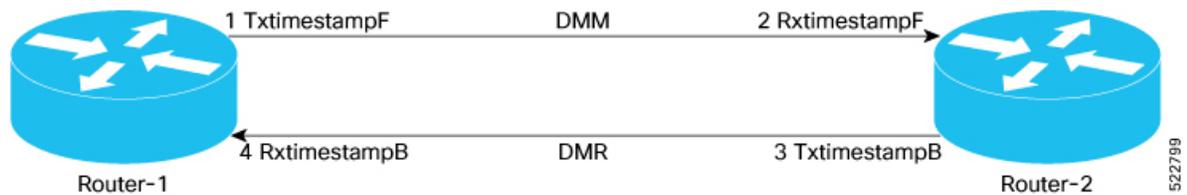
We support one-way and two-way frame delay measurement.

Frame Delay Measurement	Description
One-way frame delay measurement (1DM)	<ul style="list-style-type: none"> • Measures the frame delay on a unidirectional link between the MEPs. • 1DM requires that clocks at both the transmitting MEP and the receiving MEPs are synchronized. • Measuring frame-delay variation does not require clock synchronization and the variation can be measured using 1DM and DMR frame combination.
Two-way frame delay measurement	<ul style="list-style-type: none"> • Measures the frame delay on a bidirectional link between the MEPs. • Two-way delay measurement does not require the clocks at both the transmitting MEP and the receiving MEPs to be synchronized. • The two-way frame delay is measured using only DMM and DMR frames.

For more information about CFM, see [Configuring Ethernet OAM, on page 89](#).

Topology

Let's see how a round-trip frame delay is measured with the following sample topology.



- The sender MEP (Router-1) transmits a frame containing delay measurement request information and the timestamp at the which router sends the DMM.
- When packets pass through each interface, timestamps are written into DMMs and DMRs at both local and peer MEPs.
- When the DMM leaves the local interface, the TX timestamp is added to the packet.
- When the receiver MEP (Router-2) receives the frame, records the timestamp at which the receiver MEP receives the frame with the delay measurement request information and the remote MEP (Router-2) responds with an DMR adding the remote TX timestamp to the packet as it leaves the remote interface.

To measure a round-trip delay for a traffic exchange between Router-1 and Router-2, four timestamps get populated as the packet moves through the network.

- Router-1 adds the TxTimestampF when DMM packet is transmitted.
- Router-2 adds RxTimestampF when DMM packet is received by it.
- Router-2 adds TxTimestampB when DMR packet it transmitted.
- Router-1 adds RxTimestampB when DMR is received by it

The round-trip delay is calculated using the following formula:

$$\begin{aligned}
 \text{Delay} &= (\text{RxTimestampB} - \text{TxTimestampF}) - (\text{TxTimestampB} - \text{RxTimestampF}) \\
 &= \text{RxTimestampB} - \text{TxTimestampF} - \text{TxTimestampB} + \text{RxTimestampF} \\
 &= (\text{RxTimestampF} - \text{TxTimestampF}) - (\text{TxTimestampB} - \text{RxTimestampB})
 \end{aligned}$$

Configure Ethernet Frame Delay Measurement for L2VPN Services

Perform the following tasks to configure Ethernet Frame Delay Measurement for L2VPN Services:

1. Configure L2VPN service.
2. Enable CFM service continuity check.
3. Enable CFM on the interface.
4. Configure Ethernet frame delay measurement.

```

/* Configure L2VPN service */
Router# configure
Router(config)# l2vpn
Router(config-l2vpn)# xconnect group evpn_vpws_203
Router(config-l2vpn-xc)# p2p evpn_vpws_phy-100
Router(config-l2vpn-xc-p2p)# interface GigabitEthernet0/0/0/2.100
Router(config-l2vpn-xc-p2p)# neighbor evpn evi 30001 target 30001 source 50001
Router(config-l2vpn-xc-p2p)# commit

```

```

/* Enable CFM service continuity check */
Router# ethernet cfm
Router(config-cfm)# domain xcup1 level 7 id null
Router(config-cfm-dmn)# service xcup1 xconnect group evpn_vpws_Bund
Router(config-cfm-dmn-svc)# mip auto-create all ccm-learning
Router(config-cfm-dmn-svc)# continuity-check interval 1s
Router(config-cfm-dmn-svc)# mep crosscheck
Router(config-cfm-dmn-svc)# mep-id 4001
Router(config-cfm-dmn-svc)# commit

/* Enable CFM on the interface */
Router(config)# interface GigabitEthernet0/0/0/2.100 12transport
Router(config-subif)# encapsulation dot1q 100
Router(config-subif)# rewrite ingress tag pop 1 symmetric
Router(config-subif)# mtu 9100
Router(config-subif)# ethernet cfm
Router(config-if-cfm)# mep domain bd-domain service bd-service mep-id 4001
Router(config-if-cfm-mep)# sla operation profile test-profile1 target mep-id 1112
Router(config-if-cfm-mep)# commit

/* Configure Ethernet frame delay measurement */
Router(config)# ethernet sla
Router(config-sla)# profile EVC-1 type cfm-delay-measurement
Router(config-sla-prof)# probe
Router(config-sla-prof-pb)# send packet every 1 seconds
Router(config-sla-prof-pb)# schedule
Router(config-sla-prof-schedule)# every 3 minutes for 120 seconds
Router(config-sla-prof-schedule)# statistics
Router(config-sla-prof-stat)# measure round-trip-delay
Router(config-sla-prof-stat-cfg)# buckets size 1 probes
Router(config-sla-prof-stat-cfg)# buckets archive 5
Router(config-sla-prof-stat-cfg)# commit

```

Running Configuration

This section shows the Ethernet frame delay measurement running configuration.

```

/* Configure L2VPN service */
l2vpn
xconnect group evpn_vpws_203
p2p evpn_vpws_phy-100
interface GigabitEthernet0/0/0/2.100
neighbor evpn evi 30001 target 30001 source 50001
!
/* Enable CFM service continuity check */
ethernet cfm
domain xcup1 level 7 id null
service xcup1 xconnect group evpn_vpws_Bundle_ether203 p2p evpn_vpws-100 id number 4001
mip auto-create all ccm-learning
continuity-check interval 1s
mep crosscheck
mep-id 4001
!
/* Enable CFM on the interface */
interface GigabitEthernet0/0/0/2.100 12transport
encapsulation dot1q 100
rewrite ingress tag pop 1 symmetric
mtu 9100
ethernet cfm
mep domain bd-domain service bd-service mep-id 4001
sla operation profile test-profile1 target mep-id 1112
!

```

```

/* Configure Ethernet SLA */
ethernet sla
profile EVC-1 type cfm-delay-measurement
probe
  send packet every 1 seconds
!
schedule
  every 3 minutes for 120 seconds
!
statistics
  measure round-trip-delay
  buckets size 1 probes
  buckets archive 5
!

```

Verification

Verify the frame delay measurement. In the following example, you observe that the sent and received DMM and DMR packets are same. So there is no delay in frame transmission.

```
Router# show ethernet cfm local meps interface GigabitEthernet0/0/0/2.100 verbose
```

```

Up MEP on GigabitEthernet0/0/0/2.100 MEP-ID 4001
=====
Interface state: Up      MAC address: 0c11.6752.3af8
Peer MEPs: 1 up, 0 with errors, 0 timed out (archived)

CCM generation enabled: Yes, 10s (Remote Defect detected: No)
AIS generation enabled: No
Sending AIS:            No
Receiving AIS:         No
Sending CSF:           No
Receiving CSF:        No

Packet      Sent      Received
-----
CCM          19        9 (out of seq: 0)
DMM          473        0
DMR          0         473

```

Minimum delay bin

Y.1731 Ethernet SLA is used to collect and optionally aggregate performance metrics over time. The data collected during a specific timeframe is organized into buckets. When aggregation is enabled, each bucket contains a set of bins, which helps reduce memory consumption.

With the minimum-delay bin feature, you can configure a distinct width for the first bin. This prevents the wastage of bins that might otherwise be empty due to the inherent speed of light delays. The remaining bins can then focus on capturing variations in observed delays.

Table 29: Feature History Table

Feature Name	Release Information	Feature Description
--------------	---------------------	---------------------

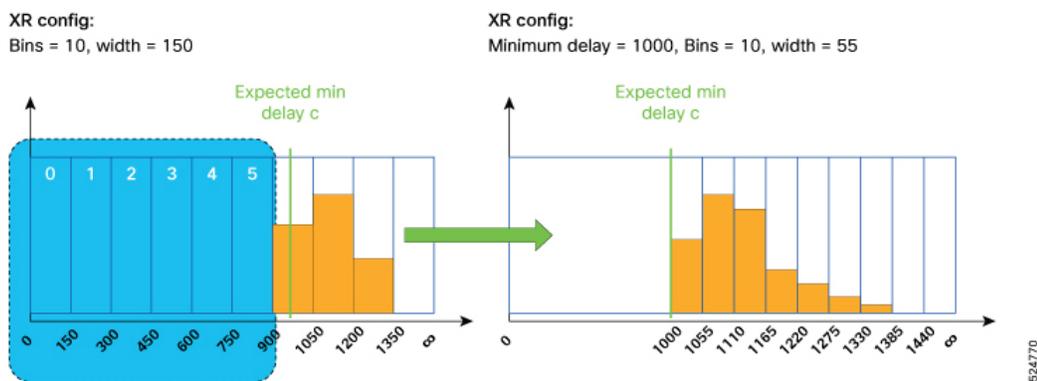
Minimum-delay bin	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: Q200, P100], 8700 [ASIC: P100, K100]); Centralized Systems (8600 [ASIC:Q200]); Modular Systems (8800 [LC ASIC: Q100, Q200, P100]).</p> <p>For statistics aggregation, you can now configure a distinct width for the first bin to adjust for large propagation delay. By using this feature, you can avoid wasting several bins that would be empty in some unavoidable situations such as delay due to speed of light limitations.</p> <p>The feature introduces these changes:</p> <p>CLI:</p> <ul style="list-style-type: none"> • The minimum-delay keyword is introduced in the aggregate and cfm-delay-measurement probe commands. <p>YANG Data Models: New XPathS for</p> <ul style="list-style-type: none"> • Cisco-IOS-XR-infra-sla-cfg.yang • Cisco-IOS-XR-um-ethernet-sla-cfg.yang • Cisco-IOS-XR-infra-sla-oper.yang <p>(see GitHub, YANG Data Models Navigator)</p>
-------------------	----------------	--

Information about minimum-delay bin

In situations where a delay is expected in first few iterations, you can specify the width of the first bin independently using the **minimum-delay** keyword in statistics aggregation configuration. See below examples to configure aggregation with and without minimum-delay support.

The below figure shows the comparison between statistics aggregation before and after minimum-delay configuration.

Figure 8: Comparison diagram before and after minimum-delay configuration



Example 1: Configuring aggregation without minimum-delay

```
statistics measure round-trip-delay aggregate bins 10 width 150
```

This configuration has 10 bins and the results being aggregated into the ranges 0-150, 150-300, 300-450 and so on, until 1350+, where the last bin has infinite width to hold all values greater than 1350.

Example 2: Configuring aggregation with minimum-delay

```
statistics measure round-trip-delay aggregate bins 10 width 55 minimum-delay 1000
```

Here, the width of the first bin is 1000ms and not 150ms. The width of the other nine bins are aggregated into 1000-1055, 1055-1110, 1110-1165, and so on. This leads to increased resolution with the same number of bins, as all the bins are utilized.



Note To specify the values of width and minimum-delay in microseconds instead of milliseconds, you must use the **usec** keyword. For more information, see [aggregate](#) and [cfm-delay-measurement probe](#) commands.

Configure minimum delay bin support

Before you begin

Make sure that the number of bins for aggregate configuration is at least two.

Procedure

Step 1 Configure an SLA Operation profile and statistics measurement for the Profile.

Example:

Configure Ethernet Frame Delay Measurement for L2VPN Services.

```
Router(config)# ethernet sla
Router(config-sla)# profile EVC-1 type cfm-delay-measurement
Router(config-sla-prof)# probe
Router(config-sla-prof-pb)# send packet every 1 seconds
Router(config-sla-prof-pb)# schedule
Router(config-sla-prof-schedule)# every 3 minutes for 120 seconds
Router(config-sla-prof-schedule)# statistics
Router(config-sla-prof-stat)# measure round-trip-delay
Router(config-sla-prof-stat-cfg)# buckets size 1 probes
Router(config-sla-prof-stat-cfg)# buckets archive 5
Router(config-sla-prof-stat-cfg)# commit
```

Step 2 Configure aggregation for the SLA profile and then configure the width of the first bin by using the **minimum-delay** keyword.

Example:

Configure aggregation with bin count of 10 and the width of the first bin as 5 ms.

```
Router(config-sla-prof-schedule)# statistics
Router(config-sla-prof-stat)# measure round-trip-delay
Router(config-sla-prof-stat-cfg)# aggregate bins 5 width 10 minimum-delay 30
Router(config-sla-prof-stat-cfg)# buckets size 1 probes
Router(config-sla-prof-stat-cfg)# buckets archive 5
Router(config-sla-prof-stat-cfg)# commit
```

Step 3 View the running configuration using the **show running-config** command.

Example:

```

ethernet sla
profile EVC-1 type cfm-delay-measurement
probe
  send packet every 1 seconds
!
schedule
  every 3 minutes for 120 seconds
!
statistics
measure round-trip-delay
  aggregate bins 10 width 2 minimum-delay 5
  buckets size 1 probes
  buckets archive 5
!

```

Step 4 Verification

Verify the output by using the below show commands.

- **show protocolsla operations detail**
- **show protocolsla probes**
- **show protocolsla statistics**

Example:

Use the below show command to verify the output.

```
router# show ethernet sla statistics history detail on-demand
```

Below is a sample output.

```

Bucket started at 15:38 on Tue 02 Jul 2024, lasting 1 hour:
Pkts sent: 1200; ...
Result count: 30
Min: 13ms; Max: 154ms; Mean: 28ms; StdDev: 11ms
Bins:
Range           Samples      Cum. Count      Mean
-----
0 to 30 ms      20 (2%)       20 (2%)         37ms
30 to 35 ms     909 (61%)     929 (77%)       67ms
35 to 40 ms     212 (18%)     1141 (95%)      75ms
40 to 45 ms     98 (11%)      1141 (95%)      75ms
> 45 ms        55 (5%)       1196 (85%)      90ms

```

Link loss forwarding

Link loss forwarding (LLF) is a mechanism used in networking to propagate the status of a network link to other connected devices. When a link experiences a failure or goes down, LLF ensures that this information is forwarded to other network devices, which can then take appropriate actions to maintain network stability and performance.

Table 30: Feature History Table

Feature Name	Release Information	Feature Description
Link loss forwarding	Release 25.4.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100], 8700 [ASIC: K100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8011-12G12X4Y-A • 8011-12G12X4Y-D • 8711-48Z-M
Link loss forwarding	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is now supported on Cisco 8011-4G24Y4H-I routers.</p>
Link loss forwarding	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100, K100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 8712-MOD-M • 88-LC1-12TH24FH-E • 88-LC1-36EH+A8:B12 • 88-LC1-52Y8H-EM
Link loss forwarding	Release 7.9.1	<p>We have now enabled high availability between two bridged interfaces by disabling both interfaces if any one of them fails. Such high availability is enabled because the functionality allows a fault detected on one side of a CFM-protected network to propagate to the other, allowing the device to re-route around the failure.</p> <p>In earlier releases, a failure on one bridged interface did not disable the other interface, and connected devices remained unaware of the link loss.</p> <p>The feature introduces these changes:</p> <ul style="list-style-type: none"> • CLI: New propagate-remote-status command • YANG Data Model: New XPath for Cisco-IOS-XR-um-ethernet-cfm-cfg.yang (see GitHub, YANG Data Models Navigator)

You can enable LLF on a network by one of the following methods:

- **Link State Monitor and Propagation by CFM:** LLF uses Connectivity Fault Management (CFM) to transmit notification of a signal loss or fault across the network. When there is a fault on a link to a device on one side of the network, the connection to the port on the other side needs to be shutdown so that the device re-routes the traffic.
- **Remote Link State Propagation:** LLF uses this method for Layer 2 transport events to propagate link failures to remote endpoints. When a link failure occurs, LLF ensures that the failure is communicated to other devices in the network. This enables the other devices to take appropriate action, such as rerouting traffic or triggering failover mechanisms.

Link state monitor and propagation by CFM

Link State Monitoring involves tracking the status of network links to ensure they are operational and performing as expected. This can include monitoring for link failures, degradations, or other issues that might affect network performance. When a link state changes, this information needs to be propagated throughout the network so that other devices can adjust their routing tables and network operations accordingly.

When there is a fault on a link to a device on one side of the network, the connection to the port on the other side needs to be shutdown so that the device re-routes the traffic. This requires the interface to be TX-disabled.

Link Loss Forwarding (LLF) uses Connectivity Fault Management (CFM) to transmit notification of a signal loss or fault across the network. If a local attachment circuit (AC) on a bridged interface fails, one of the following signals or packet types are sent to the neighboring device:

- **Continuity Check Message (CCM)** – The CCMs are heartbeat messages exchanged periodically between all the Maintenance End Points (MEPs) in a service. MEPs are members of a particular service within a domain and are responsible for sourcing and sinking CFM frames. Each MEP sends out multicast CCMs, and receives CCMs from all the other MEPs in the service. This allows each MEP to discover its peer MEPs, and to verify that there is connectivity between them.
- **Alarm Indication Signal (AIS)** – These are messages sent periodically by MEPs that have detected a fault, to the MEPs in the next highest maintenance domain level.
- **Client Signal Fail (CSF)** – A mechanism for error detection. When a MEP detects an issue, the MEP sends CSF packets to its peer MEPs.

For more information on MEPs, see [Maintenance Points, on page 104](#).

Connectivity Fault Management Daemon (CFMD) and Ether-MA are processes that run on the control plane of the router. Ether-MA handles owner channel communication and resyncs from CFMD, L2VPN, and other Ether MA processes. This module handles the TX-disable and TX-enable events, based on the notifications from CFMD.

When the system receives a CCM or AIS with fault indication, or a CSF error packet, CFMD communicates with Ether-MA to TX-disable the interface.

When an interface receives a fault notification, the transitions are handled as follows:

- The interface is transitioned to TX-disable state.
- A restore or damping timer either with a 3.5 * packet interval duration is started.
- If no other fault packets are received after the restore timer ends, the TX-disable state is cleared and the interface is transitioned to TX-enable state.

Restrictions for link loss forwarding for CFM

- Link loss forwarding is not permitted on subinterfaces.
- Link loss forwarding is permitted only on UP MEPs. The UP MEPs send the frames into the bridge relay function and not through the wire connected to the port where the MEP is configured. For more information on UP MEPs, see [MEP and CFM Processing Overview](#), on page 104.
- A damping or restore timer governs transitions of an interface from TX-disabled state to TX-enabled state. By default, the period of the damping timer is three times the configured CCM interval.
- The damping timer is not provided for transitions of an interface from TX-enabled state to TX-disabled state.
- Link loss forwarding does not work on bundle interfaces configured with LACP.

Configure link loss forwarding for CFM

To configure LLF on a network:

1. Configure a Connectivity Fault Management (CFM) domain and service.
2. Configure a Maintenance End Point (MEP) under the CFM domain and service.
3. Configure continuity check message (CCM) interval on the MEP. By default, the restore timer for a CCM notification is calculated based on the configured CCM interval.
4. Configure Client Signal Fail (CSF) transmission on the MEP, to enable CSF transmission.
5. Configure CSF logging on the MEP, to enable logging on receiving a CSF packet.



Note The CSF configuration is optional and is not required when both the devices in CFM-protected network are running with IOS-XR. This configuration is required for inter-operation with certain client-end setups that contain devices from other clients.

6. Enable LLF on an interface using the **propagate-remote-status** command. This command triggers the interface to be TX-disabled on fault detection.

Configuration Example

```
/* Configure CFM domain, service, and MEP */
Router# configure
Router(config)# ethernet cfm
Router(config-cfm)# domain dom1 level 1 service ser1 bridge group up-meps bridge-domain
up-mep

/* Configure CCM interval */

Router(config-cfm-dmn-svc)# continuity-check interval 1m

/* (Optional) Configure CSF */
Router(config-cfm-dmn-svc)# csf interval 1m cos 4
Router(config-cfm-dmn-svc)# csf-logging
Router(config-cfm-dmn-svc)# commit
```

```

/* Enable LLF on an interface */

Router# configure
Router(config)# interface GigabitEthernet0/2/0/0
Router(config-if)# ethernet cfm
Router(config-if-cfm)# mep domain dom1 service ser1 mep-id 1
Router(config-if-cfm-mep)# propagate-remote-status
Router(config-if-cfm-mep)# commit

```

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 restore-timer 5
  !
!
!

```

Verification

The following output shows LLF configuration and fault state for each interface:

```

Router# show ethernet cfm interfaces llf location 0/RP0/CPU0
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                                       5s Not running

```

The following output shows that the interface received a single CSF packet at 1 minute interval, so that the interface is TX-disabled with a damping timer of 3.5 minutes.

```

Router# show ethernet cfm local meps detail
Domain dom1 (Level 1), Service ser1

```

```
UP MEP on GigabitEthernet0/1/0/0 MEP-ID 1
```

```
=====
Interface state: UP      MAC address: 0204.3dbe.c93b
Peer MEPS: 0 up, 0 with errors, 0 timed out (archived)
Cross-check errors: 0 missing, 0 unexpected

CCM generation enabled: No
AIS generation enabled: No
Sending AIS:             No
Receiving AIS:          No
Sending CSF:             No
Receiving CSF:         Yes (Interval: 1min, started 00:03:29 ago)
TX Disable triggered: Yes (restore timer not running)
```

The following output shows that the interface received a CCM notification that the peer MEP port is down, so that the interface is TX-disabled.

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

```
Domain dom1 (Level 1), Service ser1
UP MEP on GigabitEthernet0/1/0/0 MEP-ID 1
```

```
=====
Interface state: UP      MAC address: 0204.3dbe.c93b
Peer MEPS: 1 up, 1 with errors, 0 timed out (archived)
Cross-check errors: 0 missing, 0 unexpected

CCM generation enabled: Yes, 1min (Remote Defect detected: Yes)
CCM defects detected:   P - peer port down
AIS generation enabled: No
Sending AIS:             No
Receiving AIS:          No
Sending CSF:             No
Receiving CSF:          No
TX Disable triggered:  Yes (restore timer 5s not running)
```

The following output shows that the interface received CCM notification that the peer MEP port is up, and restore timer is started for the TX-disabled interface.

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

```
Domain dom1 (Level 1), Service ser1
UP MEP on GigabitEthernet0/1/0/0 MEP-ID 1
```

```
=====
Interface state: UP      MAC address: 0204.3dbe.c93b
Peer MEPS: 1 up, 0 with errors, 0 timed out (archived)
Cross-check errors: 0 missing, 0 unexpected

CCM generation enabled: Yes, 1min (Remote Defect detected: No)
AIS generation enabled: No
Sending AIS:             No
Receiving AIS:          No
Sending CSF:             No
Receiving CSF:          No
TX Disable triggered:  Yes (restore timer running, 1183ms remaining)
```

The following output shows Ether-MA configured bundles and their members:

```
Router# show ethernet infra internal ether-ma bundles
```

```
Bundle interface: Bundle-Ether1 (TX disabled)
```

```
Bundle members:
```

```
GigabitEthernet0/1/0/1
GigabitEthernet0/1/0/2
```

```
Bundle interface: Bundle-Ether2
```

```
Bundle members:
```

```
GigabitEthernet0/2/0/1
```

Remote link state propagation

Table 31: Feature History Table

Feature Name	Release Information	Feature Description
Remote link state propagation	Release 25.4.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100]) (select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8711-48Z-M • 8011-12G12X4Y-A • 8011-12G12X4Y-D
Remote link state propagation	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])</p> <p>This feature is now supported on:</p> <ul style="list-style-type: none"> • 8712-MOD-M • 8011-4G24Y4H-I
Remote Link State Propagation	Release 24.4.1	<p>Introduced in this release on: Fixed Systems(8200, 8700);Modular Systems (8800 [LC ASIC: P100]) (select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-32FH-M • 8711-32FH-M • 88-LC1-12TH24FH-E

Feature Name	Release Information	Feature Description
Remote link state propagation	Release 24.3.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: Q200, P100], 8700 [ASIC: P100]); Centralized Systems (8600 [ASIC:Q200]); Modular Systems (8800 [LC ASIC: Q100, Q200, P100])</p> <p>Remote Link State Propagation allows the status of a link to be communicated to remote devices, ensuring that all relevant parts of the network are aware of link state changes. Link Loss Forwarding (LLF) uses this feature to propagate link failures to remote endpoints.</p> <p>By enabling remote state propagation and LLF on an interface, you can ensure that the link state changes are communicated to remote devices, allowing for quick failover and rerouting of traffic.</p> <p>This feature introduces the propagate remote-status command.</p>

Remote Link State Propagation allows the status of a link to be communicated to remote devices, ensuring that all relevant parts of the network are aware of link state changes. This is particularly useful in Layer 2 transport networks, where maintaining accurate link status information is crucial for network performance and reliability.

Link Loss Forwarding for Layer 2 Transport

Link Loss Forwarding (LLF) uses Remote Link State Propagation to propagate link failures to remote endpoints. When a link failure occurs, LLF ensures that the failure is communicated to other devices in the network, allowing them to take appropriate action, such as rerouting traffic or triggering failover mechanisms. LLF helps avoid packet loss and triggers network convergence through alternate links. It works by sending signals across the pseudowire (PW) to the neighboring device, bringing the PW and far-end attachment circuit (AC) down if the local AC goes down.

You can configure LLF for the Layer 2 transport events, using the **propagate remote-status** command. You can enable LLF on the following interface types: 1G, 10G, 25G, 40G, 100G, and 400G.

Remote State Propagation with LLF for L2 Transport

When you configure LLF for L2 transport events, the following are the processes that happen:

1. **Link State Detection:** The network device detects a change in the link state, such as a link going down.

- 2. Remote State Propagation:** When remote state propagation is enabled, the detected link state change is propagated to remote devices. This ensures that all relevant devices are aware of the link failure and can take appropriate action.
- 3. LLF Activation:** LLF uses the propagated link state information to trigger failover mechanisms or reroute traffic. This helps maintain network performance and reliability by quickly responding to link failures.

When you enable remote state propagation and LLF on an interface, the link state changes are communicated to remote devices allowing for quick failover and rerouting of traffic.

Configure Link Loss Forwarding for Layer 2 Transport

The following example shows how to configure LLF for Layer 2 transport events.

Procedure

- Step 1** To enable link loss forwarding for Layer transport events, enter the interface mode, configure **l2transport**, and then enable LLF using the **propagate remote-status** command.

```
Router(config)# interface tenGigE 0/0/0/1
Router(config-if)# l2transport
Router(config-if-l2)# propagate remote-status
Router(config-if-l2)# commit
```

- Step 2** View the running configuration.

```
interface TenGigE 0/0/0/1
  l2transport
    propagate remote-status
  !
```



CHAPTER 7

Configure Integrated Routing and Bridging

This module describes the configuration of Integrated Routing and Bridging (IRB) on the Cisco 8000 Series Routers. IRB provides the ability to exchange traffic between bridging services on the Cisco 8000 Series Router and a routed interface using a Bridge-Group Virtual Interface (BVI).

Table 32: Feature History Table

Feature Name	Release Information	Feature Description
Integrated Routing and Bridging (IRB), and IRB with Bridge-Group Virtual Interface	Release 25.4.1	Introduced in this release on: Fixed Systems (8700 [ASIC: K100])(select variants only*) * This feature is supported on: <ul style="list-style-type: none">• 8712-MOD-M• 8711-48Z-M

Feature Name	Release Information	Feature Description
Integrated Routing and Bridging with Bridge-Group Virtual Interface	Release 25.2.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100]); Fixed Systems (8700 [ASIC: P100]); Modular Systems (8800 [LC ASIC: P100])</p> <p>Integrated Routing and Bridging (IRB) provides the ability to exchange traffic between bridging services on a router and a routed interface using a Bridge-Group Virtual Interface (BVI). IRB over BVI enables seamless communication between Layer 2 bridging and Layer 3 routing by using BVI as a logical interface.</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 88-LC1-36EH • 8711-32FH-M
Integrated Routing and Bridging (IRB)	Release 25.1.1	<p>Introduced in this release on: Modular Systems (8800 [LC ASIC: P100]) (select variants only*)</p> <p>Integrated Routing and Bridging (IRB) provides the ability to exchange traffic between bridging services on a router and a routed interface using a Bridge-Group Virtual Interface (BVI). IRB over BVI enables seamless communication between Layer 2 bridging and Layer 3 routing by using BVI as a logical interface.</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 88-LC1-12TH24FH-E • 88-LC1-52Y8H-EM

- [Prerequisites to configure IRB, on page 167](#)
- [Guidelines and restrictions to configure IRB, on page 167](#)
- [Information about configuring IRB, on page 168](#)
- [How to configure IRB, on page 173](#)
- [Configuration examples for IRB, on page 179](#)

Prerequisites to configure 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:

- Confirm that you are configuring the required line cards where you plan to support IRB in support of both Layer 3 to Layer 2 traffic flows and Layer 2 to Layer 3 traffic flows
- Know the IP addressing and other Layer 3 information to be configured on the bridge virtual interface (BVI). For more information, see the [“Restrictions for Configuring IRB” section on page 265](#).
- 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.

Guidelines and restrictions to configure 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.
- The following areas are *not* supported on the BVI:
 - IP fast reroute (FRR)
 - MoFRR
 - Traffic mirroring
 - Unnumbered interface for BVI
- Multi protocol Label Switching (MPLS) on BVI is supported.
- PIM snooping. (Need to use selective flood.)
- VRF-aware DHCP relay is supported.
- BVIs are supported only on bridge domains with the following characteristics:
 - The bridge domain supports single and double-tagged dot1q- and dot1ad-encapsulated EFPs with non-ambiguous or “exact match” EFP encapsulations. Single and double-tagged encapsulation can be specified as long as the **rewrite ingress tag pop symmetric** command is configured.
 - All Layer 2 tags must be removed. VLAN ranges are not supported.
 - Untagged EFPs are supported.

- To use the sub-interface configurations **encapsulation dot1ad** (or **encapsulation dot1q**) and **encapsulation dot1ad second-dot1q any** (or **encapsulation dot1q second-dot1q any**) together on the same physical interface, use the **exact** keyword as shown below. Else, it results in traffic loss.

```
Router(config)# interface hundredGigE 0/0/0/0.0
Router(config-subif)# encapsulation dot1ad 200 exact

Router(config)# interface hundredGigE 0/0/0/0.1
Router(config-subif)# encapsulation dot1ad 200 dot1q any
Router(config-subif)# commit
```

- QoS on BVI is not supported on Cisco 8000 routers. Any QoS policy configuration on BVI will be rejected.

Information about configuring IRB

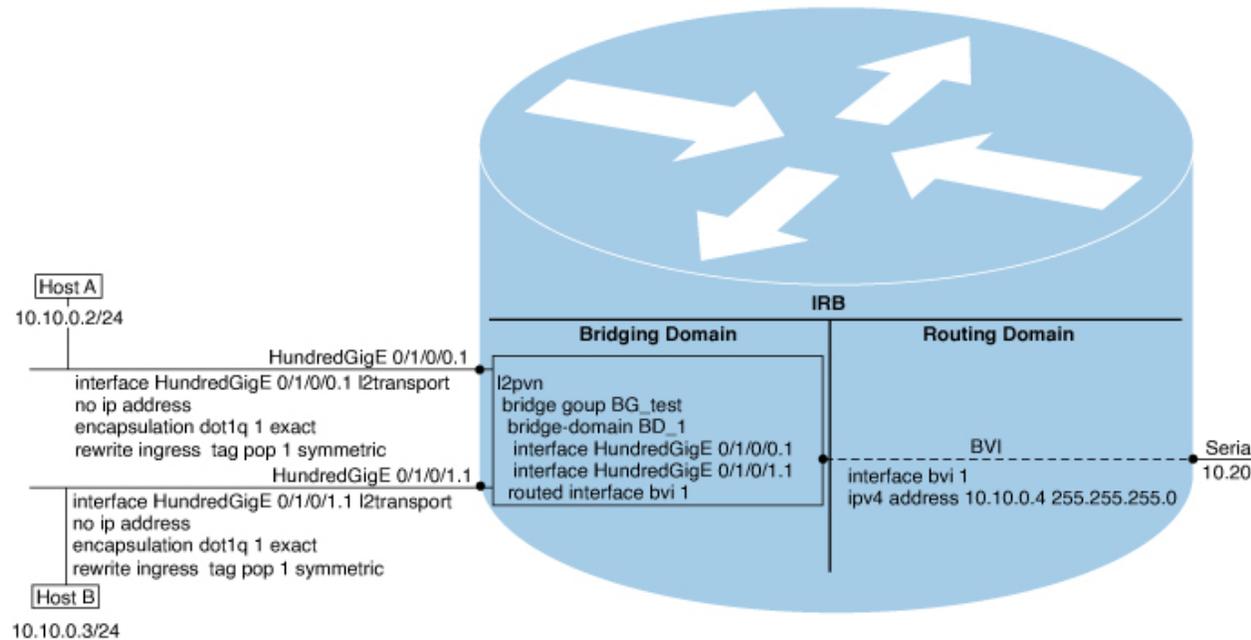
This section includes the following topics:

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.

In software releases before Cisco IOS XR 4.0.1 where IRB is not supported, you would need to implement a physical cabling solution to connect the egress Layer 2 bridge domain interface to a Layer 3 routing domain interface on the same Cisco 8000 Series Router. In Cisco IOS XR Release 4.0.1, IRB accomplishes the same functionality using a BVI and its supporting interface and bridge group configuration shown in this figure.

Figure 9: IRB Functional View and Configuration Elements



Bridge-Group Virtual Interface

This section includes the following information:

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.

Aside from supporting a configurable MAC address, a 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 BVI

The following are the supported features on BVI:

- Border Gateway Protocol (BGP)
- Bidirectional Forwarding Detection (BFD)
- Open Shortest Path First (OSPF)
- Integrated Intermediate System-to-Intermediate System (IS-IS)
- Netflow
- Access Control Lists

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**
- **mtu** (The default is 1500 bytes)
- **shutdown**
- The BVI supports IP helper addressing and secondary IP addressing.

BVI MAC address

By default, the Cisco 8000 Series Router uses one MAC address for all BVI interfaces on the router. However, this means that the MAC address is not unique globally. If you want to override the default and specify a unique MAC address at the BVI, then you can configure it at the BVI interface.

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 pseudowire is up, even if all EFPs are down.

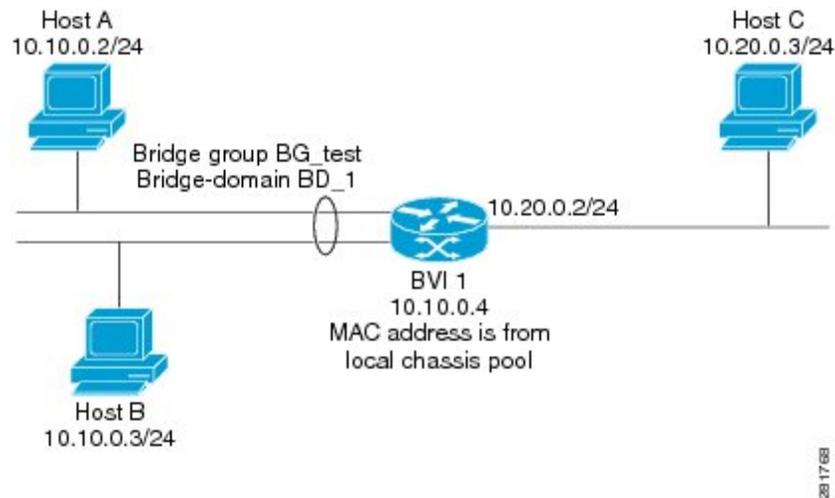
- The following characteristics determine when the the BVI line protocol state is up:
 - The bridge-domain is in Up state.

- The BVI IP address is not in conflict with any other IP address on another active interface in the router.

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 10: 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 any ARP request the 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.

Supported environments for IRB

These environments and configuration elements are supported with IRB on the Cisco 8000 Series Router:

- Configuration of one BVI per bridge domain.
- Virtual Private LAN Service (VPLS) virtual forwarding instance (VFI) configuration associated with a bridge domain configured with a BVI.
- BGP PIC edge for BVI-based prefixes.

- Traffic forwarding for the BVI using Open Shortest Path First (OSPF), Intermediate System-to-Intermediate System (IS-IS), and Border Gateway Protocol (BGP).
- Internet Group Management Protocol (IGMP) static groups.
- Dynamic Host Configuration Protocol (DHCP) relay agent. When DHCP relay is used from an aggregation node to obtain an IP address, the default gateway will be the IP address configured on the BVI. The BVI IP address should be in a common subnet as the DHCP pool that is being used by the aggregation node to assign IP addresses.
- Virtual Router Redundancy Protocol (VRRP) configuration and priority.
- Hot Standby Router Protocol (HSRP).
- Up to 255 VRRP or HSRP virtual MAC addresses can be configured on a physical or bundle interface, with each VRRP or HSRP ID assigned to only one BVI interface.
- Bridging of non-IP packets on a bridge domain configured with a BVI.
- Parity with stateful protocol support as currently supported on Layer 3 subinterfaces on the Cisco 8000 Series Router.
- IP SLA support as currently supported on Layer 3 subinterfaces on the Cisco 8000 Series Router.
- Load balancing of BVIs as ECMP paths (up to 32 paths).
- Interface-MIB.
- Packet counters for BVI interfaces.

The following sections document additional IPv4- and IPv6-specific environments supported for IRB.

Additional IPv4-specific environments supported for IRB

- Layer 3 IP multicast, with ability to take ingress IP multicast traffic and bridge it to multiple Layer 2 subinterfaces (Ethernet flow points) on a bridge domain that are part of multicast groups.
- VRFs for IPv4 (Per-VPN label VRFs only—not per prefix).

Additional IPv6-specific environments supported for IRB

Cisco IPv6 Provider Edge Router over MPLS (6PE) and IPv6 VPN Provider Edge (6VPE) support with BVI interfaces at the customer edge (CE)-facing side of the Cisco 8000 Series Router as the PE device with the following restriction:

Only per-VRF label allocation is supported (using the **label-allocation-mode per-vrf** command). For a configuration example, see the [“6PE/6VPE With BVI Configuration: Example”](#) section on page 282

How to configure IRB

This section includes the following configuration tasks:

Configure BVI

To configure a BVI, complete the following steps.

Configuration guidelines

Consider the following guidelines when configuring the BVI:

- The BVI must be assigned to 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.

Procedure

Step 1 Run the **configure** command to enter the global configuration mode.

Example:

```
Router#configure
```

Step 2 Run the **interface bvi identifier** command to specify and create a BVI, where *identifier* is a number from 1–65535.

Example:

```
Router(config)#interface bvi 1
```

Step 3 Specify a primary or secondary IPv4 address or an IPv6 address for the interface.

Example:

```
Router(config-if)#ipv4 address 10.10.0.4 255.255.255.0
```

Step 4 (Optional) Specify the amount of time (in seconds) to delay purging of Address Resolution Protocol (ARP) table entries when the interface goes down.

Example:

```
Router(config-if)#arp purge-delay 120
```

The range is 1–65535. By default, purge delay is not configured.

Step 5 (Optional) Run the **arp timeout seconds** command to configure the duration that the dynamic entries learned on the interface remain in the ARP cache.

Example:

```
Router(config-if)#arp timeout 12200
```

The range is 30–2144448000 seconds. The default value is 14,400 seconds (4 hours).

Step 6 (Optional) Specify the amount of bandwidth (in kilobits per second) to be allocated on the interface.

Example:

```
Router(config-if)#bandwidth 1000000
```

This bandwidth number is used as the cost metric in routing protocols for the BVI. The range is 0–4294967295. The default value is 10000000 (10 Gbps).

Step 7 (Optional) Specify the 48-bit MAC address for the BVI as three dotted-hexadecimal values, which override the use of the default MAC address.

Example:

```
Router(config-if)#mac-address 1111.2222.3333
```

The range for each value is 0000 to ffff. A MAC address of all 0s is not supported.

Step 8 (Optional) Run the `mtu bytes` command to configure the maximum transmission unit (MTU) size for packets on the interface.

Example:

```
Router(config-if)# mtu 2000
```

The range is 64–65535. The default value is 1514.

Step 9 Run the `end` or `commit` commands to save the configuration changes.

Example:

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

or

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

- 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.

Configure Layer 2 AC interfaces

To configure the Layer 2 AC interfaces for routing by a BVI, complete the following steps.

Before you begin

The interfaces to be configured as Layer 2 ACs in the bridge domain and routed by a BVI must be located on the Cisco 8000 Series Router.

Procedure

Step 1 Run the **configure** command to enter the global configuration mode.

Example:

```
Router#configure
```

Step 2 Run the **interface [HundredGigE] interface-path-id[.subinterface] l2transport** command to enable Layer 2 transport mode on a HundredGigE interface or subinterface and enter the interface or subinterface configuration mode.

Example:

```
Router(config)#interface HundredGigE 0/1/0/0.1 l2transport
```

The *interface-path-id* is specified as the *rack/slot/module/port* location of the interface and *.subinterface* is the optional subinterface number.

Step 3 (Optional) Run the **encapsulation dot1q vlan-id [exact]orencapsulation dot1ad vlan-id dot1q vlan-id** command to configure the IEEE 802.1Q encapsulation on the specified VLAN only.

Example:

```
Router(config-if)#encapsulation dot1q 1 exact
```

Step 4 (Required if VLAN tagging configured) Specify the one or two tags (depending on the network configuration) that should be removed from frames arriving at the ingress interface to the bridge domain.

Example:

```
Router(config-if)#rewrite ingress tag pop 1 symmetric
```

Note

When you configure double tags using dot1ad and dot1q encapsulation, you must use the **rewrite ingress tag pop 2 symmetric** command.

Step 5 Run the **end** or **commit** commands to save the configuration changes.

Example:

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

or

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

- 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.

Configure a bridge group and assign interfaces to a bridge domain

To configure a bridge group and assign interfaces to a bridge domain, complete the following steps.

Procedure

Step 1 In the global configuration mode, run the **l2vpn** command to enter the L2VPN configuration mode.

Example:

```
Router (config) #l2vpn
```

Step 2 Run the **bridge group** *bridge-group-name* command to create a bridge group and enter the L2VPN bridge group configuration mode.

Example:

```
Router (config-l2vpn) #bridge group 10
```

Step 3 Run the **bridge-domain** *bridge-domain-name* command to create a bridge domain and enter L2VPN bridge group bridge domain configuration mode.

Example:

```
Router (config-l2vpn-bg) #bridge-domain BD_1
```

Step 4 Associate a HundredGigE interface with the specified bridge domain and enter the L2VPN bridge group bridge domain attachment circuit configuration mode.

Example:

```
Router (config-l2vpn-bg-bd) #interface HundredGigE 0/1/0/0.1
```

The *interface-path-id* is specified as the *rack/slot/module/port* location of the interface and *.subinterface* is the optional subinterface number.

Repeat this step for as many interfaces as you want to associate with the bridge domain.

Step 5 Run the **end** or **commit** commands to save the configuration changes.

Example:

```
Router (config-l2vpn-bg-bd-ac) #end
```

or

```
Router (config-l2vpn-bg-bd-ac) #commit
```

- 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.

Associate BVI as a routed Interface on a bridge domain

To associate the BVI as the routed interface on a bridge domain, complete the following steps.

Procedure

Step 1 In the global configuration mode, run the **l2vpn** command to enter the L2VPN configuration mode.

Example:

```
Router(config)#l2vpn
```

Step 2 Create a bridge group and enter the L2VPN bridge group configuration mode.

Example:

```
Router(config-l2vpn)#bridge group BG_test
```

Step 3 Run the **bridge-domain** *bridge-domain-name* command to create a bridge domain and enter the L2VPN bridge group bridge domain configuration mode.

Example:

```
Router(config-l2vpn-bg)#bridge-domain 1
```

Step 4 Associate the specified BVI as the routed interface for the interfaces that are assigned to the bridge domain.

Example:

```
Router(config-l2vpn-bg-bd)# routed interface bvi 1
```

Step 5 Run the **end** or **commit** commands to save the configuration.

Example:

```
Router(config-l2vpn-bg-bd)#end
```

or

```
Router(config-l2vpn-bg-bd)#commit
```

- 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.

Display information about BVI

To display information about the BVI status and packet counters, use the following commands:

Command	Purpose
show interfaces bvi <i>identifier</i> [accounting brief description detail]	Displays interface status, line protocol state, and packet counters for the specified BVI.
show adjacency bvi <i>identifier</i> [detail remote]	Displays packet and byte transmit counters per adjacency to the specified BVI.
show l2vpn bridge-domain detail	Displays the reason that a BVI is down.

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
!
Router#configure
Router(config)#interface bvi 1
Router(config-if)#ipv4 address 10.10.0.4 255.255.255.0
Router(config-if)#exit
!
! Configure the Layer 2 AC interface
!
Router(config)#interface HundredGigE 0/1/0/0 l2transport
Router(config-if)#exit
```

```

!
! Configure the L2VPN bridge group and bridge domain and assign interfaces
!
Router(config)#l2vpn
Router(config-l2vpn)#bridge group 10
Router(config-l2vpn-bg)#bridge-domain 1
Router(config-l2vpn-bg-bd)#interface HundredGigE 0/1/0/0
Router(config-l2vpn-bg-bd-if)#exit
!
! Associate a BVI to the bridge domain
!
Router(config-l2vpn-bg-bd)#routed interface bvi 1
Router(config-l2vpn-bg-bd)#commit

```

IRB using ACs with VLANs: example

The following example shows how to configure IRB on a bridge domain with Layer 2 ACs using 802.1Q-encapsulated VLANs:

```

! Configure the BVI and its IPv4 address
!
Router#configure
Router(config)#interface bvi 1
Router(config-if)#ipv4 address 10.10.0.4 255.255.255.0
Router(config-if)#exit
!
! Configure the Layer 2 AC interfaces using dot1q encapsulation on a VLAN
!
Router(config)#interface HundredGigE 0/1/0/0.1 l2transport
Router(config-if)#no ip address
Router(config-if)#encapsulation dot1q 1 exact
Router(config-if)#rewrite ingress tag pop 1 symmetric
Router(config-if)#exit
Router(config)#interface HundredGigE 0/1/0/1.1 l2transport
Router(config-if)#no ip address
Router(config-if)#encapsulation dot1q 1 exact
Router(config-if)#rewrite ingress tag pop 1 symmetric
Router(config-if)#exit
!
! Configure the L2VPN bridge group and bridge domain and assign interfaces
!

Router(config)#l2vpn
Router(config-l2vpn)#bridge group 10
Router(config-l2vpn-bg)#bridge-domain 1
Router(config-l2vpn-bg-bd)#interface HundredGigE 0/1/0/0.1
Router(config-l2vpn-bg-bd)#interface HundredGigE 0/1/0/1.1
Router(config-l2vpn-bg-bd-if)#exit
!
! Associate a BVI to the bridge domain
!
Router(config-l2vpn-bg-bd)#routed interface bvi 1
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:

```
Router#configure
Router(config)#interface bvi 1
Router(config-if)#ipv4 address 10.10.10.4 255.255.255.0
Router(config-if)#ipv4 address 10.20.20.4 255.255.255.0 secondary
Router(config-if)#ipv4 address 10.30.30.4 255.255.255.0 secondary
Router(config-if)#commit
```

Comprehensive IRB configuration with BVI bundle interfaces and multicast configuration: example

The following example shows a more comprehensive router configuration with IRB and BVI multicast support:

```
interface Bundle-Ether25
  ipv4 address 10.21.0.2 255.255.255.0
  !
interface Loopback0
  ipv4 address 10.5.5.5 255.255.255.255
  !
interface HundredGigE0/0/0/1
  negotiation auto
  !
interface HundredGigE0/0/0/1.1 l2transport
  encapsulation dot1q 1
  rewrite ingress tag pop 1 symmetric
  !
interface HundredGigE0/0/0/1.2 l2transport
  encapsulation dot1q 2
  rewrite ingress tag pop 1 symmetric
  !

interface HundredGigE0/0/0/9
  bundle id 25 mode active
  !
interface HundredGigE0/0/0/19
  bundle id 25 mode active
  !
interface HundredGigE0/0/0/29
  bundle id 25 mode active
  !

interface HundredGigE0/0/0/39
  bundle id 25 mode active

interface BVI1
  ipv4 address 10.1.1.1 255.255.255.0
  !
interface BVI2
  ipv4 address 10.1.2.1 255.255.255.0

router ospf 100
  router-id 10.5.5.5
  area 0
```

```

interface Bundle-Ether25
  interface Loopback0
  interface BVI1
  interface BVI2
!
l2vpn
bridge group IRB
bridge-domain IRB1
  igmp snooping profile IRB_SNOOP
  interface HundredGigE0/0/0/1.1
  !
  routed interface BVI1
  !
bridge-domain IRB2
  igmp snooping profile IRB_SNOOP
  interface HundredGigE0/0/0/1.2
  !
  routed interface BVI2

multicast-routing
  address-family ipv4
  interface all enable
igmp snooping profile IRB_SNOOP
report-suppression disable
!
router pim
  address-family ipv4
  rp-address 10.10.10.10

```

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 HundredGigE0/0/0/8
  !
  routed interface BVI 100
  !
interface HundredGigE0/0/0/8
  l2transport
  !
interface BVI 100
  ipv4 address 10.21.1.1 255.255.255.0
  !
router vrrp
  interface BVI 100
  vrrp 1 ipv4 10.21.1.100
  vrrp 1 priority 100
  !

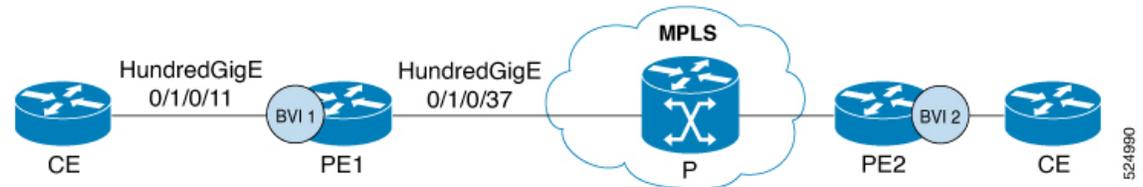
```

6PE or 6VPE with BVI configuration: example

The following example shows how to configure an MPLS 6PE/6VPE environment using BVIs at the CE-facing sides of the Cisco 8000 Series Router as the PE devices.

This figure shows the location of the BVI interfaces (green icons) on the Cisco 8000 Series Routers as the PE1 and PE2 devices.

Figure 11: BVI interfaces on the CE-facing sides in an MPLS 6PE or 6VPE network



This is a sample configuration only for the Cisco 8000 Series Router (PE1) device with a BVI interface numbered 1 on the CE-facing side, and a non-BVI interface (HundredGigE 0/1/0/37) on the core-facing side. A similar configuration would apply to the PE2 device:

```

! Be sure to configure IPv6 unicast address families
!
vrf 1
address-family ipv6 unicast
  import route-target
  100:2
  export route-target
  100:2

interface Loopback0
  ipv4 address 10.11.11.11/32
!
! Configure the BVI interface to participate in the VRF
! and with an IPv6 address.
!
interface BVI1
  vrf 1
  ipv6 address 2001:DB8:1/32
!
! Assign the Ethernet CE-facing interface to the
! L2VPn bridge domain where the routed BVI interface is also associated.
!
l2vpn
  bridge group 1
  bridge-domain 1
  interface HundredGigE 0/1/0/11
  routed interface BVI1
!
! Configure OSPF routing for the BVI interface for
! advertisement of its IPv6 address.
!
router ospfv3 1
  graceful-restart
  redistribute bgp 1
  area 1
  interface BVI1
  interface Loopback0
!
! Configure BGP routing and be sure to specify the

```

```
! IPv6 unicast address family.
! Note that the per-VRF label allocation mode is required
! and is the only supported label allocation mode.
!
router bgp 1
  bgp router-id 10.11.11.11
  bgp redistribute-internal
  bgp graceful-restart

  address-family ipv6 unicast
    redistribute ospfv3 1 match internal external
    label-allocation-mode per-vrf
    allocate-label all
  !
  address-family vpv6 unicast
  !
  neighbor 10.11.12.12
    remote-as 1
    update-source Loopback0
    address-family ipv6 unicast
      route-policy pass-all in
      route-policy pass-all out
    !
    address-family ipv6 labeled-unicast
    !
    address-family vpv6 unicast
      route-policy pass-all in
      route-policy pass-all out
    !
  vrf 1
    rd 100:2
    label-allocation-mode per-vrf
    address-family ipv6 unicast
      redistribute connected

mpls ldp
  router-id 10.11.11.11
  graceful-restart
  interface HundredGigE 0/1/0/37
```



CHAPTER 8

IP Event Dampening

Table 33: Feature History Table

Feature Name	Release	Description
IP event dampening	Release 25.1.1	Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*) *This feature is supported on Cisco 8011-4G24Y4H-I routers.

Feature Name	Release	Description
IP event dampening	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100, K100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>You can now enhance network stability by suppressing unnecessary flapping through IP Event Dampening. This feature prevents interfaces from repeatedly transitioning between up and down states due to intermittent issues, reducing operational disruptions. By configuring dampening parameters, network operators can control the penalty for each flap and set thresholds for interface recovery, thus ensuring smoother network performance and reliability. This proactive approach minimizes unnecessary alerts and interventions, allowing for focused attention on more critical network events, ultimately improving overall network uptime and efficiency.</p> <p>*Previously this feature was supported on Q200 and Q100. It is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 8712-MOD-M • 88-LC1-12TH24FH-E • 88-LC1-36EH • 88-LC1-52Y8H-EM

The IP Event Dampening feature introduces a configurable exponential decay mechanism to suppress the effects of excessive interface flapping events on routing protocols and routing tables in the network. This feature allows the network operator to configure a router to automatically identify and selectively dampen a local interface that is flapping.

Guidelines and Limitations

See the following guidelines and limitations before configuring IP Event Dampening feature:

- Due to changes in the netstack-IP component, all IP clients observe the impact of interface dampening.
 - When dampening is enabled, a penalty value is assigned to an interface. This value starts at 0 and increases by 1000 each time the interface state transitions from up to down.
 - For each flap of the interface, a certain penalty is added. The penalty decays exponentially when parameters are configured.
 - When the penalty exceeds a certain high level, the interface is dampened. It is unsuppressed when the penalty decays below a low level.
 - When an interface is dampened, the IP address and the static routes are removed from the interface. All the clients of IP get an IP delete notification.
 - When an interface is unsuppressed, the IP address and the relevant routes are added back. All the clients of IP get an IP address add notification for all the IP addresses of the interface.
 - All Layer 3 interfaces that are configured on the Ethernet interface, port changes, and SVI support this feature.
- [IP Event Dampening Overview, on page 187](#)
 - [Interface State Change Events, on page 187](#)
 - [Affected Components, on page 188](#)
 - [How to Configure IP Event Dampening, on page 189](#)

IP Event Dampening Overview

Interface state changes occur when interfaces are administratively brought up or down or if an interface changes state. When an interface changes state or flaps, routing protocols are notified of the status of the routes that are affected by the change in state. Every interface state change requires all affected devices in the network to recalculate best paths, install or remove routes from the routing tables, and then advertise valid routes to peer routers. An unstable interface that flaps excessively can cause other devices in the network to consume substantial amounts of system processing resources and cause routing protocols to lose synchronisation with the state of the flapping interface.

The IP Event Dampening feature introduces a configurable exponential decay mechanism to suppress the effects of excessive interface flapping events on routing protocols and routing tables in the network. This feature allows the network operator to configure a router to automatically identify and selectively dampen a local interface that is flapping. Dampening an interface removes the interface from the network until the interface stops flapping and becomes stable. Configuring the IP Event Dampening feature improves convergence times and stability throughout the network by isolating failures so that disturbances are not propagated. This, in turn, reduces the utilisation of system processing resources by other devices in the network and improves overall network stability.

Interface State Change Events

This section describes the interface state change events of the IP Event Dampening feature. This feature employs a configurable exponential decay mechanism that is used to suppress the effects of excessive interface flapping or state changes. When the IP Event Dampening feature is enabled, flapping interfaces are dampened from the perspective of the routing protocol by filtering excessive route updates. Flapping interfaces are

identified, assigned penalties, suppressed if necessary, and made available to the network when the interface stabilizes.

Suppress Threshold

The suppress threshold is the value of the accumulated penalty that triggers the router to dampen a flapping interface. The flapping interface is identified by the router and assigned a penalty for each up and down state change, but the interface is not automatically dampened. The router tracks the penalties that a flapping interface accumulates. When the accumulated penalty reaches the default or preconfigured suppress threshold, the interface is placed in a dampened state.

Half-Life Period

The half-life period determines how fast the accumulated penalty can decay exponentially. When an interface is placed in a dampened state, the router monitors the interface for additional up and down state changes. If the interface continues to accumulate penalties and the interface remains in the suppress threshold range, the interface will remain dampened. If the interface stabilises and stops flapping, the penalty is reduced by half after each half-life period expires. The accumulated penalty will be reduced until the penalty drops to the reuse threshold. The configurable range of the half-life period timer is from 1 to 45 minutes. The default half-life period timer is 1 minute.

Reuse Threshold

When the accumulated penalty decreases until the penalty drops to the reuse threshold, the route is unsuppressed and made available to other devices in the network. The range of the reuse value is from 1 to 20000 penalties. The default value is 750 penalties.

Maximum Suppress Time

The maximum suppress time represents the maximum time an interface can remain dampened when a penalty is assigned to an interface. The maximum suppress time can be configured from 1 to 255 seconds. The maximum penalty is truncated to maximum 20000 unit. The maximum value of the accumulated penalty is calculated based on the maximum suppress time, reuse threshold, and half-life period.

Affected Components

When an interface is not configured with dampening, or when an interface is configured with dampening but is not suppressed, the routing protocol behavior as a result of interface state transitions is not changed by the IP Event Dampening feature. However, if an interface is suppressed, the routing protocols and routing tables are immune to any further state transitions of the interface until it is unsuppressed.

Route Types

The following interfaces are affected by the configuration of this feature:

- Connected routes:
 - The connected routes of dampened interfaces are not installed into the routing table.

- When a dampened interface is unsuppressed, the connected routes will be installed into the routing table if the interface is up.
- Static routes:
 - Static routes assigned to a dampened interface are not installed into the routing table.
 - When a dampened interface is unsuppressed, the static route will be installed into the routing table if the interface is up.



Note Only the primary interface can be configured with this feature, and all subinterfaces are subject to the same dampening configuration as the primary interface. IP Event Dampening does not track the flapping of individual subinterfaces on an interface.

Supported Protocols

All the protocols that are used are impacted by the IP Event Dampening feature. The IP Event Dampening feature supports Border Gateway Protocol (BGP), Enhanced Interior Gateway Routing Protocol (EIGRP), Hot Standby Routing Protocol (HSRP), Open Shortest Path First (OSPF), Routing Information Protocol (RIP), and VRRP. Ping and SSH to the concerned interface IP address does not work.



Note The IP Event Dampening feature has no effect on any routing protocols if it is not enabled or an interface is not dampened.

How to Configure IP Event Dampening

Enabling IP Event Dampening

The `dampening` command is entered in interface configuration mode to enable the IP Event Dampening feature. If this command is applied to an interface that already has dampening configured, all dampening states are reset and the accumulated penalty will be set to 0. If the interface has been dampened, the accumulated penalty will fall into the reuse threshold range, and the dampened interface will be made available to the network. The flap counts, however, are retained.

Table 34: Procedure

Steps	Command or Action	Purpose
Step 1	configure terminal	Enters global configuration mode.
Step 2	interface <i>type number</i>	Enters interface configuration mode and configures the specified interface.

Steps	Command or Action	Purpose
Step 3	dampening [<i>half-life-period reuse-threshold</i>] [<i>suppress-threshold max-suppress [restart-penalty]</i>]	Enables interface dampening. <ul style="list-style-type: none"> Entering the <code>dampening</code> command without any arguments enables interface dampening with default configuration parameters. When manually configuring the timer for the <i>restart-penalty</i> argument, the values must be manually entered for all arguments.
Step 4	end	Exits interface configuration mode.

Verifying IP Event Dampening

Use the `show dampening interface` or `show interface dampening` commands to verify the configuration of the IP Event Dampening feature.

Table 35: Procedure

Steps	Command or Action	Purpose
Step 1	show dampening interface	Displays dampened interfaces.
Step 2	show interface dampening	Displays dampened interfaces on the local router.



CHAPTER 9

Configure Link Bundling

Table 36: Feature History Table

Feature Name	Release	Description
Configure link bundling	Release 25.4.1	Introduced in this release on: Fixed Systems (8700 [ASIC: K100])(select variants only*) *This feature is now supported on Cisco 8711-48Z-M routers.
Configure link bundling	Release 25.1.1	Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*) *This feature is supported on Cisco 8011-4G24Y4H-I routers.

Feature Name	Release	Description
Configure link bundling	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100, K100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>You can now enhance network resilience and efficiency by aggregating multiple physical interfaces into a single logical link. This bundling improves redundancy by ensuring continued data transmission even if individual links fail. It supports load balancing across bundled links, optimizing data flow and minimizing congestion. The feature is easily configurable and integrates seamlessly with existing network setups, providing a scalable solution for managing increased traffic demands. By utilizing link bundling, you achieve a more robust and flexible network infrastructure, promoting continuous connectivity and performance.</p> <p>*Previously this feature was supported on Q200 and Q100. It is now extended to:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 8712-MOD-M • 88-LC1-12TH24FH-E • 88-LC1-36EH+A8:B12 • 88-LC1-52Y8H-EM
1023 Ethernet Bundle Interfaces Support	Release 7.3.15	<p>With the introduction of this enhancement, the maximum system-wide bundle interface scale has increased from 512 to 1023 bundle interfaces. The default value remains at 64-member links for each bundle.</p>
64-bit Bandwidth Support	Release 7.3.15	<p>With this release, the Cisco 8000 Series Router supports 64-bit bandwidth field, as opposed to the previous 32-bit bandwidth field. 64-bit bandwidth enables the system to support interface bandwidths greater than 4.2T.</p>

This module describes the configuration of link bundle interfaces on the Cisco 8000 Series Router.

A link bundle is a group of one or more ports that are aggregated together and treated as a single link.

Each bundle has a single MAC, a single IP address, and a single configuration set (such as ACLs).



Note The router supports both Layer 2 and Layer 3 Link Bundles. If the Link Bundle is a Layer 3 interface, the system requires an IP address. If the Link Bundle is a Layer 2 interface, the system does not require an IP address. A Link Bundle on the router may contain Layer 2 and Layer 3 subinterfaces within it. In which case, the Layer 3 subinterfaces require IP addresses, but the Link Bundle interface does not require an IP address.

The router supports bundling for these types of interfaces:

- Ethernet interfaces

Feature History for Configuring Link Bundling

Release	Modification
Release 7.0.11	Support for this feature added on the router.
Release 7.2.1	Mixed speed bundle members feature added on the router.

- [Limitations and Compatible Characteristics of Ethernet Link Bundles, on page 193](#)
- [Prerequisites for Configuring Link Bundling on a Router, on page 195](#)
- [Information About Configuring Link Bundling, on page 195](#)
- [How to Configure Link Bundling, on page 206](#)
- [Configuration Examples for Link Bundling, on page 215](#)

Limitations and Compatible Characteristics of Ethernet Link Bundles

This list describes the properties and limitations of ethernet link bundles:

- Starting with Cisco IOS XR Release 7.2.1, the router supports mixed speed bundles, allowing member links with different bandwidths to be included in the same bundle. The traffic distribution across bundle members is based on the bandwidth of each link. Mixed speed bundles are subject to a maximum bandwidth ratio of 10:1 between the fastest and slowest member links.

For example, you can combine a 10 Gbps and a 100 Gbps link or a 100 Gbps and a 400 Gbps link in the same bundle; however, a 10 Gbps and a 400 Gbps link cannot be bundled together. Load balancing is performed in proportion to the bandwidth of each member link. Typical valid combinations include:

- 400G, 100G
- 400G, 40G
- 400G, 100G, 40G
- 100G, 40G
- 100G, 10G
- 100G, 40G, 10G

- 40G and 10G

Additionally, the total weight of the bundle must not exceed 64.

- 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 Link Aggregation Control Protocol (LACP).
- With Cisco IOS XR Release 7.3.15, a single router can support up to 1023 bundle interfaces, with each bundle accommodating up to 64 member links.

If adding a new line card causes these limits to be exceeded, the system will experience continuous Out of Resource (OOR) failures. To resolve these errors, you must either reduce the scale or disable the affected line card.

- 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.
- The MAC address that is set on the bundle becomes the MAC address of the links within that bundle.
- 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.
- QoS is supported and is applied proportionally on each bundle member.
- 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.
- In 8404-SYS-D router, the minimum LACP timer supported is 1 second. When the minimum timer is configured, and when you trigger RPFO, ensure that you use the appropriate command to prevent LACP drop.
 - With the LACP timer configured as 1 second, and if RPFO is triggered using the **reload** command, then LACP drops leading to bundle disconnection.
 - With the LACP timer configured as 1 second, and if RPFO is triggered using the **redundancy switchover** command, then LACP drop is prevented.

Prerequisites for Configuring Link Bundling on a Router

Before configuring Link Bundling, ensure that you meet the following tasks and conditions:

- You know the interface IP address (Layer 3 only).
- You know the links that you must include in the bundle that you are configuring.
- If you are configuring an Ethernet link bundle, you must install Ethernet line cards on the router.



Note For more information about physical interfaces, PLIMs, and modular services cards, refer to the *Cisco 8000 Series Router Hardware Installation Guide*.

Information About Configuring Link Bundling

To configure link bundling, you must understand the following concepts:

Link Bundling Overview

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. The system assigns a virtual interface to the bundled link. You can dynamically add and delete component links from the virtual interface.

The virtual interface is treated as a single interface on which you can configure an IP address and other software features that the link bundle uses. Packets sent to the link bundle are forwarded to one of the links in the bundle.

A link bundle is a group of ports that the system bundles together and the group then acts as a single link. Following are the advantages of link bundles:

- 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. You can add bandwidth without interrupting the packet flow.

Prior to Cisco IOS XR Software Release 7.3.15, the interface bandwidth was stored and processed as a 32-bit value, which supported bundles with an aggregate bandwidth of up to 4.2 Gbps (the sum of its members). Starting with Cisco IOS XR Software Release 7.3.15, the interface bandwidth is stored and processed as a 64-bit value. The 64-bit value supports significantly larger aggregate bandwidths, accommodating bundles with high-bandwidth members whose combined bandwidth can exceed 4.2 Tbps.

All the individual links within a single bundle must be of the same type.

Cisco IOS XR software supports the following methods 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. The system automatically removes the links from a bundle that are incompatible or have failed.

Link Aggregation Through LACP

The optional Link Aggregation Control Protocol (LACP) is defined in the IEEE 802 standard. LACP communicates between two directly connected systems (or peers) to verify the compatibility of bundle members. For the router, the peer can be either another router or a switch. LACP monitors the operational state of link bundles to ensure the following:

- All links terminate on the same two systems.
- Both systems consider the links to be part of the same bundle.
- All links have the appropriate settings on the peer.

LACP transmits frames containing the local port state and the local view of the partner system's state. The system analyzes these frames to ensure that both the systems are in agreement.

IEEE 802.3ad Standard

The IEEE 802.3ad standard typically defines a method of forming Ethernet link bundles.

For each link configured as a 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 by using 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. This determines the compatibility of the links that are configured to be members of a bundle.

Bundle MAC addresses in the router come from a set of reserved MAC addresses in the backplane. This MAC address stays with the bundle as long as the bundle interface exists. The bundle uses this MAC address until you configure a different MAC address. The member links use the bundle MAC address when passing the 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.

Configuring LACP Fallback

This section describes how to configure the LACP Fallback feature.

Procedure

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 192.168.1.27 255.0.0.0
```

Specifies a primary IPv4 address for an interface.

Step 4 **bundle lacp-fallback timeout 4 *number***

Example:

```
RP/0/RP0/CPU0:router(config-if)# bundle lacp-fallback timeout 4
```

Enables the LACP Fallback feature.

Step 5 **end or commit**

Example:

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

Saves configuration changes.

Step 6 **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 7 `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.

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.

Load Balancing

Load balancing is a forwarding mechanism that distributes traffic over multiple links that are based on certain parameters. The router support load balancing for all links in a bundle using Layer 2, Layer 3, and Layer 4

routing information. Starting with Cisco IOS XR Software Release 7.2.1, bandwidth based load-balancing is applicable to L3 unicast flows.

This section describes the load balancing support on link bundles.

For more information about other forms of load balancing on the router, see the following:

- Per-flow load balancing on non-bundle interfaces using Layer 3 and 4 routing information.
- Pseudowire (PW) Load Balancing beginning in Cisco IOS XR 4.0.1.

Layer 3 Egress Load Balancing on Link Bundles

Layer 3 load balancing support began on the router in Cisco IOS XR 7.0.11 release.

Layer 3 load balancing for link bundles is enabled globally by default.

The ingress linecard does bundle member selection and forwards the packet to the linecard and network processor (NP) corresponding to the selected bundle member. The same hash value is used for both ingress and egress linecards. Therefore, even though the egress linecard also does bundle member selection, it selects the same bundle member that was selected by the ingress linecard.

Multicast IPv4 and IPv6 Traffic

For outbound multicast IPv4 or IPv6 traffic, a set of egress linecards is predetermined by the system. If a bundle interface or bundle subinterface is an outgoing interface, the system selects the bundle member for each outgoing interface in a route based on the multicast group address. This helps with load distribution of multicast routed traffic to different bundle members, while providing traffic sequencing within a specific route.

The egress linecard does NP selection using the same approach, when bundle members are spread across multiple NPs within the egress linecard.

When the packet arrives on an egress NP, it uses the 5-tuple hash to select a bundle member within an NP for each packet. This provides better resiliency for bundle member state changes within an NP.

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.

Procedure

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
```

Configuring Custom LACP Short Period Time Intervals

This section describes how to configure custom short period time interval for sending and receiving LACP packets on a Gigabit Ethernet interface.

Procedure

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***time-interval*

Example:

```
RP/0/RP0/CPU0:router(config-if)# lacp period 300
```

Configures a custom period time interval for the sending and receiving of LACP packets. The interval can be in the range 100 to 1000 ms, in multiples of 100.

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 period time interval to the custom time of 300 milliseconds:

```
config
interface HundredGigE 0/1/0/1
  bundle id 1 mode active
  lacp period 300
commit
```

QoS and Link Bundling

On the router, when the system applies QoS on the bundle for either the ingress or egress direction, QoS is applied at each member interface. For complete information on configuring QoS on link bundles on the router, refer to the *Cisco 8000 Series Aggregation Services Router Modular Quality of Service Configuration Guide* and the *Cisco 8000 Series Aggregation Services Router Modular Quality of Service Command Reference*.

Link Bundle Configuration Overview

The following steps provide a general overview of the link bundle configuration process. Ensure that you clear all previous network layer configuration before adding it 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 that you created in Step 1 with the **bundle id** command in the interface configuration submode.

You can add up to 64 links to a single bundle.



Note The system configures a link as a member of a bundle from the interface configuration submode for that link.

Nonstop Forwarding During Card Failover

Cisco IOS XR software supports nonstop forwarding during a failover between active and standby paired RP cards. Nonstop forwarding ensures that there is no change in the state of the link bundles when a failover occurs.

For example, if an active RP fails, the standby RP becomes operational. The system replicates the configuration, node state, and checkpoint data of the failed RP to the standby RP. The bundled interfaces are present when the standby RP becomes the active RP.



Note Failover is always onto the standby RP.

You do not need to configure anything to guarantee that the system maintains the standby interface configurations.

Link Failover

When one member link in a bundle fails, the system redirects the traffic to the remaining operational member links and traffic flow remains uninterrupted.

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 switchover suppress-flaps** 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.

Designate a Member Link as Unviable

Table 37: Feature History Table

Feature Name	Release	Description
Designate a Member Link as Unviable	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8712-MOD-M • 8011-4G24Y4H-I
Designate a Member Link as Unviable	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-12TH24FH-E • 88-LC1-36EH+A8:B12 • 88-LC1-52Y8H-EM

Feature Name	Release	Description
Designate a Member Link as Unviable	Release 7.10.1	<p>Earlier, when a member link is added to an interface link bundle, the peer starts using the link as soon as the LACP communication is up. Sometimes, the hardware programming for the data-path does not get complete in this time resulting in packet loss without any notification to the source.</p> <p>You can now mark a member link as unviable to introduce a delay during which the link is treated as standby. By delaying the usage of the member link for data transmission, you can ensure that the link configuration is fully established, which enables successful data transmission.</p> <p>This feature introduces these changes:</p> <ul style="list-style-type: none"> • CLI: forwarding-unviable • YANG Data Model: New XPath for <code>Cisco-IOS-XR-bundlemgr-oper</code> (see Github, YANG Data Models Navigator).

Link bundling aggregates multiple physical links to a single logical link. When a member link is added to a link bundle, link aggregation control protocol (LACP) communication gets established with the peer to negotiate and control the link aggregation. LACP doesn't have any provision to incorporate a delay before letting data transmission over the link. Therefore, the peer starts using the link when the LACP communication is up. Occasionally, even though the link status is up, and LACP communication is up, the hardware programming for data-path packet forwarding doesn't get complete. In such scenarios, the transmitted data gets lost without any notification or error message to the source or destination of the traffic.

You can now delay the use of such member links, which aren't fully ready to handle data transmission, using the **forwarding-unviable** command. This command configures the link as forwarding-unviable and the member link is considered "standby" for bundle management. As standby member links of a bundle aren't used for data transmission, the usage of forwarding-unviable member links is delayed. When the member link is fully up, that is, the packet forwarding data-path is also completely programmed, you can disable forwarding-unviability of the link using **no forwarding-unviable** command. This removes the forwarding-unviable configuration of the link. Then, the link is treated as an "active" member of the bundle and is used in data transmission and load balancing.



Note It is recommended to wait for a few minutes before running **no forwarding-unviable** command to ensure that the packet forwarding data-path is completely programmed.

Guidelines and Restrictions for Designating Member Links as Unviable

- Forwarding-unviable is disabled on all Ethernet interfaces by default. Therefore, by default, all member links in a bundle are considered "active".
- A link bundle is considered up, only if at least one member link is active. Only the active member links in the link bundle are used for data transmission, load balancing, and redundancy.
- If a link bundle has only one member link, which is forwarding-unviable, the bundle state is considered "down".
- If all the member links in a bundle are forwarding-unviable, the bundle state is considered "down".
- Other existing threshold parameters such as minimum-active links, maximum-active links, and maximum-active-bandwidth, which are considered to determine the bundle state, continue to function along with forwarding-unviable functionality. For more details on these parameters, see [How to Configure Link Bundling, on page 206](#).
- There is no effect of forwarding-unviable configuration on individual Ethernet interfaces that are not part of a link bundle. That is, irrespective of the configuration, such non-member interfaces continue to attempt data transmission and reception.

How to Configure Link Bundling

This section contains the following procedures:

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.



Note When you use either physical Ethernet sub-interfaces or bundle Ethernet sub-interfaces, the device adds broadcast packets exiting a sub-interface to the output counters of the main interface. The output counters of the sub-interface do not increase. For example, when you send 100 broadcast packets through an egress physical or bundle sub-interface, the main interface output counters increase by 100, and the output broadcast packet counter of the sub-interface remains at zero.



Tip You can programmatically perform the configuration using `openconfig-lacp.yang`, `openconfig-if-aggregate.yang` OpenConfig data models, `Cisco-IOS-XR-bundlemgr-oper.yang` Cisco IOS XR native data model or `Cisco-IOS-XR-um-lacp-cfg.yang` Unified data model. To get started with using data models, see the *Programmability Configuration Guide for Cisco 8000 Series Routers*.

Procedure

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

- On the router, 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 **lacp fast-switchover**

Example:

```
RP/0/RP0/CPU0:router(config-if)# lacp fast-switchover
```

(Optional) If you enabled 1:1 link protection (you set the value of the **bundle maximum-active links** command to 1) on a bundle with member links running LACP, you can optionally disable the wait-while timer in the LACP state machine. Disabling this timer causes a bundle member link in standby mode to expedite its normal state negotiations, thereby enabling a faster switchover from a failed active link to the standby link.

Step 8 **exit**

Example:

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

Exits interface configuration submode for the Ethernet link bundle.

Step 9 **interface {GigabitEthernet | TenGigE} interface-path-id**

Example:

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

Enters interface configuration mode for the specified interface.

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.

Step 10 **bundle id bundle-id [mode {active | on | passive}]**

Example:

```
RP/0/RP0/CPU0:router(config-if)# bundle-id 3
```

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 11 **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 12 **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 13 **exit**

Example:

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

Exits interface configuration submenu for the Ethernet interface.

Step 14 **bundle id *bundle-id* [mode {active | passive | on}] no shutdown exit**

Example:

```
RP/0/RP0/CPU0:router(config)# interface GigabitEthernet 1/0/2/1
```

```
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 GigabitEthernet 1/0/2/3
```

```
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 15 **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 16 **exit****Example:**

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

Exits interface configuration mode.

Step 17 **exit****Example:**

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

Exits global configuration mode.

Step 18 Perform Step 1 through Step 15 on the remote end of the connection.

Brings up the other end of the link bundle.

Step 19 **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 20 **show lacp bundle Bundle-Ether *bundle-id*****Example:**

```
RP/0/RP0/CPU0:router# show lacp bundle
Bundle-Ether 3
```

(Optional) Shows detailed information about LACP ports and their peers.

Configuring VLAN Bundles

This section describes how to configure a VLAN bundle. The creation of a VLAN bundle involves three main tasks:

1. Create an Ethernet bundle.
2. Create VLAN subinterfaces and assign them to the Ethernet bundle.
3. Assign Ethernet links to the Ethernet bundle.

These tasks are described 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.

Procedure

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 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 5 **exit**

Example:

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

Exits the interface configuration submode.

Step 6 **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 4094 inclusive (0 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 7 **encapsulation dot1q**

Example:

```
RP/0/RP0/CPU0:router(config-subif)# encapsulation dot1q 100
```

Sets the Layer 2 encapsulation of an interface.

Step 8 **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 9 **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 10 **exit**

Example:

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

Exits subinterface configuration mode for the VLAN subinterface.

Step 11 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 12 **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 13 **exit**

Example:

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

Exits interface configuration mode.

Step 14 **exit**

Example:

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

Exits global configuration mode.

Step 15 **configure**

Example:

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

Enters global configuration mode.

Step 16 **interface {GigabitEthernet | TenGigE} interface-path-id**

Example:

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

Step 17 lACP fast-switchover

Example:

```
RP/0/RP0/CPU0:router(config-if)# lACP fast-switchover
```

(Optional) If you enabled 1:1 link protection (you set the value of the **bundle maximum-active links** command to 1) on a bundle with member links running LACP, you can optionally disable the wait-while timer in the LACP state machine. Disabling this timer causes a bundle member link in standby mode to expedite its normal state negotiations, thereby enabling a faster switchover from a failed active link to the standby link.

VLANs on an Ethernet Link Bundle

You can configure 802.1Q VLAN subinterfaces on 802.3ad Ethernet link bundles.



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, the system supports all VLAN subinterface configuration on that link bundle.

VLAN subinterfaces can support Ethernet Flow Points (EFPs) and Layer 3 services.

You can configure Layer 3 VLAN subinterfaces as follows:

```
interface bundle-ether instance.subinterface, encapsulation dot1q xxxxx
```

Configuring a Member Link as Unviable

Perform the following task to designate a member link as unviable.

Example Configuration

```
RP/0/RP0/CPU0:ios(config)#interface HundredGigE 0/0/0/34
RP/0/RP0/CPU0:ios(config-if)#forwarding-unviable
RP/0/RP0/CPU0:ios(config-if)#commit
RP/0/RP0/CPU0:ios(config-if)#end
```

Running Configuration

```
RP/0/RP0/CPU0:ios#show running-config interface HundredGigE 0/0/0/34
Thu Apr 20 11:11:55.744 UTC
interface HundredGigE0/0/0/34
  forwarding-unviable
!
```

Verification

Use **show bundle** command to view the forwarding-viable status of LAG members. Here, the interface HundredGigE 0/0/0/34 is added to ethernet bundle 3.

```
RP/0/RP0/CPU0:ios#show bundle
Thu Apr 20 11:29:42.500 UTC

Bundle-Ether3
  Status:                               Down
  Local links <active/standby/configured>: 0 / 0 / 1
  Local bandwidth <effective/available>:   0 (0) kbps
  MAC address (source):                   78c6.9991.3504 (Chassis pool)
  Inter-chassis link:                      No
  Minimum active links / bandwidth:        1 / 1 kbps
  Maximum active links:                    64
  Wait while timer:                        2000 ms
  Load balancing:
    Link order signaling:                  Not configured
    Hash type:                             Default
    Locality threshold:                    None
  LACP:                                     Operational
    Flap suppression timer:                Off
    Cisco extensions:                       Disabled
    Non-revertive:                          Disabled
  mLACP:                                    Not configured
  IPv4 BFD:                                 Not configured
  IPv6 BFD:                                 Not configured

Port          Device          State          Port ID          B/W, kbps
-----
Hu0/0/0/34    Local           Standby        0x8000, 0x0001  100000000
Link is not forwarding viable and in standby state
```

Configuration Examples for Link Bundling

This section contains the following examples:

Example: Configuring an Ethernet Link Bundle

The following example shows how to join two ports to form an EtherChannel bundle that runs LACP:

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

RP/0/RP0/CPU0:Router(config-if)# interface Bundle-Ether 3
RP/0/RP0/CPU0:Router(config-if)# ipv4 address 1.2.3.4/24
RP/0/RP0/CPU0:Router(config-if)# bundle minimum-active bandwidth 620000
RP/0/RP0/CPU0:Router(config-if)# bundle minimum-active links 1
RP/0/RP0/CPU0:Router(config-if)# exit
RP/0/RP0/CPU0:Router(config-if)# interface TenGigE 0/3/0/0
RP/0/RP0/CPU0:Router(config-if)# bundle id 3 mode active
RP/0/RP0/CPU0:Router(config-if)# no shutdown
RP/0/RP0/CPU0:Router(config)# exit
RP/0/RP0/CPU0:Router(config-if)# interface TenGigE 0/3/0/1
RP/0/RP0/CPU0:Router(config-if)# bundle id 3 mode active
RP/0/RP0/CPU0:Router(config-if)# no shutdown
```

```
RP/0/RP0/CPU0:Router(config-if)# exit
```

This example shows the configuration in the case of a mixed speed bundle:

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

RP/0/RP0/CPU0:Router(config-if)# interface bundle-ether 50
RP/0/RP0/CPU0:Router(config-if)# root
RP/0/RP0/CPU0:Router(config-if)# interface TenGigE 0/0/0/11
RP/0/RP0/CPU0:Router(config-if)# bundle id 50 mode active
RP/0/RP0/CPU0:Router(config-if)# no shutdown
RP/0/RP0/CPU0:Router(config-if)# interface TenGigE 0/0/0/16
RP/0/RP0/CPU0:Router(config-if)# bundle id 50 mode active
RP/0/RP0/CPU0:Router(config-if)# no shutdown
RP/0/RP0/CPU0:Router(config-if)# interface TenGigE 0/0/0/27
RP/0/RP0/CPU0:Router(config-if)# bundleid 50 mode active
RP/0/RP0/CPU0:Router(config-if)# no shutdown
RP/0/RP0/CPU0:Router(config-if)# interface HundredGigE 0/6/0/1
RP/0/RP0/CPU0:Router(config-if)# bundleid 50 mode active
RP/0/RP0/CPU0:Router(config-if)# no shutdown
RP/0/RP0/CPU0:Router(config-if)# root
RP/0/RP0/CPU0:Router(config)# commit
RP/0/RP0/CPU0:Router(config)# end
```

The following output is shown for the **show bundle bundle-ether** command:

show bundle bundle-ether50

```
Bundle-Ether50
Status: Up
Local links <active/standby/configured>: 4 / 0 / 4
Local bandwidth <effective/available>: 130000000 (130000000) kbps
MAC address (source): 0011.2233.4458 (Chassis pool)
Inter-chassis link: No
Minimum active links / bandwidth: 1 / 1 kbps
Maximum active links: 64
Wait while timer: 2000 ms
Load balancing: Default
LACP: Operational
Flap suppression timer: Off
Cisco extensions: Disabled
mLACP: Not configured
IPv4 BFD: Not configured
```

Port	Device	State	Port ID	B/W, kbps
Te0/0/0/11	Local	Active	0x8000, 0x0002	10000000
Link is Active				
Te0/0/0/16	Local	Active	0x8000, 0x0003	10000000
Link is Active				
Te0/0/0/27	Local	Active	0x8000, 0x0004	10000000
Link is Active				
Hu0/6/0/1	Local	Active	0x8000, 0x0001	100000000
Link is Active				

In order to view the weight of a mixed speed bundle, run the **show bundle load-balancing** command. The following is the truncated output of this command.

```
show bundle load-balancing bundle-ether50 location 0/0/cpu0
```

```

<snip>

Bundle-Ether50
Type:          Ether (L3)
Members <current/max>: 4/64
Total Weighting: 13
Load balance:  Default
Locality threshold: 65
Avoid rebalancing?  False
Sub-interfaces:  1

Member Information:
Port:          LON ULID BW
-----
Hu0/6/0/1      0  0 10
Te0/0/0/11     1  1  1
Te0/0/0/16     2  2  1
Te0/0/0/27     3  3  1

Platform Information:
=====

* Bundle Summary Information *
-----

Interface      : Bundle-Ether50   Ifhandle       : 0x00000ce0
Lag ID         : 1              Virtual Port    : 255
Number of Members : 4          Local to LC    : Yes
Hash Modulo Index : 13
MGSCP Operational Mode : No

Member Information:
LON   Interface  ifhandle  SFP  port  slot  remote/rack_id
-----
0     Hu0/6/0/1   0x100001c0 648 116  8    0/0
1     Te0/0/0/11 0x04000380 65  9  2    0/0
2     Te0/0/0/16 0x040004c0 67  8  2    0/0
3     Te0/0/0/27 0x04000780 72  4  2    0/0

</snip>

```

Example: Configuring a VLAN Link Bundle

The following example shows how to create and bring up two VLANs on an Ethernet bundle:

```

RP/0/RP0/CPU0:Router(config-subif)# config
RP/0/RP0/CPU0:Router(config-subif)# interface Bundle-Ether 1
RP/0/RP0/CPU0:Router(config-ifsubif)# ipv4 address 1.2.3.4/24
RP/0/RP0/CPU0:Router(config-ifsubif)# bundle minimum-active bandwidth 620000
RP/0/RP0/CPU0:Router(config-if)# bundle minimum-active links
RP/0/RP0/CPU0:Router(config-ifsubif)# exit
RP/0/RP0/CPU0:Router(config-subif)# ip addr 20.2.3.4/24
RP/0/RP0/CPU0:Router(config-subif)# interface Bundle-Ether 1.1
RP/0/RP0/CPU0:Router(config-subif)# encapsulation dot1q 100
RP/0/RP0/CPU0:Router(config-subif) # ip addr 10.2.3.4/24
RP/0/RP0/CPU0:Router(config-subif)# no shutdown
RP/0/RP0/CPU0:Router(config-subif)# exit
RP/0/RP0/CPU0:Router(config-if)# interface Bundle-Ether 1.2
RP/0/RP0/CPU0:Router(config-subif)# dot1q vlan 10
RP/0/RP0/CPU0:Router(config-subif)Router # ip addr20.2.3.4/24

```

■ Example: Configuring a VLAN Link Bundle

```
RP/0/RP0/CPU0:Router(config-subifif)# no shutdown
RP/0/RP0/CPU0:Router(config-subifif)# exit
RP/0/RP0/CPU0:Router(config)# interface gig 0/1/5/7
RP/0/RP0/CPU0:Router(config-if)# bundle-id 1 mode act
RP/0/RP0/CPU0:Router(config-if)# commit
RP/0/RP0/CPU0:Router(config-if)# exit
```



CHAPTER 10

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 (RP). The configuration and control plane are mirrored onto the standby RP and, in the event of a failover, the virtual interfaces move to the ex-standby, which then becomes the newly active RP.

Feature History for Configuring Loopback and Null Interfaces on Cisco IOS XR Software

Release	Modification
Release 7.0.11	This feature was introduced.

- [Prerequisites for Configuring Virtual Interfaces, on page 219](#)
- [Information About Configuring Virtual Interfaces, on page 219](#)
- [How to Configure Virtual Interfaces, on page 221](#)
- [Configuration Examples for Virtual Interfaces, on page 223](#)

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 that you need for each command. If you suspect a user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

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 or active. Any packet that the system transmits 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.

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 that is 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 nonbroadcast 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, the system enables the **ipv4 unreachable** command. If we do not want ICMP to send protocol unreachable, then you need to configure using the **ipv4 icmp unreachable disable** command.

By default, the system creates the Null 0 interface during boot process and you cannot remove it. You can configure the **ipv4 unreachable** command for this interface, but most configuration is unnecessary because this interface just discards all the packets that the system sends.

Use the **show interfaces null0** command to display the Null 0 interface.

Virtual Management Interface Overview

Configuring an IPv4 virtual address enables you to access the router from a single virtual address with a management network without prior knowledge of which RP is active. An IPv4 virtual address persists across route switch processor (RP) failover situations. For this to happen, the virtual IPv4 address must share a common IPv4 subnet with a management Ethernet interface on both the RPs.

On a router where each RP has multiple management Ethernet interfaces, the virtual IPv4 address maps to the management Ethernet interface on the active RP that shares the same IP subnet.

Active and Standby RPs and Virtual Interface Configuration

The standby RP is available and in a state in which it can take over the work from the active RPs should that prove necessary. Conditions that necessitate the standby RP to become the active RP and assume the active RP's duties include:

- Failure detection by a watchdog
- Administrative command to take over
- Removal of the active RP from the chassis

If a second RP is not present in the chassis while the first is in operation, a second RP may be inserted and automatically becomes the standby RP. The standby RP may also be removed from the chassis with no effect on the system other than loss of RP redundancy.

After failover, the virtual interfaces all are present on the standby (now active) RP. Their state and configuration are unchanged and there has been no loss of forwarding (in the case of tunnels) over the interfaces during the failover. The routers use nonstop forwarding (NSF) over bundles and tunnels through the failover of the host RP.



Note The user need not configure anything to guarantee that the standby interface configurations are maintained. Protocol configuration such as `tacacs source-interface`, `snmp-server trap-source`, `ntp source`, `logging source-interface` do not use the virtual management IP address as their source by default. Use the **ipv4 virtual address use-as-src-addr** command to ensure that the protocol uses the virtual IPv4 address as its source address. Alternatively, you can also configure a loopback address with the designated or desired IPv4 address and set that as the source for protocols such as TACACS+ using the **tacacs source-interface** command.

How to Configure Virtual Interfaces

This section contains the following procedures:

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.
- That IP address must not be used by another interface on the router.
- The IP address must not be used by an interface on any other router on the network.

```
RP/0/RP0/CPU0:router# configure
/* Enters interface configuration mode and names the new loopback interface */
RP/0/RP0/CPU0:router#(config)# interface Loopback 3
/* Assigns an IP address and subnet mask to the virtual loopback interface */
```

```
RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38/32

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

- 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.

```
/* Display the configuration of the loopback interface */
RP/0/RP0/CPU0:router# show interfaces Loopback 3
```

Configuring Null Interfaces

This task explains how to configure a basic null interface.

```
/* Enters global configuration mode. */

RP/0/RP0/CPU0:router# configure

/* Enter the null 0 interface configuration mode. */

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

/* Save configuration changes. */

RP/0/RP0/CPU0:router(config-null0)# end

/* Verif the configuration of the null interface. */

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

Configuring Virtual IPv4 Interfaces

This task explains how to configure an IPv4 virtual interface.

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

/* Define an IPv4 virtual address for the management Ethernet interface. */
```

```
RP/0/RSP0/CPU0:router(config)# ipv4 virtual address 10.3.32.154/8
RP/0/RSP0/CPU0:router(config-null0)# end
or
RP/0/RSP0/CPU0:router(config-null0)# commit
```

- 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.

This is an example for configuring a virtual IPv4 interface:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# ipv4 virtual address 10.3.32.154/8
RP/0/RSP0/CPU0:router(config-null0)# commit
```

Configuration Examples for Virtual Interfaces

This section provides the following configuration examples:

Configuring a Loopback Interface: Example

The following example indicates how to configure a loopback interface:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface Loopback 3
RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38/32
RP/0/RP0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes
RP/0/RP0/CPU0:router# show interfaces Loopback 3
```

```
Loopback3 is up, line protocol is up
Hardware is Loopback interface(s)
Internet address is 172.18.189.38/32
MTU 1514 bytes, BW Unknown
  reliability 0/255, txload Unknown, rxload Unknown
Encapsulation Loopback, loopback not set
Last clearing of "show interface" counters never
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
  0 packets input, 0 bytes, 0 total input drops
```

```

0 drops for unrecognized upper-level protocol
Received 0 broadcast packets, 0 multicast packets
0 packets output, 0 bytes, 0 total output drops
Output 0 broadcast packets, 0 multicast packets

```

Configuring a Null Interface: Example

The following example indicates 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 unreachable
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
Hardware is Null interface
Internet address is Unknown
MTU 1500 bytes, BW Unknown
  reliability 0/255, txload Unknown, rxload Unknown
Encapsulation Null, loopback not set
Last clearing of "show interface" counters never
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
  0 packets input, 0 bytes, 0 total input drops
  0 drops for unrecognized upper-level protocol
  Received 0 broadcast packets, 0 multicast packets
  0 packets output, 0 bytes, 0 total output drops
  Output 0 broadcast packets, 0 multicast packets

```

Configuring a Virtual IPv4 Interface: Example

```

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 11

Configure GRE Tunnels

Tunneling provides a mechanism to transport packets of one protocol within another protocol. This chapter describes GRE tunneling protocol.

Release	Feature(s) Added
Release 7.3.1	GRE Tunnel feature was introduced.

- [GRE tunnels](#), on page 225
- [Unidirectional GRE Encapsulation \(GREv4\)](#), on page 231
- [Unidirectional GRE Decapsulation \(GREv4\)](#), on page 231
- [ECMP and LAG Hashing for NVGRE Flows](#), on page 232

GRE tunnels

Table 38: Feature History Table

Feature Name	Release Information	Description
GRE tunnel	Release 25.4.1	Introduced in this release on: Fixed Systems (8700 [ASIC: K100])(select variants only*) *This feature is supported on Cisco 8711-48Z-M routers.
Disabling time-to-live (TTL) decrement at GRE encapsulation	Release 25.1.1	Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*) *This feature is supported on: <ul style="list-style-type: none">• 8712-MOD-M• 8011-4G24Y4H-I

Feature Name	Release Information	Description
Disabling time-to-live (TTL) decrement at GRE encapsulation	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-12TH24FH-E • 88-LC1-36EH • 88-LC1-52Y8H-EM
Disabling time-to-live (TTL) decrement at GRE encapsulation	Release 7.3.2	<p>This feature allows you to disable the time-to-live (TTL) decrement of the incoming packets. The result is that encapsulation of the original incoming packet takes place without any change in the TTL value.</p> <p>This feature avoids dropping incoming packets with a TTL value equal to one after GRE encapsulation.</p> <p>Before this release, the TTL value of incoming packets was decremented by one before GRE decapsulation.</p> <p>This feature introduces the tunnel ttl disable command.</p>
GRE tunnel	Release 24.4.1	<p>Introduced in this release on: Fixed Systems(8200, 8700)(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*).</p> <p>The Generic Routing Encapsulation (GRE) feature that transports packets of one protocol over another protocol in a simplified manner using encapsulation is now supported on the following hardware.</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 8712-MOD-M • 88-LC1-12TH24FH-E • 88-LC1-52Y8H-EM • 88-LC1-36EH

Feature Name	Release Information	Description
GRE tunnel	Release 7.3.1	<p>Generic Routing Encapsulation (GRE) provides a simple approach to transporting packets of one protocol over another protocol using encapsulation. This capability is now extended to the Cisco 8000 Series Routers.</p> <p>This feature supports:</p> <ul style="list-style-type: none"> • Unidirectional GRE encapsulation • Unidirectional GRE decapsulation <p>And introduces the following commands:</p> <ul style="list-style-type: none"> • show interface tunnel-ip <> accounting (encap) • show interface tunnel-ip <> accounting (decap)
Outer-header hashing support for MPLSoGRE and IPoGRE traffic	Release 7.3.1	<p>This feature allows load-balancing of GRE traffic in transit routers. A transit node distributes incoming GRE traffic evenly across all available ECMP links in a GRE tunnel topology. A hashing function uses GRE outer and inner header tuples such as source IP, destination IP, protocol, and router ID to determine traffic entropy. This capability is now extended to the Cisco 8000 Series Routers.</p>

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. GRE encapsulates a payload, that is, an inner packet that should be delivered to a destination network inside an outer IP packet. The GRE tunnel behaves as 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 toward the GRE tunnel endpoint. Upon reaching the tunnel endpoint, GRE encapsulation is removed and the payload is forwarded to the packet's ultimate destination.

A tunnel configured using encapsulation mode performs encapsulation of IPv4/IPv6 payload inside the GRE header. A tunnel configured using decapsulation mode performs the opposite. Here, outer GRE header is decapsulated and the inner IPv4/IPv6/MPLS payload is forwarded to the next hop router. Both encapsulation and decapsulation tunnel interfaces collect statistics periodically. The statistics can be displayed on demand using the CLI commands `show interface tunnel-ip1 accounting` and `show policy-map type pbr address-family ipv4 statistics`. For more information, see [Unidirectional GRE Encapsulation \(GREv4\), on page 231](#) and [Unidirectional GRE Decapsulation \(GREv4\), on page 231](#).

To perform load-balancing of GRE traffic in transit routers, a transit node distributes incoming GRE traffic evenly across all available ECMP links in a GRE tunnel topology. Furthermore, to determine traffic entropy, a hashing function uses GRE outer and inner header tuples such as source IP, destination IP, protocol, and router ID.

GRE encapsulation and decapsulation over BVI

Table 39: Feature History Table

Feature Name	Release Information	Description
GRE encapsulation and decapsulation over BVI	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100, K100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 8712-MOD-M • 88-LC1-12TH24FH-E • 88-LC1-36EH+A8:B12 • 88-LC1-52Y8H-EM
GRE encapsulation and decapsulation over BVI	Release 7.5.4	<p>You can now transport packets using the GRE protocol over Bridge-Group Virtual Interfaces (BVI).</p> <p>This feature uses GRE to encapsulate packets between two endpoints and transmit the encapsulated packets over a BVI interface. At the destination, the GRE packet is decapsulated.</p> <p>GRE encapsulation and decapsulation over BVI allows transmitting packets securely using network layer protocols while maintaining Layer 2 connectivity between the physical interfaces.</p>

From Cisco IOS XR Release 7.5.4, GRE packets are supported over a BVI interface. This support provides GRE encapsulation and decapsulation over the BVI interfaces.

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. A BVI is associated with a single bridge domain and represents the link between the bridging and the routing domains on the router.

When using GRE over BVI, the GRE header is added to the original IP packet before it is sent to the BVI. The BVI then bridges the encapsulated packet to the destination interface, which is a BVI, physical interface, or a remote network.

When the encapsulated packet reaches its destination, the receiving interface performs GRE decapsulation, which involves removing the GRE header from the original IP packet. The resulting IP packet is then forwarded to its final destination.

For information on BVI, see the *Integrated Routing and Bridging* section in the *L2VPN Configuration Guide for Cisco 8000 Series Routers*.

Supported Features on a GRE Tunnel

GRE tunnel supports the following features:

- GRE or IP-in-IP tunnels support 16 unique source addresses. These 16 unique source addresses are repeated multiple times to configure 1000 encapsulation tunnels or 64 decapsulation tunnels.
- GRE encapsulation supports the following features:
 - IPv4/IPv6 over GRE IPv4 transport
 - MPLS PoP over GRE IPv4 transport
 - ABF (Access List Based Forwarding) v4/v6 over GRE
 - VRF (Virtual Routing and Forwarding) support over GRE
- GRE decapsulation supports the following features:
 - PBR-based GRE decapsulation configuration
 - CLI-based GRE decapsulation configuration
 - IPv4/IPv6 over GRE decapsulation
 - MPLS/SRTE over GRE decapsulation
 - A GRE tunnel in decapsulation mode has only tunnel source configured, without any tunnel destination address. This decapsulated GRE tunnel behaves like a P2MP (Point-to-multipoint) tunnel, which means that an incoming GRE packet can have any source IP address and matching destination IP address to the tunnel source configured. However, once a source IP address is used for decapsulated P2MP tunnel, it cannot be re-used with other decapsulation tunnels.
- The command `tunnel ttl disable` is supported. This command controls TTL decrement of a packet being encapsulated. After configuring this command for a tunnel interface, TTL value of incoming packet is not decremented by one, and original incoming packet is encapsulated without changing the TTL. By default, `tunnel ttl disable` isn't configured. This means that the TTL of incoming packets is decremented by one before GRE encapsulation.

For example, consider an incoming packet that had the TTL value equal to one. On GRE encapsulation, the TTL value is decremented by one and becomes zero. Therefore the router will discard the packet and send an ICMP message back to the originating host. Using this feature, you can disable TTL decrement and avoid the packet discard.

Configuration Example

```
Router#configure
Router(config)#interface tunnel-ip30016
Router(config-if)#tunnel ttl disable
Router(config-if)#commit
```

Limitations for Configuring GRE Tunnels

This list describes the limitations for configuring GRE tunnels:

- GRE tunnels configured without any decapsulation or encapsulation mode support only ERPSAN feature.

- Don't create multiple GRE/IP-in-IP tunnels with the same pair of source and destination IP address or interface name. Configure all tunnels with unique source-destination pairs. In an encapsulation or decapsulation tunnel where only either source or destination is mentioned, the source-destination pair should also be unique when compared to other encapsulation or decapsulation tunnels.
- Bi-directional GRE tunnel isn't supported.
- Routing protocols over GRE tunnels aren't supported.
- Multicast over GRE isn't supported.
- GRE KA (Keep Alive) isn't supported.
- GRE parameters such as MTU (Maximum Transmission Unit) and key functionalities aren't supported.

Configure GRE Tunnels

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 source and destination. The router supports only uni-directional GRE with either encapsulation or decapsulation mode.

```
Router# configure
Router(config)# interface tunnel-ip1
Router(config-if)# ipv4 address 101.0.1.2 255.255.255.0
Router(config-if)# ipv6 address 101:0:1::2/64
Router(config-if)# tunnel mode gre ipv4 [encap | decap]
Router(config-if)# tunnel source 2.2.1.1
Router(config-if)# tunnel destination 2.2.2.1/32
Router(config-if)# commit
Router(config-if)# exit
```

To configure ABFv4/v6 over GRE:

```
router static
  address-family ipv4 unicast
    201.0.1.0/24 tunnel-ip1
  address-family ipv6 unicast
    201:0:1::0/64 tunnel-ip1

ipv4 access-list abf-gre
  1 permit ipv4 any any nexthop1 ipv4 201.0.1.2
ipv6 access-list abf6-gre
  1 permit ipv6 any any nexthop1 ipv6 201:0:1::2

interface HundredGigE0/0/0/24
  ipv4 address 24.0.1.1/24
  ipv6 address 24:0:1::1/64
  ipv4 access-group abf-gre ingress
  ipv6 access-group abf6-gre ingress
!
```

To configure MPLS PoP label over GRE:

```
router static
  address-family ipv4 unicast
    201.0.1.0/24 tunnel-ip1
  address-family ipv6 unicast
    201:0:1::0/64 tunnel-ip1

mpls static
  interface HundredGigE0/0/0/24
  lsp gre
```

```

in-label 30501 allocate
forward path 1 resolve-nextthop 201.0.1.2 out-label pop
!

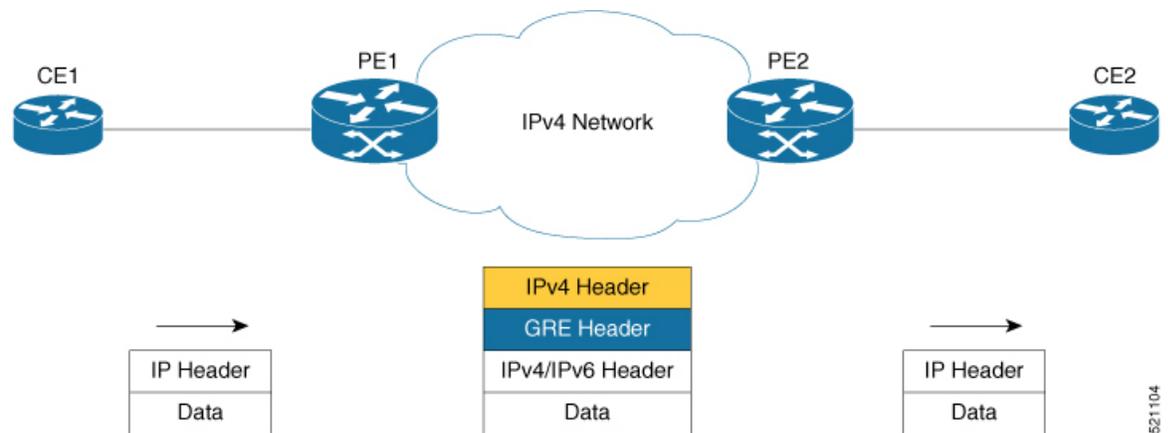
```



Note Bi-directional GRE tunnel supports only ERSPAN.

Unidirectional GRE Encapsulation (GREv4)

A tunnel configured using encapsulation mode performs encapsulation of IPv4/IPv6 payload inside the GRE header. The following figure shows GRE encapsulation. Routers in the IP cloud have no knowledge of encapsulated IP source address or destination address.



Configuration

The following example shows how to configure GRE tunnel encapsulation:

```

interface tunnel-ip1
  ipv4 address 101.0.1.1/24
  ipv6 address 101:0:1::1/64
  tunnel mode gre ipv4 encap
  tunnel source [ loopback1 | <any-ipaddress> | any-interface]
  tunnel destination [ 20.0.1.1/32 | 20.0.1.0/24 | 20.0.1.0/28]

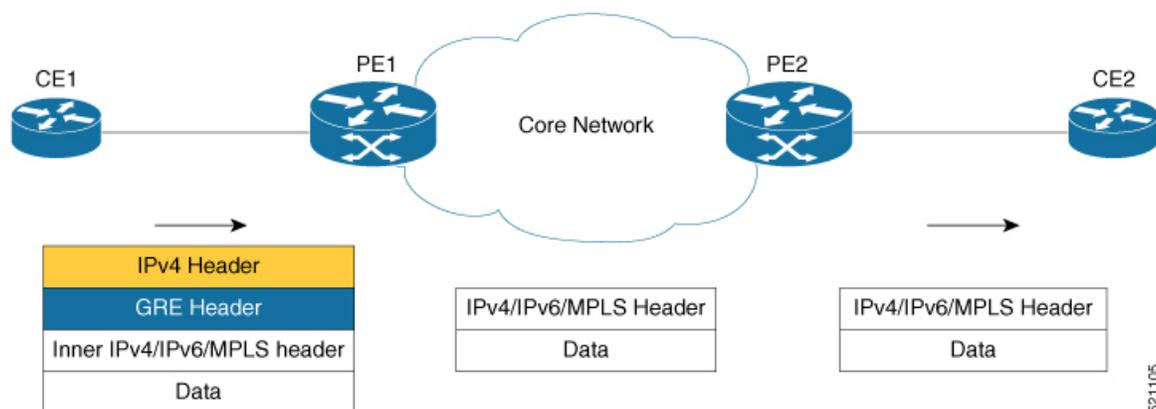
router static
  address-family ipv4 unicast
  201.0.1.0/24 tunnel-1

router static
  address-family ipv6 unicast
  201:0:1::0/64 tunnel-1

```

Unidirectional GRE Decapsulation (GREv4)

In unidirectional GRE decapsulation, the outer GRE header is decapsulated and the inner IPv4/IPv6/MPLS payload is forwarded to the next hop router. The following figure shows GRE decapsulation. In the figure, PE1 strips off outer GRE header and inner payload is forwarded as regular IPv4/IPv6/MPLS forwarding.



521105

Configuration

There are two methods to configure GRE tunnel decapsulation:

1. CLI-based tunnel decapsulation configuration

```
interface tunnel-ipl
  ipv4 address 101.0.1.1/24
  ipv6 address 101:0:1::1/64
  tunnel mode gre ipv4 decap
  tunnel source [ loopback1 | <any-ipaddress> | any-interface]
  tunnel destination [ 20.0.1.1/32 | 20.0.1.0/24 | 20.0.1.0/28]
```

2. PBR-based tunnel decapsulation configuration

```
class-map type traffic match-all test_gre1
  match protocol gre
  match destination-address ipv4 10.0.1.2 255.255.255.255
  match source-address ipv4 10.10.10.1 255.255.255.255
end-class-map

policy-map type pbr P1-test
  class type traffic test_gre1 decapsulate gre
vrf-policy vrf default address-family ipv4 policy type pbr input P1-test
```

ECMP and LAG Hashing for NVGRE Flows

Table 40: Feature History Table

Feature Name	Release Information	Feature Description
ECMP and LAG Hashing for NVGRE Flows	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8712-MOD-M • 8011-4G24Y4H-I

Feature Name	Release Information	Feature Description
ECMP and LAG Hashing for NVGRE Flows	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-12TH24FH-E • 88-LC1-36EH+A8:B12 • 88-LC1-52Y8H-EM
ECMP and LAG Hashing for NVGRE Flows	Release 7.5.2	<p>This feature allows transit routers to load balance the GRE traffic, based on GRE payload.</p> <p>A transit node distributes incoming GRE traffic across ECMP and LAG paths in a GRE tunnel topology. A hashing function uses GRE payload that consists of inner Ethernet frame with destination MAC and source MAC addresses, to derive the traffic entropy.</p> <p>ECMP and LAG hashing is enabled on Cisco 8000 series routers by default.</p>

Network Virtualization using Generic Routing Encapsulation (NVGRE) endpoints are network devices that act as interfaces between physical and virtual networks. NVGRE endpoint encapsulates Ethernet data frames to and from GRE tunnel. The encapsulated GRE packet is bridged and routed to the destination. On the destination, the NVGRE endpoint decapsulates the GRE packet to recover the original Ethernet frame. NVGRE is described in RFC 7637.

NVGRE uses the following header information for encapsulation:

Header	Parameters
Outer Ethernet Header	Destination MAC address, Source MAC
Outer IP Header	IPv4 and IPv6 addresses as delivery proto
GRE Header	GRE protocol type 0x6558 (transparent E
GRE Payload	Inner Ethernet frame with Destination M address

For load balancing the GRE traffic, the transit router uses GRE payload that consists of inner Ethernet frame with destination MAC and source MAC addresses. The transit router derives the traffic entropy information from the GRE payload.

The hashing function considers the following parameters of GRE packets, along with Router ID, for load balancing the GRE traffic:

Header	Parameters
Outer IPv4 Header	Source IP address, Destination IP address, IP
Outer IPv6 Header	Source IP address, Destination IP address, Flow ID (GRE)
Inner Header	Destination MAC address, Source MAC address

Restrictions for ECMP and LAG Hashing for NVGRE Flows

ECMP and LAG hashing does not support:

- Outer IPv4 header with Options field.
- Outer IPv6 header with extension headers.



CHAPTER 12

Configuring 802.1Q VLAN Interfaces

This module describes the configuration and management of 802.1Q VLAN interfaces.

The IEEE 802.1Q specification establishes a standard method for tagging Ethernet frames with VLAN membership information. It defines the operation of VLAN bridges that permit the definition, operation, and administration of VLAN topologies within a bridged LAN infrastructure.

The 802.1Q standard is intended to address the problem of how to divide large networks into smaller parts so broadcast and multicast traffic does not use more bandwidth than necessary. The standard also helps provide a higher level of security between segments of internal networks.

Feature History for Configuring 802.1Q VLAN Interfaces

Release	Modification
Release 7.0.11	This feature was introduced.
Release 7.2.12	Support for Layer 2 interfaces was introduced.

- [Prerequisites for Configuring 802.1Q VLAN Interfaces, on page 235](#)
- [Information About Configuring 802.1Q VLAN Interfaces, on page 236](#)
- [How to Configure 802.1Q VLAN Interfaces, on page 239](#)
- [Configuration Examples for VLAN Interfaces, on page 243](#)
- [Layer 2 interface VLAN encapsulation using VLAN ranges and lists, on page 245](#)

Prerequisites for Configuring 802.1Q VLAN 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.

Before configuring 802.1Q VLAN interfaces, ensure that you meet the following conditions:

- You must have configured a HundredGigE interface, a FourHundredGigE interface, or an Ethernet bundle interface.

Information About Configuring 802.1Q VLAN Interfaces

To configure 802.1Q VLAN interfaces, you must understand the following concepts:

802.1Q VLAN Overview

Table 41: Feature History Table

Feature Name	Release Information	Feature Description
802.1Q VLAN	Release 25.1.1	Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*) *This feature is supported on Cisco 8011-4G24Y4H-I routers.
802.1Q VLAN	Release 24.4.1	Introduced in this release on: Fixed Systems (8700) (select variants only*) * The 802.1Q VLAN functionality is now extended to the Cisco 8712-MOD-M routers.
802.1Q VLAN	Release 24.3.1	Introduced in this release on: Fixed Systems (8200 [ASIC: Q200, P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: Q100, Q200, P100])(select variants only*) * The 802.1Q VLAN functionality is now extended to: <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-52Y8H-EM • 88-LC1-36EH+A8:B12 • 88-LC1-12TH24FH-E
802.1Q VLAN	Release 24.2.11	Introduced in this release on: Modular Systems (8800 [LC ASIC: P100]) (select variants only*) A VLAN is a logical grouping of devices across different LAN segments that communicate as if they are on the same physical network, offering flexibility in user management, bandwidth allocation, and resource optimization. The IEEE 802.1Q protocol standard helps divide large networks into smaller segments to efficiently manage broadcast and multicast traffic, enhancing bandwidth usage and security. * This feature is now supported on routers with 88-LC1-36EH line cards.

Double-Tagged 802.1ad Encapsulation Options for Layer 3 Physical and Bundle Subinterfaces	Release 24.4.1	Introduced in this release on: Fixed Systems (8200 [ASIC: Q200, P100], 8700 [ASIC: P100, K100]); Centralized Systems (8600 [ASIC:Q200]); Modular Systems (8800 [LC ASIC: Q100, Q200, P100]) The support for Double-Tagged 802.1ad Encapsulation Options for Layer 3 Physical and Bundle Subinterfaces is now extended to all Systems in the Cisco 8000 Series Routers.
Double-Tagged 802.1ad Encapsulation Options for Layer 3 Physical and Bundle Subinterfaces	Release 7.3.2	This feature enables you to increase the number of VLAN tags in an interface and a subinterface. You can enable this feature either on a physical interface or a bundle interface. When you configure this feature with the dual tag, interfaces check for IP addresses along with MAC addresses. Verified Scalability Limits: <ul style="list-style-type: none"> • Total number of VLAN tags in an interface and a subinterface with single tag: 4094 • Total number of VLAN tags in an interface and a subinterface with double tag: 4094 * 4094

A VLAN is a group of devices on one or more LANs that you can configure so that the devices can communicate as if they were attached to the same wire. When in fact, they are located on several different LAN segments. Because VLANs are based on logical instead of physical connections, they are flexible for user and host management, bandwidth allocation, and resource optimization.

The IEEE 802.1Q protocol standard addresses the problem of dividing large networks into smaller parts so broadcast and multicast traffic does not consume more bandwidth than necessary. The standard also helps provide a higher level of security between segments of internal networks.

The 802.1Q specification establishes a standard method for inserting VLAN membership information into Ethernet frames.

Cisco IOS XR software supports VLAN subinterface configuration on 40Gigabit, HundredGig, FourHundredGig, and bundle interfaces.

802.1Q Tagged Frames

The IEEE 802.1Q tag-based VLAN uses an extra tag in the MAC header to identify the VLAN membership of a frame across bridges. This tag is used for VLAN and quality of service (QoS) priority identification. The VLANs can be created statically by manual entry or dynamically through Generic Attribute Registration Protocol (GARP) VLAN Registration Protocol (GVRP). The VLAN ID associates a frame with a specific VLAN and provides the information that switches must process the frame across the network. A tagged frame is four bytes longer than an untagged frame and contains two bytes of Tag Protocol Identifier (TPID) residing within the type and length field of the Ethernet frame and two bytes of Tag Control Information (TCI) which starts after the source address field of the Ethernet frame.

Subinterfaces

Subinterfaces are logical interfaces that you can create on a hardware interface. These software-defined interfaces allow the segregation of traffic into separate logical channels on a single hardware interface. It also allows for the better utilization of the available bandwidth on the physical interface.

You can distinguish subinterfaces from each other by adding an extension at the end of the interface name and designation. For instance, the system indicates Ethernet subinterface 23 on the physical interface designated TenGigE 0/1/0/0, by TenGigE 0/1/0/0.23.

Before the system allows a subinterface to pass traffic, it must have a valid tagging protocol encapsulation and VLAN identifier assigned. All Ethernet subinterfaces always default to the 802.1Q VLAN encapsulation. However, you must explicitly define the VLAN identifier.

Supported Encapsulation

Table 42: 802.1ad Encapsulation Support for Layer 3 Interfaces and subinterfaces

Interface Type	Encapsulation	Standard	Support Status
Layer 3 interface Layer 3 subinterface Layer 3 bundle subinterface	Single-Tag Encapsulation	dot1ad	Supported (From Cisco IOS XR Software Release 24.4.1 onwards)
		dot1q	Supported.
	Double-Tag Encapsulation	dot1ad <> dot1q<>	Supported.
		dot1q <> dot1q<>	¹ Supported.

¹ The **encapsulation dot1q <x> second-dot1q <y>** encapsulation type is supported on Q200-based line cards from Cisco IOS XR Software Release 24.1.1 onwards and supported for all hardware platforms in the Cisco 8000 Series Routers from Cisco IOS XR Software Release 24.4.1 onwards.

For information about supported encapsulation for Layer 2 Interfaces and subinterfaces, see [Virtual LANs in Layer 2 VPNs](#).

Subinterface MTU

The system inherits the subinterface maximum transmission unit (MTU) from the physical interface with an additional four bytes allowed for the 802.1Q VLAN tag.

Native VLAN

The router does not support a native VLAN. However, the equivalent functionality is accomplished using an **encapsulation** command as follows:

```
encapsulation dot1q TAG-ID
```

Layer 2 VPN on VLANs

The Layer 2 Virtual Private Network (L2VPN) feature enables Service Providers (SPs) to provide Layer 2 services to geographically disparate customer sites.

The configuration model for configuring VLAN attachment circuits (ACs) is similar to the model used for configuring basic VLANs, where the user first creates a VLAN subinterface, and then configures that VLAN in subinterface configuration mode. To create an AC, you need to include the **I2transport** keyword in the **interface** command string to specify that the interface is a Layer 2 interface.

VLAN ACs support three modes of L2VPN operation:

- Basic Dot1Q AC—The AC covers all frames that are received and sent with a specific VLAN tag.
- QinQ AC— Only outer tag (s-tag) of 0x88a8 and inner tag (c-tag) of 0x8100 is supported.

Keep the following in mind when configuring L2VPN on a VLAN:

- Cisco IOS XR software supports 255 ACs per LC.

Use the **show interfaces** command to display AC information.

How to Configure 802.1Q VLAN Interfaces

This section contains the following procedures:

Configuring 802.1Q VLAN Subinterfaces

This task explains how to configure 802.1Q VLAN subinterfaces. To remove these subinterfaces, see the “Removing an 802.1Q VLAN Subinterface” section.



Tip You can programmatically configure and retrieve the VLAN interfaces and subinterfaces parameters using `openconfig-vlan.yang` OpenConfig data model. To get started with using data models, see the *Programmability Configuration Guide for Cisco 8000 Series Routers*.

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

```
/* Enter subinterface configuration mode and specifies the interface type, location, and subinterface number. */
```

```
RP/0/RP0/CPU0:router(config)# interface TenGigE 0/2/0/4.10
```

- Replace the *interface-path-id* argument with one of the following instances:
 - Physical Ethernet interface instance, or with an Ethernet bundle instance. Naming notation is *rack/slot/module/port*, and a slash between values is required as part of the notation.
 - Ethernet bundle instance. Range is from 1 through 65535.
 - Replace the *subinterface* argument with the subinterface value. Range is from 0 through 4095.
 - Naming notation is *interface-path-id.subinterface*, and a period between arguments is required as part of the notation.

```
/* Set the Layer 2 encapsulation of an interface. */
```

```
RP/0/RP0/CPU0:router(config-subif)# encapsulation dot1q 100
```



- Note**
- The **dot1q vlan** command is replaced by the **encapsulation dot1q** command on the Cisco 8000 Series Router. It is still available for backward-compatibility, but only for Layer 3 interfaces.

/* Assign an IP address and subnet mask to the subinterface. */

```
RP/0/RP0/CPU0:router(config-subif)# ipv4 address 178.18.169.23/24
```

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

/* The **exit** command is not explicitly required. */

```
RP/0/RP0/CPU0:router(config-subif)# exit
RP/0/RP0/CPU0:router(config)# end
or
RP/0/RP0/CPU0:router(config)# commit
```

- 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

```
"RP/0/RP0/CPU0:S3(config)#interface fourHundredGigE 0/5/0/1.100
RP/0/RP0/CPU0:S3(config-subif)#ipv4 address 100.100.100.100/31
RP/0/RP0/CPU0:S3(config-subif)#encapsulation dot1q 100
RP/0/RP0/CPU0:S3(config-subif)#no shutdown
RP/0/RP0/CPU0:S3(config-subif)#commit
Mon Jul  8 23:05:01.979 PDT
RP/0/RP0/CPU0:S3(config-subif)#end
RP/0/RP0/CPU0:S3#show interfaces fourHundredGigE 0/5/0/1.100 brief
Mon Jul  8 23:05:08.784 PDT
```

Intf	Intf	LineP	Encap	MTU	BW
------	------	-------	-------	-----	----

```

-----
Name          State      State          Type (byte)  (Kbps)
-----
FH0/5/0/1.100 up        up             802.1Q 1518 400000000

RP/0/RP0/CPU0:S3#show interfaces brief location 0/5/CPU0 | include 802.1Q
Mon Jul  8 23:07:43.929 PDT
      FH0/5/0/1.100      up        up             802.1Q 1518 400000000
RP/0/RP0/CPU0:S3#

RP/0/RP0/CPU0:S3#
    
```

Configuring an Attachment Circuit on a VLAN

Use the following procedure to configure an attachment circuit on a VLAN.

SUMMARY STEPS

1. **configure**
2. **interface** [**HundredGigE** | **TenGigE** | **Bundle-Ether** | **TenGigE**] *interface-path* *id.subinterface* **l2transport**
3. **encapsulation dot1q** *vlan-id*
4. **end** or **commit**
5. **show interfaces** [**HundredGigE** | **TenGigE**] *interface-path-id.subinterface*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure terminal	Enters global configuration mode.
Step 2	interface [HundredGigE TenGigE Bundle-Ether TenGigE] <i>interface-path</i> <i>id.subinterface</i> l2transport Example: RP/0/RP0/CPU0:router(config)# interface TenGigE 0/1/0/0.1 l2transport	Enters subinterface configuration and specifies the interface type, location, and subinterface number. <ul style="list-style-type: none"> • Replace the argument with one of the following instances: • Physical Ethernet interface instance, or with an Ethernet bundle instance. Naming notation is <i>rack/slot/module/port</i>, and a slash between values is required as part of the notation. • Ethernet bundle instance. Range is from 1 through 65535. • Replace the <i>subinterface</i> argument with the subinterface value. Range is from 0 through 4095. • Naming notation is <i>instance.subinterface</i>, and a period between arguments is required as part of the notation.

	Command or Action	Purpose
		<ul style="list-style-type: none"> You must include the I2transport keyword in the command string; otherwise, the configuration creates a Layer 3 subinterface rather than an AC.
Step 3	encapsulation dot1q <i>vlan-id</i> Example: <pre>RP/0/RP0/CPU0:router(config-subif)# encapsulation dot1q 100</pre>	Sets the Layer 2 encapsulation of an interface.
Step 4	end or commit Example: <pre>RP/0/RP0/CPU0:router(config-if-12)# end</pre> or <pre>RP/0/RP0/CPU0:router(config-if-12)# commit</pre>	Saves configuration changes. <ul style="list-style-type: none"> When you issue the end command, the system prompts you to commit changes: <pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre> <ul style="list-style-type: none"> Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.
Step 5	show interfaces [HundredGigE TenGigE] <i>interface-path-id.subinterface</i> Example: <pre>RP/0/RP0/CPU0:router# show interfaces TenGigE 0/3/0/0.1</pre>	(Optional) Displays statistics for interfaces on the router.

Removing an 802.1Q VLAN Subinterface

This task explains how to remove 802.1Q VLAN subinterfaces that have been previously configured using the Configuring 802.1Q VLAN subinterfaces section in this module.

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

```
/* Remove the subinterface, which also automatically deletes all the configuration applied to the subinterface.
*/
```

```
RP/0/RP0/CPU0:router(config)# no interface TenGigE 0/2/0/4.10
```

- Replace the *instance* argument with one of the following instances:
 - Physical Ethernet interface instance, or with an Ethernet bundle instance. Naming notation is *rack/slot/module/port*, and a slash between values is required as part of the notation.
 - Ethernet bundle instance. Range is from 1 through 65535.
 - Replace the *subinterface* argument with the subinterface value. Range is from 0 through 4095.

Naming notation is *instance.subinterface*, and a period between arguments is required as part of the notation.



Note Repeat to remove other VLAN subinterfaces.

```
RP/0/RP0/CPU0:router(config)# end
or
RP/0/RP0/CPU0:router(config)# commit
```

- 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.

Configuration Examples for VLAN Interfaces

This section contains the following example:

VLAN Subinterfaces: Example

The following example shows how to create three VLAN subinterfaces at one time:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface TenGigE 0/2/0/4.1

RP/0/RP0/CPU0:router(config-subif)# encapsulation dot1q 100
RP/0/RP0/CPU0:router(config-subif)# ipv4 address 10.0.10.1/24
RP/0/RP0/CPU0:router(config-subif)# interface TenGigE0/2/0/4.2
```

```

RP/0/RP0/CPU0:router(config-subif)# encapsulation dot1q 101
RP/0/RP0/CPU0:router(config-subif)# ipv4 address 10.0.20.1/24
RP/0/RP0/CPU0:router(config-subif)# interface TenGigE0/2/0/4.3

RP/0/RP0/CPU0:router(config-subif)# encapsulation dot1q 102
RP/0/RP0/CPU0:router(config-subif)# ipv4 address 10.0.30.1/24
RP/0/RP0/CPU0:router(config-subif)# commit
RP/0/RP0/CPU0:router(config-subif)# exit
RP/0/RP0/CPU0:router(config)# exit

RP/0/RP0/CPU0:router# show ethernet trunk bundle-Ether 1
Trunk                               Sub types          Sub states
VLAN trunks: 1,
  1 are 802.1Q (Ether)
Sub-interfaces: 3,
  3 are up.
802.1Q VLANs: 3,
  3 have VLAN Ids,

RP/0/RP0/CPU0:router# show vlan interface
Interface      St Ly   MTU   Subs   L3
Up    Down  Ad-Down
Te0/2/0/4.1      802.1Q      10 up
Te0/2/0/4.2      802.1Q      20 up
Te0/2/0/4.3      802.1Q      30 up
RP/0/RP0/CPU0:router# show vlan trunks brief
BE1           Up L3   1514   1000
  0    1000    1000    0    0

Summary
           1000    0    1000    1000    0    0
Te0/2/0/4      802.1Q (Ether)    up

```

The following example shows how to create two VLAN subinterfaces on an Ethernet bundle:

```

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface bundle-ether 2
RP/0/RP0/CPU0:router(config-if)# ipv4 address 192.168.2.1/24
RP/0/RP0/CPU0:router(config-if)# exit
RP/0/RP0/CPU0:router(config)# interface bundle-ether 2.1

RP/0/RP0/CPU0:router(config-subif)# encapsulation dot1q 100
RP/0/RP0/CPU0:router(config-subif)# ipv4 address 192.168.100.1/24
RP/0/RP0/CPU0:router(config-subif)# exit
RP/0/RP0/CPU0:router(config)# interface bundle-ether 2.2

RP/0/RP0/CPU0:router(config-subif)# encapsulation dot1q 200
RP/0/RP0/CPU0:router(config-subif)# ipv4 address 192.168.200.1/24
RP/0/RP0/CPU0:router(config-subif)# exit
RP/0/RP0/CPU0:router(config)# commit

```

Layer 2 interface VLAN encapsulation using VLAN ranges and lists

Table 43: Feature History Table

Feature Name	Release Information	Feature Description
Layer 2 interface VLAN encapsulation using VLAN ranges and lists	Release 25.4.1	Introduced in this release on: Fixed Systems (8700 [ASIC: K100])(select variants only*) *This feature is supported on Cisco 8711-48Z-M routers.
Layer 2 interface VLAN encapsulation using VLAN ranges and lists	Release 25.1.1	Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*) *This feature is supported on Cisco 8011-4G24Y4H-I routers.

Layer 2 interface VLAN encapsulation using VLAN ranges and lists	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200, 8700 [ASIC: K100])(select variants only*); Centralized Systems (8600); Modular Systems (8800 [LC ASIC: Q200, P100])(select variants only*)</p> <p>You can now leverage the VLAN ranges and lists to effectively separate networks operating over shared links and devices. VLAN encapsulation is typically determined by the access network and customer edge (CE) device, limiting the network provider's control over the VLAN tag or Ethernet type of customer traffic.</p> <p>The VLAN ranges and lists support various customer traffic types, enhancing network flexibility and management.</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8201-32FH • 8201-24H8FH • 8202-32FH-M • 8212-48FH-M • 8608 • 8711-32FH-M • 8712-MOD-M • 88-LC0-34H14FH • 88-LC0-36FH • 88-LC0-36FH-M • 88-LC1-36EH • 88-LC1-52Y8H-EM • 88-LC1-12TH24FH-E <p>This feature modifies these changes:</p> <p>CLI:</p> <ul style="list-style-type: none"> • encapsulation dot1ad • encapsulation dot1ad dot1q • encapsulation dot1q • encapsulation dot1q second-dot1q
--	----------------	--

VLAN encapsulation is a networking technique that:

- embeds VLAN tags within Ethernet frames to distinguish and identify network traffic

- relies on unique VLAN IDs, organized through ranges and lists, to manage and distinguish different VLANs, and
- ensures logical separation of traffic types to enhance network performance, security, and manageability.

VLAN ranges and lists

VLAN ranges and lists differ in their approach to VLAN encapsulation:

- A VLAN range is a sequence of contiguous VLAN IDs used for encapsulation. For instance, VLAN IDs from 200 to 250 can be grouped in a range.
- A VLAN list is a specific enumeration of VLAN IDs used for encapsulation. For instance, VLAN IDs 100, 123, and 150 can be grouped in a list.

VLAN list and range can be used together in configuration.

Benefits of Layer 2 interface VLAN encapsulation using VLAN ranges and lists

The VLAN encapsulation using VLAN ranges and lists offers several key benefits and is crucial for efficient network management:

- **Simplified configuration:** Configuring multiple VLANs with a single command, rather than individually, reduces the time and effort required for network setup and modification.
- **Enhanced manageability:** Managing VLANs becomes easier when grouped into lists or ranges. Changes can be applied to multiple VLANs simultaneously, simplifying network management tasks.
- **Improved scalability:** Grouping VLANs supports more efficient scaling in large-scale network environments, enabling the network to handle more VLANs without significantly increasing configuration complexity. A single subinterface can be used instead of multiple subinterfaces.
- **Reduced configuration errors:** Using fewer commands to configure multiple VLANs decreases the likelihood of configuration errors, resulting in a more stable and reliable network setup.

Restrictions of Layer 2 interface VLAN encapsulation using VLAN ranges and lists

These are the restrictions of Layer 2 interface VLAN encapsulation using VLAN ranges and lists:

- Supported only on Q200, P100, and K100 line cards and systems.
- Double-tagged encapsulations support the inner or outer tag as a list or range of VLAN IDs, but not both. For example:
 - `encapsulation dot1ad 100-200 dot1q 300` (supported)
 - `encapsulation dot1ad 100 dot1q 200,300` (supported)
 - `encapsulation dot1ad 10-20 dot1q 30-40` (not supported)
- Supported only on Layer 2 subinterfaces.
- The maximum number of list entries, individual or range per encapsulation, is 64.
- Ranges of up to 4094 VLAN IDs are supported.
- For lists with more than ten list items, use **encapsulation list-extended** command.

For example, **encapsulation list-extended dot1q <x1>, ... , <x10>**

Supported encapsulations

- encapsulation dot1q <x> , <y> (For example, encapsulation dot1q 200, 210, 300, 310, 400)
- encapsulation dot1q <x> - <y> (For example, encapsulation dot1q 200 – 400, 600-801, 901-1000)
- encapsulation dot1q <x> second-dot1q <y>, <z>
- encapsulation dot1q <x> second-dot1q <y> - <z>
- encapsulation dot1q <x> - <y> second-dot1q <z>
- encapsulation dot1ad <x> dot1q <y>, <z>
- encapsulation dot1ad <x> dot1q <y> - <z>
- encapsulation dot1ad <x> - <y> dot1q <z>
- encapsulation dot1ad <x>, <y> dot1q <z>
- encapsulation dot1ad <x>, <y>
- encapsulation dot1ad <x> - <y>

Configure a VLAN range and a VLAN list on Layer 2 subinterfaces

Define the matching criteria to map VLAN-tagged frames to the appropriate service instance.

Procedure

Step 1 Configure a VLAN range and a VLAN list on Layer 2 subinterfaces using the appropriate encapsulation method based on your requirements.

- If you want to configure a service with a list of noncontiguous customer VLAN (C-VLAN) ranges:

```
Router# configure
Router(config)# interface HundredGigE0/3/0/0.1 l2transport
Router(config-subif)# encapsulation dot1ad 200-400,600-801
Router(config-subif)# commit
```

- If you want to configure a service with a list of C-VLANs:

```
Router# configure
Router(config)# interface HundredGigE0/2/0/0.1 l2transport
Router(config-subif)# encapsulation dot1q 200,210,300,310,400
Router(config-subif)# commit
```

- If you want to configure a service with a range of service provider VLANs (S-VLANs) and a single inner C-VLAN:

```
Router# configure
Router(config)# interface HundredGigE0/4/0/0.1 l2transport
Router(config-subif)# encapsulation dot1ad 100-200 dot1q 300
Router(config-subif)# commit
```

- If you want to configure a QinQ service with a single S-VLAN and a list of C-VLANs:

```
Router# configure
Router(config)# interface HundredGigE0/5/0/0.1 l2transport
Router(config-subif)# encapsulation dot1q 100 second-dot1q 200,300
Router(config-subif)# commit
```

Step 2 Execute the **show interfaces** command to verify the VLAN range and VLAN list on the Layer 2 subinterface.

The corresponding value of the outer match appears depending on the VLAN configuration. The *Outer Match* field corresponds to the Dot1ad VLAN range and list in this example.

```
Router# show interfaces HundredGigE0/3/0/0.1
HundredGigE0/3/0/0.1 is UP, line protocol is UP
Interface state transitions: 0
Hardware is VLAN sub-interface(s), address is dc05.39c7.9440
Layer 2 Transport Mode
MTU 1518 bytes, BW 100000000 Kbit (Max: 100000000 Kbit)
reliability Unknown, txload Unknown, rxload Unknown
Encapsulation 802.1ad Virtual LAN,
Outer Match: Dot1ad VLAN 200-400,600-801
Ethertype Any, MAC Match src any, dest any
loopback not set,
Last input never, output never
Last clearing of "show interface" counters never
0 packets input, 0 bytes
0 input drops, 0 queue drops, 0 input errors
0 packets output, 0 bytes
0 output drops, 0 queue drops, 0 output errors
```

In this example, the *Outer Match* and *Inner Match* fields correspond to the Dot1ad VLAN range 100-200 and VLAN list Dot1Q VLAN 300.

```
Router# show interfaces HundredGigE0/4/0/0.1
HundredGigE0/4/0/0.1 is UP, line protocol is UP
Interface state transitions: 0
Hardware is VLAN sub-interface(s), address is dc05.39c7.9440
Layer 2 Transport Mode
MTU 1522 bytes, BW 100000000 Kbit (Max: 100000000 Kbit)
reliability Unknown, txload Unknown, rxload Unknown
Encapsulation 802.1ad-802.1Q Virtual LAN,
Outer Match: Dot1ad VLAN 100-200
Inner Match: Dot1Q VLAN 300
Ethertype Any, MAC Match src any, dest any
loopback not set,
Last input never, output never
Last clearing of "show interface" counters never
0 packets input, 0 bytes
0 input drops, 0 queue drops, 0 input errors
0 packets output, 0 bytes
0 output drops, 0 queue drops, 0 output errors
```




CHAPTER 13

Configure IP-in-IP tunnels

This chapter provides conceptual and configuration information for IP-in-IP tunnels.

Table 44: Feature History

Release 7.0.11	This feature was introduced.
Release 7.0.14	Support for the following feature was introduced in Configure Tunnels: <ul style="list-style-type: none">• Extended ACL must match on the outer header for IP-in-IP Decapsulation.

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 8000 Series Routers support IP-in-IP decapsulation with all possible combinations of IPv4 and IPv6 that include

- 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.

Table 45: Feature History Table

Feature Name	Release Information	Feature Description
Increased IP-in-IP tunnel scale for gRIBI-based nexthops	Release 25.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: Q200]); Centralized Systems (8600 [ASIC:Q200]); Modular Systems (8800 [LC ASIC: Q200])</p> <p>This feature enables the router to efficiently load balance and manage high traffic volumes by increasing the scale for encapsulation and decapsulation nexthops programmed using gRPC Routing Information Base Interface (gRIBI) to 12K IP-in-IP tunnels.</p> <p>This feature introduces these changes:</p> <p>CLI: The hw-module profile cef iptunnel scale command is modified.</p>
Increase in IP-in-IP decapsulation tunnels support	Release 25.3.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: Q200]); Centralized Systems (8600 [ASIC:Q200]); Modular Systems (8800 [LC ASIC: Q200])</p> <p>With this release, we have revised the maximum number of IPv4 and IPv6 IP-in-IP decapsulation tunnels from 64 to 200 on Cisco Silicon One Q200 ASIC-based systems. An increased number enhances the router's ability to support larger and more complex IP-in-IP tunneling scenarios improving scalability, efficiency, and flexibility in network design.</p>
IPv4 packets with IPv6 outer header	Release 25.4.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100])(select variants only*)</p> <p>*This feature is now supported on Cisco 8711-48Z-M routers</p>
IPv4 packets with IPv6 outer header	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on Cisco 8011-4G24Y4H-I routers.</p>

Feature Name	Release Information	Feature Description
IPv4 packets with IPv6 outer header	Release 24.4.1	<p>Introduced in this release on: Fixed Systems(8200, 8700)(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*).</p> <p>This feature that allows decapsulation of IPv4 and IPv6 tunnels with IPv6 headers helps the administrators to benefit from an improved IPv6 routing and security without upgrading their entire network to IPv6.</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 8712-MOD-M • 88-LC1-12TH24FH-E • 88-LC1-52Y8H-EM • 88-LC1-36EH
IPv4 packets with IPv6 outer header	Release 7.5.3	<p>With this release, decapsulation of IPv4 and IPv6 tunnels with IPv6 outer headers are supported.</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>

IP-in-IP tunneling can be used to connect remote networks securely or provide virtual private network (VPN) services.

This simplified network topology provides the transport VRF as the default VRF for an IPv4 or IPv6 network.

Figure 12: IP-in-IP tunnel network topology



A maximum of 3500 IP-in-IP tunnels are supported until Cisco IOS XR Release 25.4.1.

Increased IP-in-IP tunnel scale for gRIBI-based nexthops

From Cisco IOS XR Release 25.4.1, you can use the **hw-module profile cef iptunnel scale** command to configure up to 12,000 IP-in-IP tunnels that are shared between the gRIBI-based encapsulation and decapsulation next-hop tunnels. Prior to Cisco IOS XR Release 25.4.1, the router supported up to 3500 IP-in-IP tunnels.

This feature ensures efficient load balancing and high-volume traffic management.

- [Restrictions for IP-in-IP tunnel configuration, on page 254](#)
- [Configuration example for IPv4 tunnel, on page 254](#)
- [Configuration example for IPv6 tunnel, on page 255](#)
- [Configure improved scale for IP-in-IP tunnels, on page 257](#)
- [Controlling the TTL Value of Inner Payload Header, on page 258](#)
- [Time-to-Live uniform mode, on page 259](#)
- [IP-in-IP Decapsulation, on page 263](#)
- [Hashing for load balancing, on page 273](#)
- [ECMP Hashing Support for Load Balancing, on page 280](#)

Restrictions for IP-in-IP tunnel configuration

- The feature does not support decapsulation tunnels on subinterfaces.
- Only the default Virtual Routing and Forwarding (VRF) instance is supported.
- IPv6 link local addresses are not supported.
- Regular tunnels cannot use a configured IP address as the tunnel source; only a non-existent IP address can be used.
- Configuring multiple interfaces with the same IP address is not supported.
- Each line card can have different number of Network Processor (NP) slices.
- The maximum IPv4 and IPv6 IP-in-IP decapsulation tunnels supported is 64 per slice.
- From Release 25.3.1 onwards, the maximum IPv4 and IPv6 IP-in-IP decapsulation tunnels supported is 200 per slice.

Configuration example for IPv4 tunnel

This example provides the configuration for IPv4 tunnel.

Table 46: PE1 and PE2 Router configuration

PE1 Router configuration	PE2 Router configuration
<pre> interface GigabitEthernet0/0/0/0 !! Link between PE1-PE2 ipv4 address 100.1.1.1/24 ! 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 100.1.1.2/24 ! 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>

Table 47: CE1 and CE2 Router configuration

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

This example provides the configuration for IPv6 tunnel.

Table 48: PE1 and PE2 Router configuration

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>

Table 49: CE1 and CE2 Router configuration

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>

Configure improved scale for IP-in-IP tunnels

Procedure

Step 1 Enter the **hw-module profile cef iptunnel scale** command to configure the maximum IP-in-IP tunnel scale to 12K for gRIBI-based encapsulation and decapsulation nexthops.

Example:

```
Router# configure
Router(config)# hw-module profile cef iptunnel scale
Router(config)# commit
Router(config)# exit
```

Step 2 Reload the line cards with the **reload location all** command to enable increased IP-in-IP scale limit.

```
Router# reload location all
```

Step 3 Execute the **show hw-module profile cef** command to view the configuration status of CEF hardware-modules.

```
Router# show hw-module profile cef

Fri Aug 22 13:23:17.796 UTC
-----
Knob                               Status      Applied    Action
-----
CBF Enable                          Unconfigured  N/A       None
CBF forward-class-list              Unconfigured  N/A       None
BGPLU                               Unconfigured  N/A       None
LPTS ACL                            Unconfigured  N/A       None
Dark Bandwidth                     Unconfigured  N/A       None
SR-OPT                              Unconfigured  N/A       None
IP Redirect Punt                    Unconfigured  N/A       None
IPv6 Hop-limit Punt                 Unconfigured  N/A       None
MPLS Per Path Stats                 Unconfigured  N/A       None
SRv6 Per Path Stats                 Unconfigured  N/A       None
Tunnel TTL Decrement                Unconfigured  N/A       None
High-Scale No-LDP-Over-TE           Unconfigured  N/A       None
Label over TE counters               Unconfigured  N/A       None
```

Highscale LDPoTE No SRoTE	Unconfigured	N/A	None
LPTS Pifib Entry Counters	Unconfigured	N/A	None
Unipath surpf	Unconfigured	N/A	None
Source-based rtbh	Unconfigured	N/A	None
Vxlan ipv6 tunnel scale	Unconfigured	N/A	None
Encap Exact	Unconfigured	N/A	None
Iptunnel scale	Configured	Yes	None
BGPLU over RSVPTE Enable	Unconfigured	N/A	None
MPLS Decap Stats	Unconfigured	N/A	None

Controlling the TTL Value of Inner Payload Header

Cisco 8000 Routers allow you to control the TTL value of inner payload header of IP-in-IP tunnel packets before it gets forwarded to the next-hop router. This feature enables a router to forward custom formed IP-in-IP stacked packets even if the inner packet TTL is 1. Therefore, this feature enables you to measure the link-state and path reachability from end to end in a network.



Note After you enable or disable the decrement of the TTL value of the inner payload header of a packet, you do not need to reload the line card.

Configuration

To disable the decrement of the TTL value of inner payload header of an IP-in-IP packet, use the following steps:

1. Enter the global configuration mode.
2. Disable the decrement of TTL value of inner payload header of an IP-in-IP packet.

Configuration Example

```
/* Enter the Global Configuration mode. */
Router# configure

/* Disable the decrement of TTL value of inner payload header of an IP-in-IP packet. */
Router(config)# hw-module profile cef ttl tunnel-ip decrement disable
Router(config)# commit
```



Note Starting from Release 7.3.3, Cisco IOS XR 8000 router supports a maximum of 16 IP-in-IP decap tunnels with unique source addresses. If 15 unique tunnel sources are configured that is rounded to 95% of the tunnel hardware resource OOR threshold level. As a result, the OOR State displays *Red* in **show controllers npu resources sipidxtbl location all** command output.

Associated Commands

- [hw-module profile cef ttl tunnel-ip decrement disable](#)

Time-to-Live uniform mode

Time-to-Live (TTL) uniform mode is a mechanism that:

- ensures consistent TTL management by synchronizing the TTL values between inner and outer packet headers during encapsulation and decapsulation, allowing the receiving device to accurately interpret the packet's remaining lifespan
- allows you to copy the TTL values from inner headers to outer headers during encapsulation (ENCAP) and from outer headers to inner headers during decapsulation (DECAP), and
- ensures consistent TTL management across various network scenarios.

For more information on the various network scenarios, see [Use cases for TTL uniform mode on a router, on page 260](#).

Table 50: Feature History Table

Feature Name	Release Information	Feature Description
Copy TTL value to IP headers	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: Q200]; Centralized Systems (8600 [ASIC:Q200]); Modular Systems (8800 [LC ASIC: Q200])</p> <p>We've introduced support for Time-to-Live (TTL) uniform mode, which ensures consistent TTL management by synchronizing the TTL values between inner and outer packet headers during encapsulation and decapsulation, allowing the receiving device to accurately interpret the packet's remaining lifespan. TTL uniform mode is enabled only for the pbr vrf-redirect mode in IP-in-IP tunnels.</p>

Benefits of TTL uniform mode

Enabling TTL uniform mode offers these advantages:

- **Enhanced packet integrity and lifespan accuracy:** The TTL uniform mode ensures consistent TTL management by allowing the copying of TTL values between inner and outer headers during encapsulation and decapsulation. This consistency helps in accurately interpreting the packet's remaining lifespan at the receiving device.
- **Network diagnostics and troubleshooting:** By controlling and monitoring the TTL values, you can better diagnose and troubleshoot network paths and performance issues.

Configuration guidelines for TTL uniform mode

These configuration guidelines apply to the TTL uniform mode:

- **Hardware and feature prerequisites:** TTL uniform mode is only enabled on the Cisco Silicon One Q200 ASIC-based systems when the **pbr vrf-redirect** mode in the **hw-module profile** command is enabled.
- **Post-configuration requirements:** You must reload the router by using the **reload location all** command for the configuration changes to take effect.

Use cases for TTL uniform mode on a router

This table details the various scenarios of encapsulation and decapsulation, and their corresponding TTL action that the router performs.

Table 51: Use cases for TTL uniform mode on a router

Use case if the TTL uniform mode is..	Then..	Example
encapsulation-only	The router decrements the TTL value and copies the value from the inner header to outer header.	<p>Consider this example for the encapsulation-only use case for a packet:</p> <ul style="list-style-type: none"> • Initial state: TTL value of the packet is 100 • Decrement: The router decrements the inner header TTL value by one, making it 99. • Copy: The router then copies the inner header TTL value to the outer header TTL value. The outer header TTL value becomes 99. • Result: The inner and outer header TTL becomes 99, making it uniform.

Use case if the TTL uniform mode is..	Then..	Example
decapsulation-only	The router decrements the TTL value and copies the value from the outer header to inner header.	<p>Consider this example for the decapsulation-only use case for a packet:</p> <ul style="list-style-type: none">• Initial state: The outer header TTL value is 77 and the inner header TTL value is 99• Decrement: The router decreases the outer header TTL value by 1, making it 76.• Copy: The router then copies the outer header TTL value to the inner header TTL value.• Removal: The router removes the outer header TTL value.• Result: The final packet TTL value becomes 76.

Use case if the TTL uniform mode is..	Then..	Example
decapsulation and encapsulation	The router copies the TTL value from the outer header and applies it to the new outer header. Decrements the TTL value by 1 and forwards the packet.	<p>Consider this example for the encapsulation and decapsulation use case. Such scenarios apply when packet travels through IP tunnels:</p> <ul style="list-style-type: none"> • Encapsulation <ul style="list-style-type: none"> • Initial state: TTL value of the packet is 100 • Decrement: The router decrements the inner header TTL value by one, making it 99. • Copy: The router then copies the inner header TTL value to the outer header TTL value. The outer header TTL value becomes 99. • Result: The inner and outer header TTL becomes 99, making it uniform. • Decapsulation <ul style="list-style-type: none"> • Initial state: The outer header TTL value is 77 and the inner header TTL value is 99 • Decrement: The router decreases the outer header TTL value by 1, making it 76. • Copy: The router then copies the outer header TTL value to the inner header TTL value. • Removal: The router removes the outer header TTL value. • Result: The final packet TTL value becomes 76.

Use case if the TTL uniform mode is..	Then..	Example
decapsulation and lookup	The router copies the TTL value from the outer header to the inner header. Decrements the TTL value by 1 and forwards the packet.	Consider this example for the decapsulation and lookup use case for a packet: <ul style="list-style-type: none"> • Initial state: The outer header TTL value is 77 and the inner header TTL value is 99. • Decrement: The router decreases the outer header TTL value by 1, making it 76. • Copy: The router copies the outer header TTL value to the inner header TTL value. • Removal: The router removes the outer header TTL value. • Lookup: The router then forwards the packet to the next hop.
repair	The router keeps the inner header TTL value unchanged and forwards the packet as usual. The repair use case occurs after the encapsulation-only and decapsulation-encapsulation use cases.	Consider this example for the repair use case for a packet: <ul style="list-style-type: none"> • Initial state: The primary path for a packet on a tunnel is unavailable. • Initiate recycle: The router begins the recycling of the packet by re-entering the packet into the ingress pipeline from the egress pipeline. • Header update: The router updates the packet by replacing the old outer header with a new outer header.

IP-in-IP Decapsulation

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.

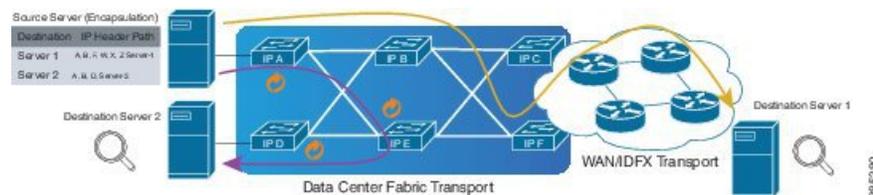
In IP-in-IP encapsulation and decapsulation has two types of packets. The original IP packets that are encapsulated are called Inner packets and the IP header stack added while encapsulation are called the Outer packets.



Note The router only supports decapsulation and no encapsulation. Encapsulation is done by remote routers.

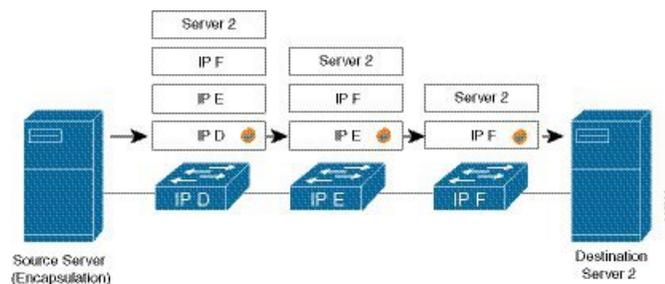
The following topology describes a use case where IP-in-IP encapsulation and decapsulation are used for different segments of the network from source to destination. The IP-in-IP tunnel consists of multiple routers that are used to decapsulate and direct the packet through the data center fabric network.

Figure 13: IP-in-IP Decapsulation Through a Data Center Network



The following illustration shows how the stacked IPv4 headers are decapsulated as they traverse through the decapsulating routers.

Figure 14: 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.

Figure 15: Encapsulated Packet

[-] Frame	
[-] EthernetII	
Preamble (hex)	fb55555555555d5
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 in the routers to decapsulate the packet as it traverses the IP-in-IP tunnel:

```
Router(config)# interface loopback 0
Router(config-if)# ipv4 address 127.0.0.1/32
Router(config-if)# no shutdown
Router(config-if)# interface tunnel-ip 10
```

```
Router(config-if)# ipv4 unnumbered loopback 1
Router(config-if)# tunnel mode ipv4 decap
Router(config-if)# tunnel source loopback 0
```

- **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 with respect to the router interface.



Note You can configure the tunnel destination only if you want to decapsulate packets from a particular destination. If no tunnel destination is configured, then all the ip-in-ip ingress packets on the configured interface are decapsulated.

Running Configuration

```
Router# show running-config interface tunnel-ip 10
...
interface tunnel-ip 10
ipv4 unnumbered loopback 1
tunnel mode ipv4 decap
```

Extended ACL to Match the Outer Header for IP-in-IP Decapsulation

Starting with Cisco IOS XR Software Release 7.0.14, extended ACL has to match on the outer header for IP-in-IP Decapsulation. Extended ACL support reduces mirrored traffic throughput. This match is based only on the IPv4 protocol, and extended ACL is applied to the received outermost IP header, even if the outer header is locally terminated.

Sample configuration:

```
Router#show running-config interface bundle-Ether 50.5
Tue May 26 12:11:49.017 UTC
interface Bundle-Ether50.5
ipv4 address 101.1.5.1 255.255.255.0
encapsulation dot1q 5
ipv4 access-group ExtACL_IPinIP ingress
ipv4 access-group any_dscpegg egress
!

Router#show access-lists ipv4 ExtACL_IPinIP hardware ingress location$
Tue May 26 12:11:55.940 UTC
ipv4 access-list ExtACL_IPinIP
10 permit ipv4 192.168.0.0 0.0.255.255 any ttl gt 150
11 deny ipv4 172.16.0.0 0.0.255.255 any fragments
12 permit ipv4 any any
```

Decapsulation using tunnel source direct

Table 52: Feature History Table

Feature Name	Release Information	Feature Description
Decapsulation using tunnel source direct	Release 25.4.1	Introduced in this release on: Fixed Systems (8700 [ASIC: K100])(select variants only*) *This feature is supported on Cisco 8711-48Z-M routers.
Decapsulation using tunnel source direct	Release 25.1.1	Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*) *This feature is supported on Cisco 8011-4G24Y4H-I routers.
Decapsulation using tunnel source direct	Release 24.4.1	Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100, K100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*) *This feature is now supported on: <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 8712-MOD-M • 88-LC1-12TH24FH-E • 88-LC1-36EH+A8:B12 • 88-LC1-52Y8H-EM
Decapsulation using tunnel source direct	Release 7.5.3	Tunnel source direct allows you to decapsulate the tunnels on any L3 interface on the router. You can use the tunnel source direct 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 8000 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.
- Source-direct and regular decap tunnels can't co-exist for a specific address-family. Any configuration that attempts to enable both is automatically rejected, and an error message is displayed to indicate the conflict.
- Inline modification of an existing regular decap tunnel (**tunnel source interface | IP address**) to a source-direct tunnel (**tunnel source direct**), or changing a source-direct tunnel to a regular decap tunnel, is not supported. Commit-replace may fail if the same tunnel-id is used as part of the commit-replace operation. You must delete the tunnel and recreate it.

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 Tunnel1
  tunnel mode ipv6ipv6 decapsulate-any
  tunnel source direct
  no shutdown

Router#show interface tunnel 1
Tunnel1 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
```

Configure Tunnel Destination with an Object Group

Table 53: Feature History Table

Feature Name	Release Information	Description
Configure Tunnel Destination with an Object Group	Release 7.5.4	<p>You can now assign an object group as the destination for an IP-in-IP decapsulation tunnel. With this functionality, you could configure an IPv4 or IPv6 object group consisting of multiple IPv4 or IPv6 addresses as the destination for the tunnel instead of a single IPv4 or IPv6 address. Using an object group instead of a singular IP address. This helps reduce the configuration complexity in the router by replacing the multiple tunnels with one destination with a single decapsulation tunnel that supports a diverse range of destinations</p> <p>The feature introduces these changes:</p> <ul style="list-style-type: none"> • CLI: New tunnel destination command. • YANG Data Model: New object-group option supported in Cisco-IOS-XR-um-if-tunnel-cfg.yang Cisco native model (see GitHub).

In IP-in-IP Decapsulation, the router accepts a packet on a tunneled interface only when the tunnel IP address matches the source IP address of the incoming packets. With this implementation, the user needs to configure separate interface tunnels for each IP address that the router needs to receive the traffic packets. This limitation often leads to configuration overload on the router.

You can eliminate the configuration overload on the router by assigning an object group as the tunnel destination for IPv4 and IPv6 traffic types. That is, the router matches the source IP address of the incoming packet against the object group available as the tunnel destination. The decapsulation tunnel accepts the incoming traffic packets when there's a match between the packet source and the object group. Otherwise, the router drops the packets.

Restrictions

The following restrictions are applicable to the tunnel destination with an object group feature:

- GRE tunnels don't support configuring object groups as the tunnel destination.

- The router supports configuring tunnel destination with an object group only when the tunnel source is tunnel source direct.
- You can configure the object group as tunnel destination only on default VRF.
- Configuring object groups as the tunnel destination isn't applicable to tunnel encapsulation.
- Subinterfaces don't support configuring object groups as the tunnel destination.
- Configuring object groups as the tunnel destination feature is mutually exclusive with ACL and QoS features.
- The tunnel destination feature supports only IPv4 and IPv6 object groups.
- The router does not support changing tunnel configuration after its creation. Configure the tunnel source direct and tunnel destination with an object group while creating the tunnel only.

Prerequisites

- Define an object group including the network elements for the tunnel destination.
- Enable the tunnel source direct feature. For more information, see [Decapsulation using tunnel source direct, on page 267](#).

Configuration example

This section provides an example for configuring the tunnel destination with an object group.

IPv4 configuration

```
Router# configure
/* Configure the IPv4 object group */
Router(config)# object-group network ipv4 Test_IPv4
Router(config-object-group-ipv4)# 192.0.2.0/24
Router(config-object-group-ipv4)# 198.51.100.0/24
Router(config-object-group-ipv4)# 203.0.113.0/24
Router(config-object-group-ipv4)# commit
Router(config-object-group-ipv4)# exit

/* Enters the tunnel configuration mode */
Router(config)# interface tunnel-ip 1

/* Configures the tunnel mode */
Router(config-if)# tunnel mode ipv4 decap

/* Configures the tunnel to accept all packets with destination address matching the IP
addresses on the router */
Router(config-if)# tunnel source direct

/* Configures the destination of the tunnel as the defined object-group */
Router(config-if)# tunnel destination object-group ipv4 Test_IPv4

Router(config-if)# no shutdown
Router(config-if)# commit
Router(config-if)# exit
```

IPv6 configuration

```
Router# configure
/* Configure the IPv6 object group */
Router(config)# object-group network ipv6 Test_IPv6
```

```

Router(config-object-group-ipv6)# 2001:DB8::/32
Router(config-object-group-ipv6)# 2001:DB8::/48
Router(config-object-group-ipv6)# commit
Router(config-object-group-ipv6)# exit

/* Enters the tunnel configuration mode */
Router(config)# interface tunnel-ip 2

/* Configures the tunnel mode */
Router(config-if)# tunnel mode ipv6 decap

/* Configures the tunnel to accept all packets with destination address matching the IP
addresses on the router */
Router(config-if)# tunnel source direct

/* Configures the destination of the tunnel as the defined object-group */
Router(config-if)# tunnel destination object-group ipv6 Test_IPv6

Router(config-if)# no shutdown
Router(config-if)# commit
Router(config-if)# exit

```

Running Configuration

```

Router# show running-config object-group
object-group network ipv4 Test_IPv4
 192.0.2.0/24
 198.51.100.0/24
 203.0.113.0/24
!
object-group network ipv6 Test_IPv6
 2001:DB8::/32
 2001:DB8::/48
!

Router#show running-config interface tunnel-ip 1
interface tunnel-ip1
 tunnel mode ipv4 decap
 tunnel source direct
 tunnel destination object-group ipv4 Test_IPv4
!

Router#show running-config interface tunnel-ip 2
Fri Nov 29 11:26:54.716 UTC
interface tunnel-ip2
 tunnel mode ipv6 decap
 tunnel source direct
 tunnel destination object-group ipv6 Test_IPv6
!

```

Verification

```

Router# show tunnel ip ea database

----- node0_0_CPU0 -----
tunnel ifhandle 0x80022cc
tunnel source 161.115.1.2
tunnel destination address group Test_IPv4
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

```

Hashing for load balancing

Hashing is a technique used by network devices such as routers and switches to distribute traffic flows or packets efficiently across multiple available paths, links, or resources. Hashing technique

- ensures all packets from the same traffic flow (e.g., TCP connection) follow the same path, avoiding out-of-order delivery
- distributes multiple traffic flows across all paths thus preventing overload of any single path, and
- works automatically as paths or links are added or removed.

How hashing works

1. Field extraction:

The device extracts certain fields from the packet header (e.g., source/destination IP addresses, ports, protocol).

2. Hash function:

These fields are combined and fed into a mathematical function called a *hash function*. This function produces a fixed-size output known as the *hash value*.

3. Path selection:

The resulting hash value is used to select one of several possible paths, links, or next hops. For example, if there are 4 links in a bundle, the device might use:

```
path = hash_value % 4
```

Here, % indicates the modulus operation.

If multiple routers use the same hash algorithm and field selection, the same flows may be mapped to the same paths on each router, resulting in polarization. This can lead to uneven traffic distribution and underused network capacity.

By increasing the variety of hash profiles using extended entropy profiles, each router hashes differently thus reducing the chance of polarization while improving the overall load balancing. For more information, see [Enhanced hashing functions using extended profiles](#).

Configure hash rotation value

These configurations control the `node-id` (also referred to as `HASH_ROTATE`), which influences how the hashing algorithms process input fields, effectively reordering them to de-correlate flows. This is a mechanism to combat polarization.

Procedure

Step 1 Configure hash rotation value using one of these options.

- Set a specific `node-id` (hash rotation value) for the entire chassis as this `node-id` determines one of the 216 possible permutations of input fields for the hashing algorithm.

```
Router# config
Router (config)# cef platform load-balancing algorithm adjust 10
```

A specific value might be chosen to optimize load balancing for a known network topology or to address observed polarization issues.

- Set a specific `node-id` (hash rotation value) for a particular NPU instance on a specific line card, location `0/0/CPU0` and instance `0` to identify the NPU as this `node-id` provides granular control, allowing different NPUs within the same chassis to use distinct hash rotation values.

```
Router# config
Router (config)# cef platform load-balancing algorithm adjust 20 instance 0 location 0/0/CPU0
```

This configuration can be useful in complex scenarios where traffic characteristics vary significantly across different NPU-handled interfaces.

- Enable the system to automatically determine and set the `node-id` (hash rotation value) globally for the entire chassis as this configuration aims to dynamically select an optimal `node-id` to improve load balancing and minimize polarization without requiring manual intervention.

```
Router# config
Router (config)# cef platform load-balancing algorithm adjust auto-global
```

- Enable the system to automatically determine and set the `node-id` (hash rotation value) for each individual NPU instance across all NPUs in the chassis as it allows each NPU to independently optimize its hash rotation based on its specific traffic load and characteristics.

```
Router# config
Router (config)# cef platform load-balancing algorithm adjust auto-instance
```

Step 2 Execute this command to view the `node-id` (hash rotation value) specified on the location, `0/0/CPU0`.

Example:

```
Router(config)# show running-config | incl cef platform load-balancing
Tue Nov 18 07:14:00.722 UTC
cef platform load-balancing algorithm adjust 20 instance 0 location 0/0/CPU0
Router(config)#
```

Hashing functions for load balancing

A hashing function is a network algorithm that

- computes a hash value based on selected user-defined packet header fields such as source IP, destination IP, source port, destination port, and next header or protocol
- enables efficient distribution of traffic flows across network paths, and
- minimizes traffic polarization by supporting varied entropy profiles.

Table 54: Feature History Table

Feature Name	Release Information	Description
Enhanced hashing functions using extended entropy profiles	Release 25.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: Q200]); Modular Systems (8800 [LC ASIC: Q200])</p> <p>This enhancement significantly reduces the risk of traffic polarization thus ensuring more even and efficient distribution of traffic across multiple network paths in large-scale networks.</p> <p>Load balancing hashing functions have been improved with extended entropy profiles that generate thousands of unique hash function selections on certain user-defined header fields.</p> <p>This feature introduces these changes:</p> <p>CLI:</p> <ul style="list-style-type: none"> • A new keyword, extended-entropy, has been added to the cef platform load-balancing command. • Two new keywords, ecmp-seed and spa-seed, are added to the cef platform load-balancing algorithm adjust command. • Two new keywords, hash and ip-field-duplication, are added to the hw-module cef command.

Addressing traffic polarization with extended entropy profiles

In large-scale networks, optimal load balancing is critical. Routers use hash algorithms to distribute traffic flows across multiple paths. However, if the hashing functions on consecutive routers are too similar, this can cause polarization—many flows get mapped to the same path on multiple routers, reducing network efficiency and resilience.

Starting with Cisco IOS XR version 25.4.1, load balancing hashing functions have been improved with extended entropy profiles. These profiles utilize specific packet header fields, such as IP addresses and port numbers, to generate thousands of unique hash function combinations. This feature introduces a new **extended-entropy** keyword in the **cef platform load-balancing** command to configure the extended entropy.

Configuring ECMP and SPA seed values

You can configure 16-bit *seed values*² for ECMP and SPA (System port aggregate), which is an internal load balancing component. To set the seed values for ECMP and SPA, use the **cef platform loading-balancing algorithm adjust ecmp-seed** and **cef platform loading-balancing algorithm adjust spa-seed** commands respectively.

Enabling ASIC property for single IP header traffic

We recommend you to enable the **hw-module profile cef hash ip-field-duplication** command when using **cef platform load-balancing extended-entropy**. This setting activates an ASIC property that enhances load balancing for traffic with a single IP header. Without this configuration, certain devices may not detect all varying bits in IPv4 or IPv6 plain traffic, potentially causing traffic polarization on some nodes.

How enhanced hashing functions for load balancing work

Summary

The key components involved in the process are:

- **Algorithm adjustments:** Modify the load balancing algorithms to utilize additional entropy from inner header fields during hash calculation.
- **Extended entropy profiles:** Define two sets of offsets and widths for fields within IPv4 or IPv6 inner headers, increasing the available entropy for hashing.

The **cef platform load-balancing extended-entropy** configuration leverages the variation or entropy used in the inner header fields and incorporates this information into the outer header's hash calculation. This approach is beneficial when the inner headers have sufficient entropy, but the outer header does not vary.

- **ECMP and SPA seed values:** Use specific seed values to further diversify the hash results, ensuring unique traffic distribution across routers.

Enhanced hashing functions for load balancing optimize how traffic is distributed across network paths by incorporating greater entropy from packet headers. This process leverages extended entropy profiles, algorithm adjustments, and specific ECMP and SPA seed values to create highly diversified hash results, minimizing the risk of traffic collisions while promoting balanced network utilization.

² Seed values introduce an additional element of randomness into the hashing process, further helping to de-correlate flows and prevent polarization

Workflow

These stages describe the enhanced hashing functions for load balancing:

1. **Profile selection:** The system selects one of 256 available extended-entropy profiles, each specifying which fields within the inner packet header are used for entropy.
2. **Hash input extraction:** The selected profile identifies two sets of offsets and widths in the IPv4 or IPv6 inner header, and extracts these field values as hash input.
3. **Algorithm application:** The load balancing algorithm incorporates the extracted entropy, along with ECMP and SPA seed values, into the hash calculation.
4. **Hash calculation:** The router computes a hash value based on the combined entropy from the inner header and the seed values.
5. **Traffic distribution:** The computed hash value determines how each packet flow is distributed across available network paths, ensuring varied and balanced load sharing.

Result

The enhanced hashing process produces highly unique hash inputs, enabling routers to split traffic flows more effectively and consistently achieve balanced utilization of all network paths, even when outer packet headers lack sufficient variability.

Benefits of enhanced hashing functions

These are some benefits of the enhanced hashing functions using extended entropy profiles:

- Improves traffic distribution across paths and devices, even with random node-id assignments.
- Suitable for very large, multi-router, multi-path networks.
- Supports a range of configurations and traffic types.

Configuration guidelines for enhanced hashing functions

- Do not configure both **cef platform load-balancing extended-entropy**, and **cef platform load-balancing fields user-data** at the same time. These configurations are mutually exclusive.
- When using **cef load-balancing algorithm adjust auto-instance** on adjacent devices with small and symmetric topologies, use **cef load-balancing algorithm adjust auto-global** to avoid uneven load balancing.
- Do not configure the same seed value on adjacent devices or within the same topology.
- Enable **cef platform load-balancing extended-entropy** before enabling **hw-module profile cef hash ip-field-duplication**.

Restrictions for enhanced hashing functions

Only Cisco Silicon One Q200-based systems support the enhanced hashing functions for load balancing.

Configure ECMP and SPA seed values

Follow this procedure to configure seed values for ECMP and SPA hashing.

By default, the hashing algorithm uses a 16-bit seed from the router ID, providing differentiation across devices.

You can customize ECMP or SPA seeds to introduce randomness into the hash function. To avoid load-balancing issues, configure different seed values on adjacent devices or within the same topology.

Procedure

Step 1 Set a specific hexadecimal ECMP seed value (0xaa in this example) for the chassis to ensure that the traffic is distributed consistently and without bias across multiple equal-cost paths, preventing polarization.

Example:

```
Router# config
Router (config)# cef platform load-balancing algorithm adjust ecmp-seed 0xaa
Router (config)# commit
```

Step 2 Set a specific hexadecimal SPA seed value (0xbb in this example) for the chassis so that the seed is incorporated into the hashing calculation to add an element of uniqueness, which can help distribute traffic more evenly across multiple paths.

Example:

```
Router# config
Router (config)# cef platform load-balancing algorithm adjust spa-seed 0xbb
Router (config)# commit
```

Step 3 Set a specific hexadecimal ECMP seed value (0xdd in this example) on a particular NPU instance to provide NPU-specific control over the ECMP hashing randomness, which can be beneficial in complex network designs.

Example:

```
Router# config
Router (config)# cef platform load-balancing algorithm adjust ecmp-seed 0xdd instance 0 location 0/0/CPU0
Router (config)# commit
```

Step 4 Set a specific hexadecimal SPA seed value (0xcc in this example) on a particular NPU instance as this seed allows for fine-grained control, enabling different NPUs to use distinct SPA seed values if their traffic patterns or load-balancing requirements differ.

Example:

```
Router# config
Router (config)# cef platform load-balancing algorithm adjust spa-seed 0xcc instance 0 location 0/0/CPU0
Router (config)# commit
```

Step 5 Verify the ECMP and SPA seed value configurations.

Example:

```
Router# show running-config | include cef platform load-balancing
Tue Nov 18 07:08:56.704 UTC
cef platform load-balancing algorithm adjust spa-seed 0xcc instance 0 location 0/0/CPU0
cef platform load-balancing algorithm adjust ecmp-seed 0xdd instance 0 location 0/0/CPU0
Router#
```

What to do next

Make sure that you have configured the extended entropy profiles and algorithm adjustments for the enhanced hashing functions for load balancing to work properly.

Configure extended entropy profile

This procedure allows you to configure **cef platform load-balancing extended-entropy**. This configuration increases the number of uncorrelated hash selection algorithms by modifying how input fields are processed.

Before you begin

Make sure that you have configured the ECMP and SPA seed values and algorithm adjustments along with the extended entropy profiles configuration so that the enhanced hashing functions for load balancing work correctly.

Procedure

Step 1 Configure the extended entropy profile using one of these commands:

- Enable the system to automatically select the most suitable extended entropy profile globally for the entire chassis.

```
Router# config  
Router (config)# cef platform load-balancing extended-entropy auto-global
```

This is the recommended option.

- Enables the system to automatically select the most suitable extended entropy profile for each individual NPU instance across all NPUs in the chassis.

```
Router# config  
Router (config)# cef platform load-balancing extended-entropy auto-instance
```

This provides NPU-level optimization, allowing each NPU to dynamically choose a profile that best suits the traffic it is processing, further enhancing load balancing and polarization avoidance.

- Selects a specific extended entropy profile (identified by `profile-index 10` in this example) to be used for hashing across the entire chassis.

```
Router# config  
Router (config)# cef platform load-balancing extended-entropy profile-index 10
```

These profiles define how the hashing functions are applied to various header fields such as inner source or destination IP, TCP or UDP ports, and so on to maximize entropy and de-correlation. This allows an administrator to explicitly choose a profile optimized for a particular traffic mix or to address specific polarization issues.

Step 2 (Optional) Enable the ASIC property using the **hw-module profile cef hash ip-field-duplication** command.

Example:

```
Router# config  
Router (config)# hw-module profile cef hash ip-field-duplication  
Router (config)# commit  
Router (config)# exit  
Router# reload location all
```

Enabling the ASIC property improves load balancing for traffic with a single IP header. Without this configuration, some devices may not capture the varying bits in IPv4 or IPv6 plain traffic, which results in polarization on certain nodes.

Step 3 Execute this command to view the running configuration.

Example:

```
Router# show running-config | incl cef platform load-balancing  
Mon Nov 17 08:16:06.749 UTC
```

```
cef platform load-balancing extended-entropy auto-global
Router(config)#
```

ECMP Hashing Support for Load Balancing

The system inherently supports the n-tuple hash algorithm. The first inner header in the n-tuple hashing includes the source port and the destination port of UDP / TCP protocol headers.

The load balancing performs these functions:

- Incoming data traffic is distributed over multiple equal-cost connections.
- Incoming data traffic is distributed over multiple equal-cost connections member links within a bundle interface.
- Layer 2 bundle and Layer 3 (network layer) load-balancing decisions are taken on IPv4, and IPv6. If it is an IPv4 or an IPv6 payload, then an n-tuple hashing is done.
- An n-tuple hash algorithm provides more granular load balancing and used for load balancing over multiple equal-cost Layer 3 (network layer) paths. The Layer 3 (network layer) path is on a physical interface or on a bundle interface.
- The n-tuple load-balance hash calculation contains:
 - Source IP address
 - Destination IP address
 - IP Protocol type
 - Router ID
 - Source port
 - Destination port
 - Input interface
 - Flow-label (for IPv6 only)

User-defined fields for ECMP hashing

Table 55: Feature History Table

Feature Name	Release Information	Description
User-defined fields for ECMP hashing	Release 25.4.1	Introduced in this release on: Fixed Systems (8700 [ASIC: K100])(select variants only*) *This feature is supported on Cisco 8711-48Z-M routers.

Feature Name	Release Information	Description
User-defined fields for ECMP hashing	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on Cisco 8011-4G24Y4H-I routers.</p>
User-defined fields for ECMP hashing	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100, K100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 8712-MOD-M • 88-LC1-12TH24FH-E • 88-LC1-36EH • 88-LC1-52Y8H-EM

Feature Name	Release Information	Description
User-defined fields for ECMP hashing	Release 24.2.11	<p>We ensure that in cases where multiple paths are used to carry packets from source to destination, each path is utilized for this purpose and no path is over-utilized or congested. This is made possible because we now provide customized ECMP hashing fields that are used for path computation.</p> <p>Previously, the router relied on fixed packet header fields for hashing, which were not user configurable. With additional user-defined bytes considered for hashing, the granularity at which the traffic can be analyzed for ECMP load balancing increases, resulting in better load balancing and path utilization.</p> <p>The feature introduces these changes:</p> <p>CLI:</p> <ul style="list-style-type: none"> • cef load-balancing fields user-data • The show cef exact-route command is modified with a new user-data keyword. • The show cef ipv4 exact-route command is modified with a new user-data keyword. • The show cef ipv6 exact-route command is modified with a new user-data keyword. <p>YANG:</p> <ul style="list-style-type: none"> • New Xpath for <code>Cisco-IOS-XR-8000-fib-platform-cfg.yang</code> (see Github, YANG Data Models Navigator).

ECMP hashing is used to distribute traffic across multiple equal-cost paths. See [ECMP Hashing Support for Load Balancing](#), on page 280 for the default static hashing algorithm details.

You can now add user-defined packet header fields for ECMP path calculation for ipv4 and ipv6 flows using the **cef platform load-balancing fields user-data** command. Ensure you specify these user-defined fields based on the type of traffic flow that requires load balancing. You can include the following parameters:

- **Hash header:** The hash header specifies which packet header is being considered for load balancing. You can enable any or all of the available six profiles.
 - IPv4: tcp, udp, non-tcp-udp
 - IPv6: tcp, udp, non-tcp-udp

If any hash header profile is defined for load balancing, along with the fixed fields considered for hashing, additional bytes in the payload are also used for path computation.

- **Hashing offset:** The hashing offset specifies the byte location from the end of the configured header.
- **Hash size:** The hash size specifies the number of bytes that is considered from the start of the hash offset by the ECMP hashing algorithm. Range is 1 to 4 bytes.
- **Location:** This specifies the location of the ingress line card that receives the incoming traffic. The user-defined hashing configuration is applied on the specified line card.

The addition of the user-defined packet header fields increases the granularity at which the traffic is analyzed for ECMP load balancing. When multiple paths with equal cost are available for routing a specific type of packet from a source to a destination, this granularity ensures that the intended type of traffic is evenly distributed across these paths. This ensures all available paths are used efficiently and prevents congestion or over-utilization of a single path.

You can also retrieve the exact-route information based on the configured user-data using the **show cef exact-route** command with **user-data** keyword.



Note

- When the user-defined hashing configuration is active, any additional options or optional keywords are disregarded during the parsing of incoming packets for retrieving the user-defined bytes.
- The hashing results based on user-defined hash feature is applicable to BGP/IGP ECMP and LAG hashing.
- The use of the user-defined hashing configuration changes the load balancing behavior of GRE and IPinIP traffic. This includes all traffic that begins with ipv4, ipv6, ipv4+udp, ipv6+udp, ipv4+tcp, and ipv6+tcp, regardless of the payload.

Configure User-Defined Fields for ECMP Hashing

The command **cef load-balancing fields user-data** configures the additional user-defined fields that are to be considered for the hashing algorithm.

This example shows how to configure the additional IPv4 header fields for TCP packets:

```
Router# configure terminal
Router(config)# cef load-balancing fields user-data ipv4 tcp offset 5 size 3 location 0/0/CPU0
Router(config)# commit
```

- offset 5: The payload considered for hashing starts from byte 6 from the end of TCP header.
- size 3: Three bytes of payload are considered.

- location 0/0/CPU0: Specifies the line card on which the configuration is applied.

In the above example, the sixth, seventh, and eighth bytes of the payload are considered additionally for the hashing.

This example shows how to configure the additional IPv6 header fields for UDP packets:

```
Router# configure terminal
Router(config)#cef load-balancing fields user-data ipv6 udp offset 0 size 2 location 0/0/CPU0
Router(config)#commit
```

- offset 0: The payload considered for hashing starts from the end of UDP header.
- size 2: Two bytes of payload are considered.
- location 0/0/CPU0: Specifies the line card on which the configuration is applied.

In the above example, the first two bytes of payload of a UDP packet are considered additionally for the hashing.

Running Configuration

The following example shows the running configuration:

```
Router#show running-config | include cef
Fri Jul 28 12:02:01.002 UTC
cef load-balancing fields user-data ipv4 tcp offset 5 size 3 location 0/0/CPU0
cef load-balancing fields user-data ipv6 udp offset 0 size 2 location 0/0/CPU0
Router#
```

Verification

The following example shows the difference in load balancing before and after applying user-defined hashing, for a flow with data that exhibits good hashing behavior.

Before applying user-defined hashing

```
Router#show interfaces accounting | i IPV6_U
Protocol          Pkts In      Chars In      Pkts Out      Chars Out
IPV6_UNICAST      1             72             0              0
IPV6_UNICAST      2            144            0              0
IPV6_UNICAST      1             72             0              0
IPV6_UNICAST      0              0            3979416       1981749168
IPV6_UNICAST     4191438      2087336124    0              0
IPV6_UNICAST      1             72             0              0
IPV6_UNICAST      1             72             0              0
IPV6_UNICAST      1             72             0              0
Router#
```

After applying user-defined hashing

```
Router#show interfaces accounting | i IPV6_U
Protocol          Pkts In      Chars In      Pkts Out      Chars Out
IPV6_UNICAST      0              0              39119         19481262
IPV6_UNICAST      0              0              39801         19820898
IPV6_UNICAST      0              0              40483         20160534
IPV6_UNICAST      0              0              40524         20180952
IPV6_UNICAST      0              0              40573         20205354
IPV6_UNICAST      0              0              40614         20225772
IPV6_UNICAST      0              0              39368         19605264
```

IPV6_UNICAST	0	0	40734	20285532
IPV6_UNICAST	0	0	40777	20306946
IPV6_UNICAST	0	0	40171	20005158
IPV6_UNICAST	0	0	40858	20347284
IPV6_UNICAST	0	0	40269	20053962
IPV6_UNICAST	0	0	41603	20718294
IPV6_UNICAST	0	0	40363	20100774
IPV6_UNICAST	0	0	40407	20122686
IPV6_UNICAST	0	0	41098	20466804
IPV6_UNICAST	850393	423495714	0	0

To view the exact route information allocated to the packets, use **show cef exact-route** command with **user-data** keyword.

The packet contains value 0x2 in the packet position for the ipv6 packet, for which the user-defined configuration has been added for a non-tcp-udp ipv6 flow.

```
Router#show cef ipv6 exact-route 100::10 60::1 flow-label 0 protocol 59 source-port 0
destination-port 0 user-data 0x2 ingress-interface HundredGigE0/0/0/2 location 0/0/cpu0
Unsupported protocol value 59
60::/16, version 1293, internal 0x1000001 0x20 (ptr 0x8b78ef00) [1], 0x400 (0x8e9cfc48),
0x0 (0x0)
Updated Aug 14 07:50:20.022
local adjacency to Bundle-Ether3.30

Prefix Len 16, traffic index 0, precedence n/a, priority 2
via Bundle-Ether3.30
via fe80::72b3:17ff:feae:d703/128, Bundle-Ether3.30, 7 dependencies, weight 0, class 0
[flags 0x0]
path-idx 7 NHID 0x0 [0x8db8bed8 0x0]
next hop fe80::72b3:17ff:feae:d703/128
local adjacency
```




CHAPTER 14

Configure Generic UDP Encapsulation

Read this section to get an overview and know how to configure the Generic UDP Encapsulation.

Table 56: Feature History Table

Feature Name	Release Information	Feature Description
Generic UDP Encapsulation	Release 25.4.1	Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*) *This feature is supported on: <ul style="list-style-type: none">• 8011-12G12X4Y-A• 8011-12G12X4Y-D
Outer IP Header-Driven Hash Computation for Incoming GUE Packets	Release 25.1.1	Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*) *This feature is supported on: <ul style="list-style-type: none">• 8712-MOD-M• 8011-4G24Y4H-I

Feature Name	Release Information	Feature Description
Outer IP Header-Driven Hash Computation for Incoming GUE Packets	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*).</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-32FH-M • 8212-48FH-M • 8711-32FH-M • 88-LC1-12TH24FH-E • 88-LC1-36EH • 88-LC1-52Y8H-EM
Outer IP Header-Driven Hash Computation for Incoming GUE Packets	Release 7.11.1	<p>We now offer you the flexibility of using only the outer IP header to calculate the hashing for incoming Generic UDP Encapsulation (GUE) packets. On enabling this feature, only the outer IP source and destination addresses are used for hashing calculations. The inner IP addresses are not considered, providing a simpler method of distribution. Previously, both inner IP and outer IP headers were used for ECMP hashing the incoming GUE packets.</p> <p>The feature introduces these changes:</p> <p>CLI:</p> <ul style="list-style-type: none"> • hw-module profile gue underlay-hash <p>YANG Data Models:</p> <ul style="list-style-type: none"> • New XPath for Cisco-IOS-XR-npu-hw-profile-cfg.yang (see GitHub, YANG Data Models Navigator) <p>The command is supported on Q200-based ASICs.</p>

Feature Name	Release Information	Feature Description
Generic UDP Encapsulation	Release 7.3.1	<p>This feature enables you to add an additional header to packets to identify or authenticate the data using UDP. Encapsulating packets in UDP leverages the use of the UDP source port to provide entropy to Equal Cost Multi-Path (ECMP) hashing. It provides significant performance benefits for load-balancing.</p> <p>This command is introduced for this feature:</p> <p>decapsulate gue</p>

- [Understand UDP encapsulation, on page 289](#)
- [Generic UDP encapsulation, on page 289](#)
- [Flexible Assignment of UDP Port Numbers for Decapsulation, on page 299](#)
- [GUEv1 static tunnel configuration over IPv4 networks, on page 307](#)

Understand UDP encapsulation

UDP encapsulation is a technique of adding network headers to 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 handle only the outer Source IP address and parts of the destination address. They may not provide sufficient load balancing entropy.

Generic UDP encapsulation

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, which 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, and
- congestion control data.

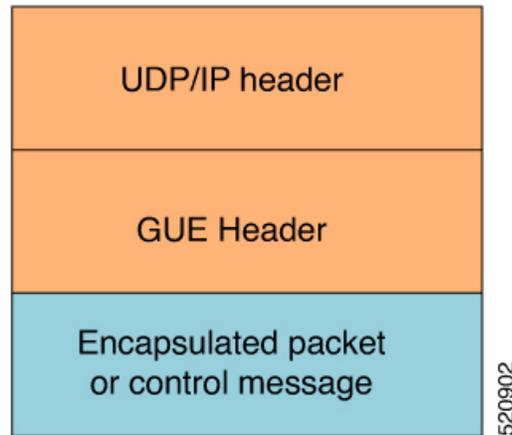
Table 57: Feature History Table

Feature Name	Release Information	Feature Description
Single UDP port for IPv4 and IPv6 packets	Release 25.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: Q200])</p> <p>This feature enhances memory and network performance by using a single UDP port for both IPv4 and IPv6 packets in GUE variant 1 UDP encapsulation and decapsulation.</p> <p>The default UDP port for IPv4 and IPv6 is 6080.</p> <p>As part of this enhancement, the hw-module profile gue udp-dest-port command is deprecated.</p> <p>This feature introduces these changes:</p> <p>CLI:</p> <ul style="list-style-type: none"> • nve overlay-encap guev1, • show cef global udp-ports gue-v1
Generic UDP Decapsulation for IPv6 Traffic	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8712-MOD-M • 8011-4G24Y4H-I
Generic UDP Decapsulation for IPv6 Traffic	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-12TH24FH-E • 88-LC1-36EH • 88-LC1-52Y8H-EM
Generic UDP Decapsulation for IPv6 Traffic	Release 24.1.1	<p>We've now made it possible for you to decapsulate GUEv6 packets by adding an additional header to packets that identifies or authenticates the data by using User Datagram Protocol (UDP).</p> <p>In GUE, the payload is encapsulated in an IP packet that can be an 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.</p>

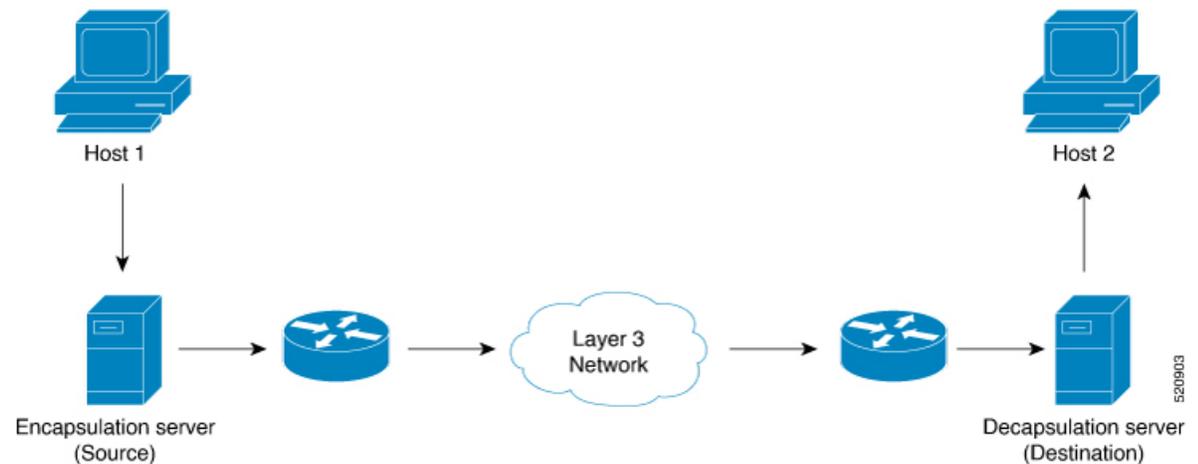
In GUE, the payload is encapsulated in an IP packet that can be IPv4 or IPv6 Carrier. The UDP header is added to provide extra hashing parameters, and optional payload demultiplexing. At the decapsulation node, the Carrier IP and UDP headers are removed, and the packet is forwarded based on the inner payload.

A GUE packet has the general format:

Figure 16: GUE Packet Format



For example, in this scenario, if the data stream is sent from Host 1 to Host 2. The server acts as a GUE encapsulator that sends 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 two bits of the version number in the IP header.

Starting from Cisco IOS XR Release 25.4.1, you can use a single UDP port for both IPv4 and IPv6 packets in GUE encapsulation and decapsulation. The default UDP port for IPv4 and IPv6 is 6080. You can modify the default UDP ports using **`nve overlay-encap guev1 udp-port destination`**.



Note The **hw-module profile gue udp-dest-port** command used to configure the UDP ports individually for IPv4, IPv6, and MPLS is deprecated from this release.

Benefits of using GUE

- Allows direct encapsulation of payloads, such as 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. The destination too 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.

Benefits of single UDP port for IPv4 and IPv6 packets

- Uses single class map per IP header.
- Allocates one counter per class map.
- Uses simpler logic to reduce duplication of end-to-end objects.

Usage guidelines for GUE

- Receives IPv4 packets with the defined GUE port of 6080.
- Receives MPLS packets with the UDPO MPLS port of 6635.

Restrictions

- Supports Generic UDP Decapsulation for only variant 1.
- Range of source or destination ports is not supported.
- Range, Source, or Destination addresses are not supported, but subnet mask entries are allowed.
- 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 does not clear the interface of all its existing configurations.

- GUE for IPv6 traffic is only supported on the Cisco 8202-32FH-M router, and 88-LC0-36FH-M and 88-LC0-36FH line cards.
- GUE IPv6 decapsulation is only supported on Layer 3 ports.
- GUE IPv6 is not supported over BVI interfaces.



Note To use only outer IP header (L3 and L4) for calculating the hashing for incoming GUE packets, use the **hw-module profile gue underlay-hash enable** command. Otherwise, by default, both outer IP header (L3 and L4) and inner IP header (L3 and L4) are considered for calculating the hashing for incoming GUE packets. The **hw-module profile gue underlay-hash enable** command is currently not supported on the P100-based and Q100-based ASICs.

Configure GUE IPv4

Perform the steps to configure GUE IPv4.

Procedure

Step 1 Configure separate GUE decapsulation tunnel UDP destination port numbers for IPv4 using the **nve overlay-encap guev1** command.

Example:

```
Router# configure
Router (config-if)# nve overlay-encapsulation guev1
Router (config-nve-encap-guev1)# udp-port destination ipv4 8000
Router (config-nve-encap-guev1)# commit
```

Step 2 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.

Example:

```
Router# configure
Router (config)# class-map type traffic match-all gue-v4-udp
Router (config-cmap)# match destination-address ipv4 220.100.20.0 255.255.255.255
Router (config-cmap)# match source-address ipv4 210.100.20.0 255.255.255.255
Router (config-cmap)# match protocol udp
Router (config-cmap)# end-class-map
Router (config)# commit
```

Step 3 Configure a policy map: Define a policy map and associate the traffic class with the traffic policy.

Example:

```
Router (config)# policy-map type pbr magic-decap

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

```

Step 4 Apply the policy for each VRF, and apply this policy on all the interfaces that are part of the VRF.

Example:

```

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

```

Step 5 Execute this command to view the set of counter values accumulated for the packets that match the class-map:

Example:

```
Router# show policy-map type pbr addr-family ipv4 statistics
```

```

VRF Name:      default
Policy-Name:   pmap
Policy Type:   pbr
Addr Family:   IPv4

Class:      cmap-loop1
  Classification statistics (packets/bytes)
  Matched      :              0/0
  Transmitted statistics (packets/bytes)
  Total Transmitted :              0/0

Class:      cmap-loop6
  Classification statistics (packets/bytes)
  Matched      :              0/0
  Transmitted statistics (packets/bytes)
  Total Transmitted :              0/0

Class:      cmap-loop2
  Classification statistics (packets/bytes)
  Matched      :              0/0
  Transmitted statistics (packets/bytes)
  Total Transmitted :              0/0

Class:      cmap-loop3
  Classification statistics (packets/bytes)
  Matched      : 198325306/17849277540
  Transmitted statistics (packets/bytes)
  Total Transmitted : 198325306/17849277540

Class:      cmap-loop4
  Classification statistics (packets/bytes)
  Matched      :              0/0
  Transmitted statistics (packets/bytes)
  Total Transmitted :              0/0

```

Step 6 Execute this command to view the default configuration if the GUE variant 1 encapsulation is not configured for IPv4 packets.

Example:

```
Router# show nve global
Fri Jul 11 16:11:10.172 UTC
NVE Global details
  VNI Scope Local : No
  VxLAN l3vni bring up mode: V1
  GUEv1 UDP Destination Port (IPv4): 6080
  GUEv1 UDP Destination Port (IPv6): 6615
  GUEv1 UDP Destination Port (mpls): 6635
  Count of NVE interfaces with mpls-udp encap: 0
  Global system mac: 0033.3ebe.8f00
```

Step 7 Execute the **show cef global udp-ports gue-v1** command to view the UDP ports for IPv4, IPV6, and MPLS packets in GUE variant 1 encapsulation.

Example:

```
Router# show cef global udp-ports gue-v1
Router# show cef global udp-ports gue-v1
Fri Jul 11 16:11:05.391 UTC
UDP ports for gue-v1 [0x309c3a70f8]
  IPv4: 6080 (default)
  IPv6: 6080 (default)
  MPLS: 6635 (default)
Policy update time: Not Yet Recorded
Platform update time: Not Yet Recorded
```

Step 8 Execute the **clear vrf** command to clear the policy-map counters for each class-map rule.

Example:

```
Router# clear vrf default address-family ipv4 statistics
```

Configure GUE IPv6

Perform the steps to configure GUE IPv6.

Procedure

Step 1 Configure separate UDP port numbers for IPv4, IPv6, and MPLS using the **nve overlay-encap guev1** command.

Example:

```
Router# configure
Router (config-if)# nve overlay-encapsulation guev1
Router (config-nve-encap-guev1)# udp-port destination ipv6 9000
Router (config-nve-encap-guev1)# commit
```

Step 2 Configure a traffic class.

Example:

```
Router# configure
```

```

Router(config)# class-map type traffic match-all gue-v6-udp
Router(config-cmap)# match protocol udp
Router(config-cmap)# match destination-address ipv6 11:1:1::1/128
Router(config-cmap)# end-class-map
Router(config)# commit

```

Step 3 Define a policy map, and associate the traffic class with the traffic policy.

Example:

```

Router(config)# policy-map type pbr guev6_decap
Router(config-pmap)# class type traffic guev6_v4
Router(config-pmap-c)# decapsulate gue variant 1
Router(config-pmap-c)# exit

Router(config-pmap)# class type traffic guev6_v6
Router(config-pmap-c)# decapsulate gue variant 1
Router(config-pmap-c)# exit

Router(config-pmap)# class type traffic guev6_mpls
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

```

Step 4 Apply the policy for each VRF.

Example:

```

Router# configure
Router(config)# vrf-policy
Router(config-vrf-policy)# vrf default address-family ipv6 policy type pbr input guev6_decap
Router(config-vrf-policy)# commit

```

Step 5 Execute this command to view the set of counter values accumulated for the packets that match the class-map.

Example:

```

Router# show policy-map type pbr vrf default addr-family ipv6 statistics

```

```

VRF Name:          default
Policy-Name:       guev6_decap
Policy Type:       pbr
Addr Family:       IPv6

Class:      guev6_v4
  Classification statistics (packets/bytes)
  Matched      :          190/24320
  Transmitted statistics (packets/bytes)
  Total Transmitted :          190/24320

Class:      guev6_v6
  Classification statistics (packets/bytes)
  Matched      :          175/24150
  Transmitted statistics (packets/bytes)
  Total Transmitted :          175/24150

Class:      guev6_mpls
  Classification statistics (packets/bytes)

```

```

    Matched          :          160/22720
Transmitted statistics (packets/bytes)
    Total Transmitted :          160/22720

Class:      class-default
Classification statistics (packets/bytes)
    Matched          :          0/0
Transmitted statistics (packets/bytes)
    Total Transmitted :          0/0

```

Step 6 Execute this command to view the default configuration if the GUE variant 1 encapsulation is not configured for IPv4 packets.

Example:

```

Router# show nve global
Fri Jul 11 16:11:10.172 UTC
NVE Global details
  VNI Scope Local : No
  VxLAN l3vni bring up mode: V1
  GUEv1 UDP Destination Port (IPv4): 6080
  GUEv1 UDP Destination Port (IPv6): 6615
  GUEv1 UDP Destination Port (mpls): 6635
  Count of NVE interfaces with mpls-udp encap: 0
  Global system mac: 0033.3ebe.8f00

```

Step 7 Execute the `show cef global udp-ports gue-v1` command to view the UDP ports for IPv4, IPV6, and MPLS packets in GUE variant 1 encapsulation.

Example:

```

Router# show cef global udp-ports gue-v1
Router# show cef global udp-ports gue-v1
Fri Jul 11 16:11:05.391 UTC
UDP ports for gue-v1 [0x309c3a70f8]
  IPv4: 6080 (default)
  IPv6: 6080 (default)
  MPLS: 6635 (default)
Policy update time: Not Yet Recorded
Platform update time: Not Yet Recorded

```

Outer IP Header-Driven Hash Computation for Incoming GUE Packets

When multiple paths with the same cost are available for forwarding traffic, ECMP hashing is used to determine the path to select for each packet. Each packet that needs to be forwarded is processed using a hashing algorithm. The hashing algorithm considers specific packet fields such as source IP, destination IP, source port, and destination port, and generates a hash value. The generated hash value is then mapped to one of the available paths. The selected path is used to forward the packet to its destination. The goal is to distribute the traffic evenly across the available paths to prevent congestion and utilize the network resources efficiently.

Now you can use only the outer IP header (L3 and L4) for calculating the hash value for incoming GUE packets and completely ignore the usage of the inner IP header. This functionality is configurable using the CLI command `hw-module profile gue underlay-hash`. This is supported for both GUE termination (decapsulation) and GUE transit (pass-through) nodes. By default, the feature is disabled; that is, both outer IP header (L3 and L4) and inner IP header (L3 and L4) are used for calculating the hashing for GUE packets.

Benefits

- **Load Balancing Efficiency:** By hashing only on the outer IP and L4 information, the packets with the same source and destination IP addresses and L4 ports consistently follow the same path in a load-balanced environment. This helps maintain session affinity or stickiness, as the inner IP addresses or L4 port numbers may change dynamically within the encapsulated packets.
- **Network Security:** Ignoring the inner IP helps preserve privacy and confidentiality within the encapsulated packets. By focusing on the outer IP and L4 headers, the network device does not have visibility into the inner IP addressing scheme or the specific content encapsulated within the packet, which enhances security.
- **Network Scalability:** Ignoring the inner IP reduces the complexity and overhead of packet processing, improving overall network performance and scalability, especially in high-throughput environments.

Configure Outer IP Header-Driven Hash Computation for Incoming GUE Packets

This section describes how to configure hashing with only outer IP for GUE packets.

Configuration Example

Use the following configuration to enable hashing with only outer IP for GUE packets:

```
Router# configure
Router# hw-module profile gue underlay-hash enable
Router# commit
```

Running Configuration

```
RP/0/RP0/CPU0:ios(config)#show running-config
hw-module profile gue underlay-hash enable
end
```

Verification

Following is the show command output before enabling hashing with only outer IP for GUE packets.

```
RP/0/RP0/CPU0:ios#show dpa objects sys location 0/RP0/CPU0 | include gue
uint32_t gue_ipv4_port => 0
uint32_t gue_ipv6_port => 0
uint32_t gue_mpls_port => 0
ofa_bool_t gue_underlay_hash => FALSE
```

Following is the show command output after enabling hashing with only outer IP for GUE packets.

```
RP/0/RP0/CPU0:ios#show dpa objects sys location 0/RP0/CPU0 | include gue
uint32_t gue_ipv4_port => 0
uint32_t gue_ipv6_port => 0
uint32_t gue_mpls_port => 0
ofa_bool_t gue_underlay_hash => TRUE
```

Flexible Assignment of UDP Port Numbers for Decapsulation

Table 58: Feature History Table

Feature Name	Release Information	Feature Description
Flexible Assignment of UDP Port Numbers for Decapsulation	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8712-MOD-M • 8011-4G24Y4H-I
Flexible Assignment of UDP Port Numbers for Decapsulation	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-12TH24FH-E • 88-LC1-36EH • 88-LC1-52Y8H-EM

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 <port number> ipv6 <port number> mpls <port number></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. Define different port numbers for IPv4, IPv6, and MPLS.

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.

Restrictions

The following restrictions are applicable while configuring unique GUE destination port numbers to decapsulate IPv4, IPv6, and MPLS packets using UDP:

- While configuring the tunnel, select one of the following:
 - Match only 16 unique source IP addresses as shown in the example:


```
Router(config-cmap)#match source-address ipv4 210.100.20.0 255.255.255.255
```
 - Match a combination of 64 unique source and destination IP addresses as shown in the example:


```
Router(config-cmap)# match destination-address ipv4 220.100.20.0 255.255.255.255
Router(config-cmap)# match source-address ipv4 210.100.20.0 255.255.255.255
```
- The Classless Inter-Domain Routing (CIDR) value in the source IP address subnet mask must be only /32.
- The destination address subnet mask supports all CIDR values. However, the destination address along with the subnet mask must be unique for all the three UDP payload types—IPv4, IPv6, and MPLS. The configuration fails when the destination IP address and the subnet mask are the same for all three payloads as seen in this example:

```
Router(config)#class-map type traffic match-all SRTE-GUE-DECAP-IPv4
Router(config-cmap)#match destination-address ipv4 10.216.101.0 255.255.255.255
..
Router(config)#class-map type traffic match-all SRTE-GUE-DECAP-IPv6
Router(config-cmap)#match destination-address ipv4 10.216.101.0 255.255.255.255
..
Router(config)#class-map type traffic match-all SRTE-GUE-DECAP-MPLS
Router(config-cmap)#match destination-address ipv4 10.216.101.0 255.255.255.255
..
```

Configuring Port Numbers for Decapsulation

By configuring different port numbers on the destination router, you can match and direct traffic to different paths. For example, traffic for a specific video service can be decapsulated and sent through different ports. The steps that are involved in configuring port numbers for decapsulation are:

1. Configure the UDP destination ports for decapsulation of the required payloads.
2. Configure the traffic class to match the ports.
3. Define a policy map, and associate the traffic class with the traffic policy.

4. Apply the policy for each VRF.



Note For the hardware module flexible port configuration to take effect you must reload the line card.

Configuration Example

```

Hw-module configuration:
=====
Router# configure
Router# hw-module profile gue udp-dest-port ipv4 1001 ipv6 1002 mpls 1003

Class-map configuration:
=====
Router# configure
Router(config)# class-map type traffic match-all udp-v4
Router(config-cmap)# match protocol udp
Router(config-cmap)# match source-address ipv4 210.100.20.0 255.255.255.255
Router(config-cmap)# match destination-address ipv4 220.100.20.0 255.255.255.255
Router(config-cmap)# match destination-port 1001
Router(config-cmap)# end-class-map
Router(config)# commit

Router(config)# class-map type traffic match-all udp-v6
Router(config-cmap)# match protocol udp
Router(config-cmap)# match destination-address ipv4 220.100.20.0 255.255.255.255
Router(config-cmap)# match source-address ipv4 210.100.20.0 255.255.255.255
Router(config-cmap)# match destination-port 1002
Router(config-cmap)# end-class-map
Router(config)# commit

Router(config)# class-map type traffic match-all udp-mpls1
Router(config-cmap)# match protocol udp
Router(config-cmap)# match destination-address ipv4 220.100.20.0 255.255.255.255
Router(config-cmap)# match source-address ipv4 210.100.20.0 255.255.255.255
Router(config-cmap)# match destination-port 1003
Router(config-cmap)# end-class-map
Router(config)# commit

Ingress Policy-map configuration:
=====
Router(config)# policy-map type pbr magic-decap
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

```

```

Applying policy per VRF:
=====
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

```

Running Configuration

```

!! File saved at 16:01:32 UTC Mon Feb 07 2022 by cisco
!! IOS XR Configuration 7.3.3.10I
!! Last configuration change at Mon Feb 7 15:35:11 2022 by cisco
!
logging console disable
username cisco
  group root-lr
  group cisco-support
  secret 10
$6$gHKmELYZAo71BE1.$3KYogrVodJxTRPZgYPGXUXkO4PqQMr2E6oYvJO4ngBmuaGsF2nAB/m1NP5I13zh9HTzBI/k4r8PwWSbsARsmp.
!
vrf vrf-gre
  address-family ipv4 unicast
  !
  address-family ipv6 unicast
  !
!
line console
  exec-timeout 0 0
  absolute-timeout 0
  session-timeout 0
!
line default
  exec-timeout 0 0
  absolute-timeout 0
  session-timeout 0
!
!arp vrf default 29.0.1.2 0000.1122.2929 ARPA
call-home
  service active
  contact smart-licensing
  profile CiscoTAC-1
    active
  destination transport-method http
!
!
ipv6 access-list abf6-gre
  1 permit ipv6 any any nexthop1 ipv6 201:0:1::2
!
ipv4 access-list abf-gre
  1 permit ipv4 any any nexthop1 ipv4 201.0.1.2
!
class-map type traffic match-all udp-v4
  match destination-address ipv4 220.100.20.0 255.255.255.255
  match source-address ipv4 210.100.20.0 255.255.255.255
  match protocol udp
  match destination-port 1001
  end-class-map
!
class-map type traffic match-all udp-v6
  match destination-address ipv4 220.100.20.0 255.255.255.255
  match source-address ipv4 210.100.20.0 255.255.255.255
  match protocol udp

```

```

    match destination-port 1002
  end-class-map
!
class-map type traffic match-all udp-mpls1
  match destination-address ipv4 220.100.20.0 255.255.255.255
  match source-address ipv4 210.100.20.0 255.255.255.255
  match protocol udp
  match destination-port 1003
end-class-map
!
policy-map type pbr pbr-gre
  class type traffic class-default
    redirect ipv4 nexthop 202.0.1.2
  !
end-policy-map
!
policy-map type pbr magic-decap
  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
!
interface Bundle-Ether25
  ipv4 address 25.0.1.1 255.255.255.0
  ipv6 address 25:0:1::1/64
  ipv6 enable
  shutdown
!
interface Bundle-Ether28
  ipv4 address 28.0.1.1 255.255.255.0
!
interface Loopback0
  ipv4 address 10.10.10.1 255.255.255.255
!
<output truncated>
interface MgmtEth0/RP0/CPU0/0
  ipv4 address dhcp
!
interface MgmtEth0/RP1/CPU0/0
  ipv4 address dhcp
!
interface BVI23
  ipv4 address 23.0.1.1 255.255.255.0
  ipv6 address 23:0:1::1/64
  ipv6 enable
  shutdown
!
interface BVI29
  ipv4 address 29.0.1.1 255.255.255.0
  ipv6 enable
  shutdown
!
interface HundredGigE0/0/0/0
  shutdown
!

```

```
<output truncated>
l2transport
!
!
interface HundredGigE0/0/0/24
 service-policy type pbr input pbr-gre
 ipv4 address 24.0.1.1 255.255.255.0
 ipv6 address 24:0:1::1/64
 ipv6 enable
!
interface HundredGigE0/0/0/24.24
 ipv4 address 24.0.24.1 255.255.255.0
 ipv6 enable
 encapsulation dot1q 24
!
interface HundredGigE0/0/0/25
 bundle id 25 mode on
!
interface HundredGigE0/0/0/26
 ipv4 address 26.0.1.1 255.255.255.0
 ipv6 address 26:0:1::1/64
 ipv6 enable
!
interface HundredGigE0/0/0/27
 ipv4 address 27.0.1.1 255.255.255.0
 ipv6 enable
!
interface HundredGigE0/0/0/27.27
 ipv4 address 27.0.27.1 255.255.255.0
 ipv6 address 27:0:27::1/64
 ipv6 enable
 shutdown
 encapsulation dot1q 27
!
interface HundredGigE0/0/0/28
 bundle id 28 mode active
!
interface HundredGigE0/0/0/29
 ipv4 address 29.0.1.1 255.255.255.0
 ipv6 enable
!
<output truncated>
interface HundredGigE0/1/0/24
 ipv4 address 124.0.1.1 255.255.255.0
 ipv6 address 124:0:1::1/64
 ipv6 enable
!
<output truncated>
!
interface HundredGigE0/1/0/30
 bundle id 28 mode active
!
interface HundredGigE0/1/0/31
 ipv4 address 31.0.1.1 255.255.255.0
 ipv6 address 31:0:1::1/64
 shutdown
!
<output truncated>
!
route-policy pass
 pass
end-policy
!
router static
```

```

address-family ipv4 unicast
 201.0.1.0/24 tunnel-ip1
 201.0.1.0/24 tunnel-ip2
 201.0.1.0/24 tunnel-ip3
 201.0.1.0/24 tunnel-ip4
!
address-family ipv6 unicast
 201:0:1::/64 tunnel-ip1
 201:0:1::/64 tunnel-ip2
 201:0:1::/64 tunnel-ip3
 201:0:1::/64 tunnel-ip4
!
!
router ospf 10
  router-id 1.1.1.1
  area 0
    ! interface Bundle-Ether28
    interface Loopback0
    !
    interface HundredGigE0/0/0/26
    !
    !
    !
    ! interface HundredGigE0/0/0/27
    ! interface HundredGigE0/0/0/27.27
router bgp 200
  bgp router-id 1.1.1.1
  address-family ipv4 unicast
    maximum-paths ibgp 64
  !
  ! redistribute connected
  ! neighbor 26.0.1.2
  ! remote-as 200
  ! address-family ipv4 unicast
  ! multipath
  ! route-policy pass in
  ! route-policy pass out
  ! next-hop-self
  neighbor 27.0.1.2
  remote-as 200
  address-family ipv4 unicast
    multipath
    route-policy pass in
    route-policy pass out
    next-hop-self
  !
  !
  neighbor 28.0.1.2
  remote-as 200
  address-family ipv4 unicast
    multipath
    route-policy pass in
    route-policy pass out
    next-hop-self
  !
  !
  neighbor 29.0.1.2
  remote-as 200
  address-family ipv4 unicast
    multipath
    route-policy pass in
    route-policy pass out
    next-hop-self
  !

```

```

!
!
vrf-policy
 vrf default address-family ipv4 policy type pbr input magic-decap
!
l2vpn
 bridge group bg
  bridge-domain bd
  ! interface HundredGigE0/0/0/29
  !   static-mac-address 0000.1122.2929
  !   routed interface BVI29
 bridge group bgl
  bridge-domain bdl
  interface HundredGigE0/0/0/23
  static-mac-address 0000.1122.2323
  !
  routed interface BVI23
  !
!
!
mpls static
 interface HundredGigE0/0/0/24
 lsp gre
  in-label 35001 allocate per-prefix 202.0.1.2/32
 forward
  path 1 nexthop tunnel-ip1 out-label 35002
  path 2 nexthop tunnel-ip2 out-label 35002
  !
!
!
ssh server vrf default
hw-module profile gue udp-dest-port ipv4 1001 ipv6 1002 mpls 1003
end

```

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

```

GUEv1 static tunnel configuration over IPv4 networks

A GUEv1 static tunnel is a user-specified tunnel that

- is configured between two endpoints using static route configuration that includes a next-hop IP address.
- enables efficient, scalable, and secure deployment of GUEv1 tunnels, and
- carries IPv4 and IPv6 traffic over IPv4 networks.

Table 59: Feature History Table

Feature Name	Release Information	Feature Description
GUEv1 static tunnel configuration over IPv4 networks	Release 25.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: Q200]); Centralized Systems (8600 [ASIC:Q200]); Modular Systems (8800 [LC ASIC: Q200])</p> <p>This feature enables highly efficient, scalable, and secure deployment of Generic UDP Encapsulation variant 1 (GUEv1) static tunnels to carry IPv4 and IPv6 traffic over IPv4 networks.</p> <p>In addition, this feature enables the router to use GUEv1 as the overlay encapsulation type for network virtualization endpoint (NVE) and send encapsulated overlay traffic to IPv4, IPv6, and MPLS destinations using UDP ports.</p> <p>With this feature, you can also increase the GUEv1 scale up to 12K tunnels.</p> <p>This feature introduces these changes:</p> <p>CLI:</p> <ul style="list-style-type: none"> The address-family ipv4 unicast and address-family ipv6 unicast commands are modified to include these keywords: remote-next-hop, tunnel, source-ip, dscp, and ttl. A new command, nve overlay-encap guev1 udp-port destination, has been introduced.

Starting from Cisco IOS XR Release 25.4.1, you can configure GUEv1 static tunnels to carry IPv4 and IPv6 traffic over IPv4 networks.

Configuration guidelines for GUEv1 static tunnels

These are some guidelines to configure GUEv1 static tunnels:

- Configure **hw-module profile cef iptunnel scale** to increase the GUEv1 tunnel scale up to 12K tunnels. This tunnel scale may vary based on the number of ECMP tunnels used.
- By default, up to 3500 GUEv1, MPLS over UDP, and IP-in-IP tunnels including the number of L3 interfaces are supported.
- Configure the **nve overlay-encap guev1 udp-port destination** command for IPv4, IPv6, and MPLS to send encapsulated overlay traffic to IPv4, IPv6, and MPLS destinations using UDP ports 6080, 6615, and 6635 respectively.

Restrictions for GUEv1 static tunnels

These are some restrictions specific to the GUEv1 static tunnel configuration:

- Traceroute over GUEv1 static tunnels is not supported both in uniform and pipe modes.

- Non-default VRF is not supported for static GUEv1 static tunnels.

Configure IPv4 and IPv6 GUEv1 static tunnels

This procedure provides the steps to configure the IPv4 and IPv6 GUEv1 static tunnels.

Procedure

Step 1 Execute the `prefix mask remote-next-hop` command to configure the IPv4 or IPv6 GUEv1 static tunnels.

Example:

This example provides the IPv4 GUEv1 static tunnel configuration.

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# router static
RP/0/RP0/CPU0:router(config-static)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-static-afi)# 9.8.7.6/32 remote-next-hop 5.5.5.6 tunnel gue-v1 source-ip
 1.1.1.1 dscp 46 ttl 250
RP/0/RP0/CPU0:router(config-static-afi)# commit
```

Example:

This example provides the IPv6 GUEv1 static tunnel configuration.

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# router static
RP/0/RP0/CPU0:router(config-static)# address-family ipv6 unicast
RP/0/RP0/CPU0:router(config-static-afi)# 11:1:1::1/128 remote-next-hop 10.10.10.11 tunnel gue-v1
source-ip 2.2.2.2 dscp 8 ttl 250
RP/0/RP0/CPU0:router(config-static-afi)# commit
```

Step 2 Execute the `show running-config router static` command to view the running configuration for the GUEv1 static tunnel.

```
RP/0/RP0/CPU0:router# show running-config router static
Thu Mar 20 20:12:16.055 IST
router static
  address-family ipv4 unicast
    9.8.7.6/32 remote-next-hop 5.5.5.6 tunnel gue-v1 source-ip 1.1.1.1 dscp 46 ttl 250
  address-family ipv6 unicast
    11:1:1::1/128 remote-next-hop 10.10.10.11 tunnel gue-v1 source-ip 2.2.2.2 dscp 8 ttl 25
!
```

Step 3 Execute the `show static afi-all topology` to verify the IPv4 and IPv6 static routes.

Example:

```
RP/0/0/CPU0:router# show static afi-all topology
Thu Mar 20 20:13:29.394 IST
IPv4 Unicast:
-----
VRF: default Table Id: 0xe0000000 AFI: IPv4 SAFI: Unicast
Prefix/Len Interface Nexthop Object
Explicit-path Metrics Local-Label
127.127.127.8/32 None 11.11.11.2 None
None [0/0/1/0/0] No label

IPv6 Unicast:
-----
```

```

VRF: default Table Id: 0xe0800000 AFI: IPv6 SAFI: Unicast
Prefix/Len Interface Nexthop Object
Explicit-path Metrics Local-Label
127:127:127::8/128 None 11.11.11.2 None
None [0/0/1/0/0] No label

```

What to do next

Make sure that you configure the global UDP ports for GUEv1 static tunnels.

Configure global UDP ports for GUEv1 static tunnels

This configuration enables the router to use GUEv1 as the overlay encapsulation type for NVE and to send encapsulated overlay traffic to IPv4, IPv6, and MPLS destinations using UDP ports 6080, 6615, and 6635 respectively.

Follow these steps to configure global UDP ports for GUEv1 static tunnels.

Before you begin

Make sure that you have configured GUEv1 static tunnels.

Procedure

Step 1 Set the destination UDP port for the encapsulated packets to port 6080 on IPv4.

```

RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# nve overlay-encap guev1 udp-port destination ipv4 6080
RP/0/RP0/CPU0:router(config)# commit

```

Step 2 Set the destination UDP port for the encapsulated packets to port 6615 on IPv6.

Example:

```

RP/0/RP0/CPU0:router(config)# nve overlay-encap guev1 udp-port destination ipv6 6615
RP/0/RP0/CPU0:router(config)# commit

```

Step 3 Set the destination UDP port for the encapsulated packets to port 6635 for MPLS.

Example:

```

RP/0/RP0/CPU0:router(config)# nve overlay-encap guev1 udp-port destination mpls 6635
RP/0/RP0/CPU0:router(config)# commit

```

Step 4 Execute **show running-config nve overlay-encap guev1** command to verify the configured UDP ports.

Example:

```

RP/0/RP0/CPU0:router# show running-config nve overlay-encap guev1
Wed Nov 19 19:47:11.595 UTC
nve
  overlay-encap guev1
    udp-port destination ipv4 6080
    udp-port destination ipv6 6615

```

```
udp-port destination mpls 6635  
!  
!
```



CHAPTER 15

Configuring 400G Digital Coherent Optics

Table 60: Feature History Table

Feature Name	Release Information	Description
Support for DP04QSDD-ULH-A1 optical module	Release 25.2.1	<p>This release introduces support for the Cisco 400G QSFP-DD Ultra Long-Haul (ULH) coherent optical module on these line cards:</p> <ul style="list-style-type: none"> • 88-LC0-36FH • 88-LC0-36FH-M <p>The 400G QSFP-DD ULH optics are supported on even-numbered ports only. The supported port numbers are: 0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34.</p>
Extended Support for QDD-400G-ZR-S and QDD-400G-ZRP-S Optical Module	Release 25.1.1	<p>This release introduces support for the QDD-400G-ZR-S and QDD-400G-ZRP-S optical module on these line cards:</p> <ul style="list-style-type: none"> • 88-LC1-12TH24FH-E • 88-LC1-36EH
Extended Support for DP04QSDD-ER1 Optical Module	Release 24.4.1	<p>This release introduces support for the Cisco 400G Quad Small Form-Factor Pluggable Double Density (QSFP-DD) optical module DP04QSDD-ER1 on these routers and line cards:</p> <p>Routers:</p> <ul style="list-style-type: none"> • Cisco 8201-32FH • Cisco 8201-24H8FH • Cisco 8608 <p>Line cards:</p> <ul style="list-style-type: none"> • 88-LC1-36EH

Feature Name	Release Information	Description
Extended Support for DP04QSDD-HE0 Optical Module	Release 24.4.1	This release introduces support for the Cisco 400G QSFP-DD High-Power (Bright) Optical Module DP04QSDD-HE0, Ethernet Variant on these line card: <ul style="list-style-type: none"> • 88-LC1-36EH
Added Support for DP04QSDD-ER1 and DP01QSDD-ZF1 Optical Modules	Release 24.3.1	This release introduces support for the following Optical Modules on the Cisco 8711-32FH-M router: <ul style="list-style-type: none"> • DP04QSDD-ER1 - Cisco 400G Quad Small Form-Factor Pluggable Double Density (QSFP-DD) • DP01QSDD-ZF1 - Cisco 100G Quad Small Form-Factor Pluggable Double Density (QSFP-DD)
Extended Support for DP04QSDD-HE0 Optical Module	Release 24.1.1	This release introduces support for the Cisco 400G QSFP-DD High-Power (Bright) Optical Module DP04QSDD-HE0, Ethernet Variant on these routers and line cards: <p>Routers:</p> <ul style="list-style-type: none"> • Cisco 8201 • Cisco 8202 <p>Line cards:</p> <ul style="list-style-type: none"> • 8800-LC-36FH • 88-LC0-36FH-M
Extended Support for DP04QSDD-HE0 Optical Module	Release 7.10.2	From this release, the DP04QSDD-HE0 optical module is supported on these router and line cards: <p>Router:</p> <ul style="list-style-type: none"> • Cisco 8202-32FH-M <p>Line cards:</p> <ul style="list-style-type: none"> • 88-LC0-36FH
Extended Support for DP04QSDD-HE0 Optical Module	Release 7.10.1	This release introduces support for the Cisco 400G QSFP-DD High-Power (Bright) Optical Module DP04QSDD-HE0, Ethernet Variant on the Cisco 8608 router.

Feature Name	Release Information	Description
oFEC Traffic Configuration for QDD-400G-ZRP-S	Release 7.9.1	<p>New Modulation and DAC Rate traffic configurations are supported on QDD-400G-ZRP-S optical module:</p> <ul style="list-style-type: none"> • 400G-TXP-1x1-16 QAM • 4x100G-MXP-1x1-16 QAM • 3x100G-MXP-1x1-8 QAM • 2x100G-MXP-1x1-QPSK • 2x100G-MXP-1x1.25-16 QAM <p>This increases the interoperability of the QDD-400G-ZRP-S optical module across network components supporting these formats.</p>
Support for DP04QSDD-HE0 Optical Module	Release 7.9.1	<p>The Cisco 400G QSFP-DD High-Power (Bright) Optical Module is an enhanced version of the currently available QSFP-DD ZR+ Optical Module. It leverages the same operational modes but provides a major enhancement by increasing the Tx Optical Power up to +1dBm.</p> <p>From this release, the DP04QSDD-HE0 optical module is supported on the Cisco 8201-32FH and Cisco 8201-24H8FH routers.</p>
Support for QDD-400G-ZRP-S Optical Module	Release 7.9.1	<p>This release introduces support for the Cisco 400G QSFP-DD-ZRP-S Ethernet Variant on the Cisco 88-LC0-34H14FH line card.</p>

Cisco offers a range of the new 400G Digital Coherent QSFP-DD optical modules. The optical modules that are available are:

- QDD-400G-ZR-S
- QDD-400G-ZRP-S
- DP04QSDD-HE0
- DP04QSDD-HE0-A1
- DP04QSDD-ER1
- DP01QSDD-ZF1
- DP04QSDD-ULH

For more information about the optic module portfolio, see the [Cisco 400G Digital Coherent Optics QSFP-DD Optical Modules Data Sheet](#).

The 400ZR-ULH modules support C-band and L-band operation and are managed entirely via OpenConfig, YANG models. You can configure and monitor modules using gNMI/gNOI protocols.

This chapter describes various optical modules and their supported configurations. The following fixed-port routers, line cards, from the indicated Cisco IOS XR software releases, support these optical modules.

Table 61: Fixed-port routers and optical modules support by Cisco IOS XR Release

Fixed-Port Routers	Optics PID	Minimum IOS XR Software Release
Cisco 8201	QDD-400G-ZR-S	Release 7.3.15
	QDD-400G-ZRP-S	
	DP04QSDD-HE0	Release 24.1.1
	DP04QSDD-HE0-A1	Release 24.2.11
Cisco 8202	QDD-400G-ZR-S	Release 7.3.15
	QDD-400G-ZRP-S	
	DP04QSDD-HE0	Release 24.1.1
	DP04QSDD-HE0-A1	Release 24.2.11
Cisco 8711-32FH-M	DP04QSDD-ER1	Release 24.3.1
	DP01QSDD-ZF1	
Cisco 8101-32FH	QDD-400G-ZR-S	Release 7.3.2
	QDD-400G-ZRP-S	
Cisco 8201-32FH	DP04QSDD-HE0	Release 7.9.1
	DP04QSDD-HE0-A1	Release 24.2.11
Cisco 8201-24H8FH	DP04QSDD-HE0	Release 7.9.1
	DP04QSDD-HE0-A1	Release 24.2.11
Cisco 8608	DP04QSDD-HE0	Release 7.10.1
Cisco 8202-32FH-M	DP04QSDD-HE0	Release 7.10.2
	DP04QSDD-HE0-A1	Release 24.2.11
Cisco 8212-48FH-M	DP04QSDD-HE0-A1	Release 24.2.11
	DP04QSDD-HE0	Release 24.3.1

Table 62: Line cards and optical modules support by Cisco IOS XR Release

Line Cards	Optics PID	Minimum IOS XR Software Release
8800-LC-36FH	QDD-400G-ZR-S	Release 7.3.15
	QDD-400G-ZRP-S	
	DP04QSDD-HE0	Release 24.1.1
	DP04QSDD-HE0-A1	Release 24.2.11
88-LC0-36FH-M	QDD-400G-ZR-S	Release 7.3.15
	QDD-400G-ZRP-S	
	DP04QSDD-HE0	Release 24.1.1
	DP04QSDD-HE0-A1	Release 24.2.11
	DP04QSDD-ULH	Release 25.2.1
	88-LC0-36FH	QDD-400G-ZR-S
QDD-400G-ZRP-S		Release 7.3.2
DP04QSDD-HE0		Release 7.10.2
DP04QSDD-HE0-A1		Release 24.2.11
DP04QSDD-ULH		Release 25.2.1
88-LC1-36EH	DP04QSDD-HE0-A1	Release 24.2.11
	DP04QSDD-HE0	
	QDD-400G-ZR-S	Release 25.1.1
	QDD-400G-ZRP-S	Release 25.1.1
88-LC1-12TH24FH-E	DP04QSDD-HE0-A1	Release 24.2.11
	QDD-400G-ZR-S	Release 25.1.1
	QDD-400G-ZRP-S	Release 25.1.1
88-LC0-34H14FH	QDD-400G-ZRP-S	Release 7.9.1
	DP04QSDD-HE0-A1	Release 24.2.11

These restrictions apply to optics and routers:

Table 63: Restrictions on optics and routers

The...	not supported on...
QDD-400G-ZR-S and QDD-400G-ZRP-S optics are	Cisco 8102-64H fixed-port routers.
Tail Trace Identifier (TTI) feature is	QDD-400G-ZR-S and QDD-400G-ZRP-S optics.
QDD-400G-ZRP-S and DP04QSDD-HE0 optics are	odd-numbered ports of the following routers and line cards: <ul style="list-style-type: none"> • Cisco 8201 • Cisco 8202 • 8800-LC-36FH • 88-LC0-36FH-M

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 a 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 339](#)
 - [Configuring Muxponder Mode, on page 330](#)
 - [Configuring Modulation, on page 337](#)
 - [Configuring FEC, on page 341](#)
- Optical configuration – Comprises configuring frequency, chromatic dispersion, and optical transmit power. Applicable for optics controllers:
 - [Configuring Frequency, on page 324](#)
 - [Configuring Chromatic Dispersion, on page 326](#)
 - [Configuring Optical Transmit Power, on page 327](#)

- 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 346](#)
 - [Configuring PM Parameters, on page 347](#)
- Loopback configuration – Configures loopback. Applicable for coherent DSP controller:
 - [Configuring Loopback, on page 342](#)
- 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 351](#)
 - [Configuring FEC Alarm Threshold, on page 353](#)

The following table contains the possible traffic configuration values for the 400G Digital Coherent QSFP-DD optical modules, in the transponder and muxponder mode:

Table 64: 400G Digital Coherent QSFP-DD Traffic Capabilities and Example Settings

	Client Speed	Trunk Speed	Frequency	FEC	Modulation	DAC-Rate	Chromatic Dispersion (CD)	Transmitted (Tx) Power
QDD-400G-ZR-S	1x400G, 4x100G	400G	C-Band, 196.1 To 191.3 THz	cFEC	16QAM	1x1	-2400 to +2400	Each optical module has its own Tx power optimal values. You can change the Tx power value based on the module capability.

	Client Speed	Trunk Speed	Frequency	FEC	Modulation	DAC-Rate	Chromatic Dispersion (CD)	Transmitted (Tx) Power
QDD-400G-ZRP-S	1x400G, 4x100G, 3x100G, 2x100G, 1x100G Note Release 7.3.15 supports only 1x400 and 4x100 client speed.	400G, 300G, 200G, 1x100 Note Release 7.3.15 supports only 400G trunk speed.	C-Band, 196.1 To 191.3 THz	oFEC, cFEC	16QAM, 8QAM, QPSK Release 7.3.15 supports only 16QAM.	1x1 (OIF, OpenZR+ Mode without Tx shaping), 1x1.25 (Cisco Mode with Tx shaping), 1x5 (OpenZR+ Mode with Tx shaping)	Release 7.3.15: -80000 to +80000 Release 7.3.2: -160000 to +160000	Each optical module has its own Tx power optimal values. You can change the Tx power value based on the module capability.
DP04QSDD-HE0	1x400G, 4x100G, 3x100G, 2x100G, 1x100G	400G, 300G, 200G, 100G	C-Band, 196.1 To 191.3 THz	oFEC, cFEC	16QAM, 8QAM, QPSK	1x1.25 (Cisco Mode), 1x5 (OIF, OpenZR+ Mode)	400G: -52000 to +52000 100G: -160000 to +160000	Each optical module has its own Tx power optimal values. You can change the Tx power value based on the module capability.
DP04QSDD-ER1	1x400G, 2x200G, 4x100G	400G	193.7THz	cFEC, oFEC	16QAM	1x1	-2400 to +2400	Fixed at maximum output around -9 dBm.
DP01QSDD-ZF1	1x100G	100G	193.7THz	oFEC	QPSK	1x1	-2400 to +2400	Fixed at maximum output around -6 dBm.

	Client Speed	Trunk Speed	Frequency	FEC	Modulation	DAC-Rate	Chromatic Dispersion (CD)	Transmitted (Tx) Power
DP04QSDD-HE0-A1	1x400G, 4x100G, 3x100G, 2x100G, 1x100G	400G, 300G, 200G, 100G	C-Band, 196.1 to 191.3 THz	cFEC, oFEC	16QAM, 8QAM, QPSK	1x1.25 (Cisco Mode), 1x5 (OIF, OpenZR+ Mode)	400G: -52000 to +52000 100G: -160000 to +160000	Each optical module has its own Tx power optimal values. You can change the Tx power value based on the module capability.
400G Transponder oFEC OpenZR+ Mode (example settings)	No need to configure 400G transponder mode	No setting required; tied to Client Speed	client-freq 100MHz frequency <i>1931000</i>	fec <i>OFEC</i> (default; can be omitted)	modulation <i>16Qam</i> (default; can be omitted)	DAC-Rate <i>1x1.5</i>	cd-min <i>-20000</i> cd-max <i>20000</i>	transmit-power <i>10 (+1 dBm)</i>
400G Muxponder oFEC Cisco Mode (example settings)	breakout <i>4x100</i>	No setting required; tied to Client Speed	client-freq 100MHz frequency <i>1931000</i>	fec <i>OFEC</i> (default; can be omitted)	modulation <i>16Qam</i> (default; can be omitted)	DAC-Rate <i>1x1.25</i> (default; can be omitted)	cd-min <i>-20000</i> cd-max <i>20000</i>	transmit-power <i>10 (+1 dBm)</i>

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 65: 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 66: QDD-400G-ZRP-S Transponder and Muxponder Configuration Values

TXP/MXP	Client	Trunk	Modulation	FEC	DAC Rate	OpenZR+ Support
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	oFEC	1x1	
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 Client, 100G speed	1 trunk, 400G speed	16 QAM	oFEC	1x1	
4x100G-MXP	4 clients, 100G speed	1 trunk, 400G speed	16 QAM	cFEC	1x1	
3x100G-MXP	3 clients, 100G speed	1 trunk, 400G speed	8 QAM	oFEC	1x1.25	
3x100G-MXP	3 Client, 100G speed	1 trunk, 400G speed	8 QAM	oFEC	1x1	
2x100G-MXP	2 clients, 100G speed	1 trunk, 200G speed	QPSK	oFEC	1x1.50	
2x100G-MXP	2 Client, 100G speed	1 trunk, 400G speed	QPSK	oFEC	1x1	
2x100G-MXP	2 Client, 100G speed	1 trunk, 400G speed	16 QAM	oFEC	1x1.25	
1x100G-MXP	1 client, 100G speed	1 trunk, 100G speed	QPSK	oFEC	1x1.50	

The high optical performance DP04QSDD-HE0 QSFP-DD pluggable coherent optical module is developed for easy deployment in Reconfigurable Optical Add/Drop Multiplexer (ROADM) line systems.

DP04QSDD-HE0/DP04QSDD-HE0-A1 Transponder and Muxponder Configuration Values

This table lists the possible Transponder and Muxponder configuration values for the DP04QSDD-HE0 and DP04QSDD-HE0-A1 optical modules.

Table 67: DP04QSDD-HE0/DP04QSDD-HE0-A1 Transponder and Muxponder Configuration Values

TXP/MXP	Client	Trunk	Modulation	FEC	DAC Rate	MSA Support
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	1x1.5	400ZR
400G-TXP	1 client, 400G speed	1 trunk, 400G speed	16 QAM	oFEC	1x1.5	OpenZR+
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	1x1.5	400ZR
4x100G-MXP	4 clients, 100G speed	1 trunk, 400G speed	16 QAM	oFEC	1x1.5	OpenZR+
3x100G-MXP	3 clients, 100G speed	1 trunk, 300G speed	8 QAM	oFEC	1x1.25	
3x100G-MXP	3 clients, 100G speed	1 trunk, 300G speed	8 QAM	oFEC	1x1.5	OpenZR+
2x100G-MXP	2 clients, 100G speed	1 trunk, 200G speed	QPSK	oFEC	1x1.5	OpenZR+
2x100G-MXP	2 clients, 100G speed	1 trunk, 200G speed	8 QAM	oFEC	1x1.25	
2x100G-MXP	2 clients, 100G speed	1 trunk, 200G speed	16 QAM	oFEC	1x1.25	
1x100G-TXP	1 client, 100G speed	1 trunk, 100G speed	QPSK	oFEC	1x1.5	OpenZR+

- [Configuring Frequency](#), on page 324
- [Configuring Chromatic Dispersion](#), on page 326
- [Configuring Optical Transmit Power](#), on page 327
- [Configuring Muxponder Mode](#), on page 330
- [Configure 2x200G DACs with 2x200 Breakout](#), on page 332
- [Configure 100G operating modes with 200G DAC](#), on page 334
- [Configuring Modulation](#), on page 337
- [Configuring DAC Rate](#), on page 339
- [Configuring FEC](#), on page 341
- [Configuring Loopback](#), on page 342
- [Disable Auto-Squelching](#), on page 344
- [Configuring Performance Monitoring](#), on page 346

- [Configuring PM Parameters, on page 347](#)
- [Configuring Alarms Threshold, on page 351](#)
- [Configuring FEC Alarm Threshold, on page 353](#)
- [Enhanced SOP tolerance mode for ULH optics , on page 359](#)
- [Media Link-down PreFEC Degrade Enablement, on page 362](#)
- [Application select code provisioning, on page 365](#)
- [Alarms Troubleshooting, on page 371](#)

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: 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 = -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 68: Default CD Search Range

Muxponder Rate	FEC Value	Default CD Search Range (Min-Max)
400	OFEC	-26000 to +26000
400	CFEC	-2400 to +2400
300	OFEC	-50000 to +50000
200	OFEC	-50000 to +50000
100	OFEC	-80000 to +80000

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
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                 : 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 QDD-400G-ZR-S and QDD-400G-ZRP-S optical module has its own optical transmit (TX) power range. You can change the optical transmit (TX) power value based on the module capability. For "Transmitter specifications", see the [Cisco 400G Digital Coherent Optics QSFP-DD Optical Modules Data Sheet](#).

Table 69: Optical Transmit Power Values

Optical Module	Trunk Speed ^{1,3}	Optical Transmit Power (Tx) Shaping	Interval	Supported Range of Optical Transmit Power (Tx) Values (in units of 0.1dBm) ²		
				Minimum Value	Maximum Value - Typical	Maximum Value - Worst Case
DP04QSDD-ULH	400G	Yes	1	depends on the appsel configuration	depends on the appsel configuration	depends on the appsel configuration
QDD-400G-ZR-S	400G	No	1	-150	-100	-100
QDD-400G-ZRP-S	400G	Yes	1	-150	-110	-130
	300G			-150	-104	-119
	200G			-150	-90	-105
	100G			-150	-59	-75
DP04QSDD-HE0, DP04QSDD-HE0-A1	400G	Yes	1	-100	20	10
	300G					
	200G					
	100G					

¹. Release 7.3.15 supports 4x100G muxponder mode or trunk speed.

². 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.

³. Release 7.3.2 and future releases support 3x100G, 2x100G, and 1x100G muxponder modes or trunk speed.

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
  WV-OOO = 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 QDD-400G-ZR-S and QDD-400G-ZRP-S optical modules in the 400G transponder mode.

Using the **breakout muxponder mode** command, you can configure muxponder mode on optics controllers. Based on the muxponder mode, you can choose the modulation.

Muxponder mode options available for QDD-400G-ZR-S are:

- 4x100

Muxponder mode options available for QDD-400G-ZRP-S are:

- 4x100
- 3x100
- 2x100



Note Release 7.3.15 supports only 4x100 muxponder mode.

See the following tables for the modulation values, based on the muxponder mode:

- [QDD-400G-ZR-S Transponder and Muxponder Configuration Values, on page 321](#)
- [QDD-400G-ZRP-S Transponder and Muxponder Configuration Values, on page 322](#)

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/IP** 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/IP** 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
```

Configure 2x200G DACs with 2x200 Breakout

The 2x200G DAC Cable with 2x200G Breakout CLI feature allows you to manually configure 2x200G breakout interfaces when using 2x200G modules.

Table 70: Feature History Table

Feature Name	Release Information	Feature Description
Configure 2x200G DACs with 2x200 Breakout	Release 25.4.1	Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*) *This feature is supported on the Cisco 8011-32Y8L2H2FH routers.

Feature Name	Release Information	Feature Description
Configure 2x200G DACs with 2x200 Breakout	Release 25.2.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: Q200](select variants only*), Modular Systems (8800 [LC ASIC: Q200]) (select variants only*)</p> <p>This feature enables support for configuring 2x200G DAC (Direct Attach Copper) cables with a 2x200G breakout. It addresses the issue where certain 2x200G DAC cables are incorrectly detected as 400G cables and allows you to explicitly configure the 2x200G breakout using the CLI (Command-Line Interface).</p> <p>*The feature is supported on:</p> <ul style="list-style-type: none"> • 88-LC0-36FH • 88-LC0-36FH-M • 8201-32FH • 8202-32FH-M <p>This feature introduces these changes:</p> <p>CLI:</p> <p>The breakout keyword is enhanced to include 2x200 option in the controller optics command</p>

The 2x200G DAC Cable with 2x200G Breakout feature allows you to manually configure 2x200G breakout interfaces when using 2x200G modules. By default, the system might detect these modules as 400G and create 400G interfaces. This feature provides a CLI command to explicitly set the breakout configuration to 2x200G, creating the correct interfaces. The CLI command is implemented under the existing [controller optics](#) command.



Note Starting from Cisco IOS XR Release 25.2.1, the 2x200 implicit breakout is deprecated. To operate 2x200G modules in 2x200 mode, you must explicitly apply the breakout configuration.

Configuring 2x200G DACs with 2x200 Breakout

Before you begin

- Associated line cards should be operational.

- Supported 2x200G modules should be inserted.

Procedure

Step 1 Configure 2x200G DAC with 2x200G breakout.

Example:

This example shows how to configure 2x200G DAC with 2x200G breakout.

Example:

```
Router#configure
Router(config)#controller optics 0/0/1/1
Router(config-if)# breakout 2x200
Router(config-if)#commit
```

Step 2 Configure autonegotiation on the connected interfaces.

You must enable auto negotiation on the connected interfaces after applying the 2x200G breakout configuration for the DACs.

Example:

```
Router#configure
Router(config)#interface TwoHundredGigE 0/0/1/1/0
Router(config-if)# negotiation auto
Router(config-if)#commit
```

Configure 100G operating modes with 200G DAC

The configuration of 100G operating modes with 200G DAC is a process that

- allows devices to operate at various speeds and lane combinations, and

- provides high-bandwidth links between networking devices using 200G QSFP56 DAC and 4x100 DAC.

Table 71: Feature History Table

Feature Name	Release Information	Feature Description
Configure 100G operating modes with 200G DAC	Release 25.3.1	<p>Introduced in this release on: Fixed Systems (8010 [ASIC: A100], 8200 [ASIC: Q200]) (select variants only*)</p> <p>The feature addresses the need for flexible speed configuration, particularly for connecting to custom servers that support specific speed and lane modes, and to prevent alarms when optics with different speeds are inserted.</p> <p>You will have 100G operating modes with 200G QSFP56 DAC, that allows to configure multi-rate optics and passive copper cables to operate at various speeds and lane combinations.</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8201-32FH • 8101-32FH <p>The feature introduces these changes:</p> <p>CLI:</p> <p>The speed keyword is included along with the 100G [<i>host-lanes</i> < 4 / 2 >] option in the controller optics command.</p>

Flexible 100G modes configuration for 200G DAC

The configuration for 100G operating modes with 200G DAC feature allows you to manually configure the speed of the port as 100G when using 200G DAC modules. This feature provides a CLI command to explicitly set the speed configuration to 100G operating modes and optionally specify the number of host lanes. The CLI command is implemented under the existing **controller optics** command which allows users to configure the speed of a port and optionally specify the number of host lanes.

Benefits of the 100G modes in 200G DAC

- Allows the users to use the same 200G QSFP56 DAC for different speeds based on the speed support of peer device. This means a single cable type can serve multiple connectivity needs, reducing the variety of hardware required.
- The feature allows setting the unused port to a speed that suits the system, even if a different speed optics is inserted, thereby avoiding false alarms and streamlining operations.

Configure 100G operational modes with 200G and 4x100 DAC

Procedure

Step 1 Configure 100G operational modes with 200G and 4x100 DAC.

Example:

This example shows how to configure the speed of port as 100G with host lane valuse as 2. The supported host lanes for 100G speed are 2 and 4.

Example:

```
Router#configure
      Router(config)#controller optics 0/0/0/0
      Router(config-Optics)# speed 100g host-lanes 2
      Router(config-Optics)#commit
```

Step 2 Run the **show running-config controller optics** CLI command to verify the running configuration of the speed port.

Example:

```
Router#show running-config controller optics 0/0/0/0
Thu Aug 14 01:16:52.946 UTC
controller Optics0/0/0/0
  speed 100g host-lanes 2
```

Step 3 *Optional:* Run the **show configuration failed** CLI command to verify if the speed port configuration is failed.

Example:

This example shows the failure scenario, when the breakout is configured on the same port.

```
Router#show config failed
Tue Oct 29 13:07:55.478 UTC
!! SEMANTIC ERRORS: This configuration was rejected by
!! the system due to semantic errors. The individual
!! errors with each failed configuration command can be
!! found below.

Controller Optics0/0/0/0
  speed 100g host-lanes 2
!!% Breakout is configured on this port, please remove breakout configuration
  configuration
!
end
```

Example:

This example shows the failure scenario, when the unsupported host lanes are configured.

```

Router# :ios(config)#show config failed
Tue Oct 29 13:07:55.478 UTC
!! SEMANTIC ERRORS: This configuration was rejected by
!! the system due to semantic errors. The individual
!! errors with each failed configuration command can be
!! found below.

controller Optics0/0/0/0
  speed 100g host-lanes 3
!!% The list of supported host lanes for speed 100g is 2, 4
!
end

```

Note

After the CLI is verified, the alarm is raised when the optics is present and optics driver cannot configure the optics in such speed or host lanes:

```

Router# :Oct 29 12:25:42.808 UTC: optics_driver[274]: %PKT_INFRA-FM-3-
FAULT_MAJOR : ALARM_MAJOR : MODULE AND SPEED CONFIG MISMATCH :DECLARE
:0/RP0/CPU0: Optics0/0/0/18

```

If you remove the module, the alarm is cleared. Similarly, when a new module is inserted, the same alarm is triggered if the module does not support the configured speed.

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:

- [QDD-400G-ZR-S Transponder and Muxponder Configuration Values, on page 321](#)
- [QDD-400G-ZRP-S Transponder and Muxponder Configuration Values, on page 322](#)

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: 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
  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 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 and DP04QSDD-HE optical module.



Note QDD-400G-ZR-S supports 1x1 dac-rate in cFEC mode. QDD-400G-ZRP-S and DP04QSDD-HE supports 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
  WV-LOL = 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 = 59.843750000 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 and DP04QSDD-HE optical module. 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 and DP04QSDD-HE supports 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      FLEXP_GIDM = 0
FLEXP-MM = 0      FLEXP-LOM = 0  FLEXP-RDI = 0
FLEXP-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 media loopback and host loopback on optics controllers. Loopback can be performed only in the maintenance mode.



Note Line loopback mode is supported only on Cisco 8000 series line cards and fixed-port routers based on Q100 and Q200 silicon.

Loopback Configuration Example

This example shows how to enable loopback configuration on optics controllers.

Use `show controllers optics R/S/I/P information loopback` command to check the supported loopback types.

```

Router#show controllers optics 0/0/0/4 information loopback
Supported Loopback Types :
=====

```

```

[1.] Media Line
[2.] Host Internal

```

```

Unsupported Loopback Types :
=====

```

```

[1.] Media Internal
[2.] Host Line
[3.] Host Per Lane
[4.] Media Per Lane
[5.] Simultaneous Media Host
Media Configured Loopback : Media Loopback None
Media Applied Loopback    : Media Loopback None

```

```

Host Configured Loopback : Host Loopback None
Host Applied Loopback    : Host Loopback None

```

Use `loopback` and `host loopback` commands in `config-optics` sub mode to configure the media and host loopback modes respectively. Loopback mode for both media and host can be configured to either internal or line, depending on the supported loopback types.

```

Router#config
Router(config)#controller optics 0/0/0/4
Router(config-Optics)#sec-admin-state maintenance
Router(config-Optics)#loopback line /* configures the media loopback to line */

```

```
Router(config-Optics)#host loopback internal /* configures the host loopback to internal
*/
Router(config-Optics)#commit
```

Running Configuration

This example shows the running configuration on optics controllers.

```
Router#show run controller optics 0/0/0/4
Thu May 13 19:51:08.175 UTC
controller Optics0/0/0/4
  loopback line
  host loopback internal
  sec-admin-state maintenance
!
```

Verification

This example shows how to verify the loopback configuration on optics controllers.

```
Router#show controllers optics 0/0/0/4
Controller State: Up
Transport Admin State: In Service
Laser State: On
Host Squelch Status: Enable
Media linkdown preFEC degrade : Disabled
LED State: Yellow
FEC State: FEC ENABLED
Power Mode: High
Dom Data Status: Ready
Last link flapped: 00:02:32
Optics Status
  Optics Type: QSPDD 400G ZR
  DWDM carrier Info: C BAND, MSA ITU Channel=1, Frequency=196.10THz,
  Wavelength=1528.773nm
  Loopback Host : Internal
  Loopback Media : Line

  Alarm Status:
  -----
  Detected Alarms: None

  LOS/LOL/Fault Status:
  ...

Router#show controllers optics 0/0/0/4 information loopback
Supported Loopback Types :
=====
[1.] Media Line
[2.] Host Internal

Unsupported Loopback Types :
=====

[1.] Media Internal
[2.] Host Line
[3.] Host Per Lane
[4.] Media Per Lane
[5.] Simultaneous Media Host
Media Configured Loopback : Media Loopback Line
Media Applied Loopback   : Media Loopback Line

Host Configured Loopback : Host Loopback Internal
Host Applied Loopback   : Host Loopback Internal
```

Disable Auto-Squelching

Table 72: Feature History Table

Feature Name	Release Information	Description
Disable Auto-Squelching	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8712-MOD-M • 8011-4G24Y4H-I
Disable Auto-Squelching	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-12TH24FH-E • 88-LC1-36EH+A8:B12 • 88-LC1-52Y8H-EM

Feature Name	Release Information	Description
Disable Auto-Squelching	Release 7.11.1	<p>This release introduces support to disable Auto squelching. This helps to detect weak signals that are hidden within the laser source noise. By disabling Auto squelch, you can reduce the processing overhead in systems that have stable laser sources and minimal noise, helping you optimize the performance of your system. When the Auto squelch function is enabled, the optical module will generate a local fault signal on the host side if it detects a fault on the media side. By default, Auto squelch is enabled.</p> <p>The feature introduces these changes:</p> <p>CLI:The following keyword has been added.</p> <ul style="list-style-type: none"> • host auto-squelch disable <p>YANG DATA models:</p> <ul style="list-style-type: none"> • New XPath for <code>Cisco-IOS-XR-controller-optics-cfg</code> (see Github, YANG Data Models Navigator)

This release introduces the support to disable auto-squelch functionality on the module on the host side. When enabled, the squelch function is activated on the module when no suitable media-side input signal from the remote end is available to be forwarded to the host-side output (example: Rx LOS is asserted). Auto squelching is commonly used to suppress unwanted noise from laser sources in communication systems. When disabled and no valid signal is detected on the module from the remote end, the module will generate a local fault towards the NPU. However, disabling auto-squelching provides you with expanded signal detection. This enables you to detect extremely weak signals that are embedded within the laser source noise. Also, by eliminating the need to continuously monitor and suppress unwanted noise, system resources can be allocated more efficiently, leading to improved performance.

In this feature, we introduced the **host auto-squelch disable** command to disable the auto-squelch functionality when there is an invalid input signal from the remote end. This feature provides you with the flexibility to customize the system's behavior according to your requirements.

Disabling Laser Squelching Configuration Example

This example shows how to disable laser squelching for a host on controller optics:

```
router#config
router(config)#controller 0/0/0/0
```

```
router(config-Optics)#host auto-squelch disable
router(config-Optics)#commit
```

Verification

This example shows how to verify the laser squelching disabled configuration:

```
router#show controllers optics 0/0/0/0
Host Squelch Status: disable
```

Configuring Performance Monitoring

Feature Name	Release Information	Description
New Parameters for Performance Monitoring	Release 24.2.11	<p>In this release, we've added two new parameters for performance monitoring. These parameters allow you to check the quality of electrical signals between an ASIC or NPU and the 400G Digital Coherent QSFP-DD optical modules periodically. This helps to detect errors occurring during data transmission and link initialization.</p> <p>This feature introduces new parameters Host-Intf-{n}-FEC-BER and Host-Intf-{n}-FEC-FERC in the show controllers coherentdsp command.</p>

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.

Release 24.2.11 introduces the following host PM parameters:

- **Host-Intf-{n}-FEC-BER**
- **Host-Intf-{n}-FEC-FERC**

The following table lists the details of the host PM parameters:

Table 73: Performance Monitoring Parameters

PM Parameter	Mode Type	Number of Host Interfaces	Description
Host-Intf-{n}-FEC-BER	Transponder	1	$n=0$ For example, Host-Intf-0-FEC-BER
	Muxponder	4	$n = 0, 1, 2, \text{ and } 3.$ For example: <ul style="list-style-type: none"> • Host-Intf-0-FEC-BER • Host-Intf-1-FEC-BER • Host-Intf-2-FEC-BER • Host-Intf-3-FEC-BER
Host-Intf-{n}-FEC-FERC	Transponder	1	$n=0$ For example, Host-Intf-0-FEC-FERC
	Muxponder	4	$n = 0, 1, 2, \text{ and } 3$ For example: <ul style="list-style-type: none"> • Host-Intf-0-FEC-FERC • Host-Intf-1-FEC-FERC • Host-Intf-2-FEC-FERC • Host-Intf-3-FEC-FERC

These parameters check the quality of electrical signals between an ASIC or NPU and the 400G Digital Coherent QSFP-DD optical modules. The Cisco IOS XR software installed on the router collects Bit Error Rate (BER) and Frame Error Counters (FERC) information from the optical modules and maintains history for 30-sec, 15-minute, and 24-hours intervals.



Note You can check these signals for host interfaces in both transponder and muxponder mode.

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 74: 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.
SOPMD	Sets the SOPMD (second order polarization mode dispersion) threshold or TCA reporting status.

Table 75: 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.

PM Parameters	Description
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.
Host-Intf-0-FEC-BER	Sets the Host-Intf-0-FEC-BER threshold or TCA reporting status, where: <ul style="list-style-type: none"> • AVG - specifies the number of corrected bits received from the host interface prior to a PM interval. • MIN - specifies the minimum number of corrected bits received from the host interface over a sub-interval and prior to a PM interval. • MAX - specifies the maximum number of corrected bits received from the host interface over a sub-interval and prior to a PM interval.
Host-Intf-0-FEC-FERC	Sets the Host-Intf-0-FEC-FERC threshold or TCA reporting status, where: <ul style="list-style-type: none"> • AVG - specifies the number of frames received from the host interface during a sub-interval. • MIN - specifies the minimum number of frames received from the host interface with uncorrected errors over a sub-interval and prior to a PM interval. • MAX - specifies the maximum number of frames received from the host interface with uncorrected errors over a sub-interval and prior to a PM interval.

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 Configured	AVG TCA	MAX	Operational	Configured Threshold (min)	TCA (min)	Operational Threshold (max)
LBC[mA]	: 0.0 NA	0.0 NO	0.0	0.0	NA	NO	100.0
OPT[dBm]	: -9.98 NA	-9.98 NO	-9.98	-15.09	NA	NO	0.00
OPR[dBm]	: -40.00 NA	-40.00 NO	-40.00	-30.00	NA	NO	8.00
CD[ps/nm]	: 0 100	0 NO	0	-80000	-100	NO	100
DGD[ps]	: 0.00 NA	0.00 NO	0.00	0.00	NA	NO	80.00
SOPMD[ps^2]	: 0.00 NA	0.00 NO	0.00	0.00	NA	NO	2000.00
OSNR[dB]	: 0.00 NA	0.00 NO	0.00	0.00	NA	NO	40.00
PDL[dB]	: 0.00 NA	0.00 NO	0.00	0.00	NA	NO	7.00
PCR[rad/s]	: 0.00 NA	0.00 NO	0.00	0.00	NA	NO	2500000.00
RX_SIG[dBm]	: -40.00 NA	-40.00 NO	-40.00	-30.00	NA	NO	1.00
FREQ_OFF[Mhz]	: 0 NA	0 NO	0	-3600	NA	NO	3600
SNR[dB]	: 0.00 NA	0.00 NO	0.00	7.00	NA	NO	100.00

```
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)                           (min)    (min)    (max)    (min)      (enable)
PreFEC BER   : 0E-15   0E-15   0E-15   0E-15   NO
0E-15        NO
PostFEC BER  : 0E-15   0E-15   0E-15   0E-15   NO
0E-15        NO
Q[dB]        : 0.00    0.00    0.00    9.00 YES 120.00 YES
Q_Margin[dB] : 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 76: 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
    
```

Configuring FEC Alarm Threshold

Table 77: Feature History Table

Feature Name	Release Information	Description
Configurable FDD and FED Alarm Threshold Values	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8712-MOD-M • 8011-4G24Y4H-I

Feature Name	Release Information	Description
Configurable FDD and FED Alarm Threshold Values	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-12TH24FH-E • 88-LC1-36EH+A8:B12 • 88-LC1-52Y8H-EM

Feature Name	Release Information	Description
Configurable FDD and FED Alarm Threshold Values	Release 24.1.1	

Feature Name	Release Information	Description
		<p>We now ensure that you have accurate data to initiate proactive maintenance for non-critical FEC errors or take prompt action to prevent potential optical link data loss in your network. This is made possible because we've enabled the configuration of FEC (Forward Error Correction) Detected Degrade (FDD) alarm threshold values for non-critical FEC errors and FEC Excessive Degrade (FED) alarm threshold values for critical FEC errors. You can configure or clear these values for QDD-400G-ZR, QDD-400G-ZRP, and DP04QSDD-HE0 optical modules.</p> <p>Prior to this release, the router would automatically generate FEC alarms based on default threshold values.</p> <p>The feature introduces these changes:</p> <p>CLI:</p> <p>Modified the controller optics command by adding the following keywords:</p> <ul style="list-style-type: none"> • host fec-threshold excess-degrade raise • media fec-threshold excess-degrade raise • host fec-threshold excess-degrade clear • media fec-threshold excess-degrade clear • host fec-threshold detected-degrade raise • media fec-threshold detected-degrade raise • host fec-threshold detected-degrade clear • media fec-threshold detected-degrade clear

Feature Name	Release Information	Description
		<p>The fec-thresholds keyword is added to the show controller optics command.</p> <p>YANG Data Model:</p> <ul style="list-style-type: none"> • New XPaths for <ul style="list-style-type: none"> Cisco-IOS-XR-controller-optics-oper.yang Cisco-IOS-XR-controller-optics-fec-threshold-cfg.yang

Forward Error Correction (FEC) is used to control errors during data transmission. FEC works by adding data redundancy to the transmitted message. This redundancy allows the receiver to detect and correct a limited number of errors occurring anywhere in the message, instead of the transmitter resending the entire message. For additional information on FEC, see [Understanding FEC and Its Implementation](#).

There are two types of FEC alarms:

- **FEC Detected Degrade (FDD) alarm:** The FDD alarm is raised when the link degradation is within the permissible limit and does not cause traffic disruption. This alarm indicates the system is working harder than usual to maintain data transmission. Link degradation could be due to issues in the cable, network congestion, or other hardware failure.
- **FEC Excessive Degrade (FED) alarm:** The FED alarm is raised when the link degradation exceeds beyond the permissible limit and causes traffic disruption. This alarm indicates the system is working harder than usual to maintain data transmission. Without corrective measures, network performance deteriorates further and eventually results in traffic loss. Link degradation could be due to issues in the cable, network congestion, or other hardware failure.

The FEC alarms threshold values can now be configured to control alarms (raise and clear FEC alarms) on both media and host side of the optical transceiver. The optical transceiver is divided into two sides, the host side, which is positioned towards the router, and the media side, which is positioned towards the wire or cable media.

When the average bit error rate (BER) exceeds the **raise threshold value**, the FEC alarm is raised (or asserted). Similarly, when the BER drops below the **clear threshold value**, then the alarm is cleared (or de-asserted).

Guidelines and Restrictions for Setting the FEC Alarm Thresholds

- The **raise threshold value** must always be greater than the **clear threshold value** for both FDD and FED alarms.
- The **raise** or **clear threshold value** of FED alarm must always be greater than the **raise** or **clear threshold value** of the FDD alarm.
- While the router configuration permits a range of 1 to 18446744073709551615, the router only supports a range of 1 to 204600000000000000. The threshold value provided by users is converted from a 64 bit number to a 16 bit number. As a result, there is minor variation between the user provided value (configured value) and the programmed value. The user input (threshold value) is appended with exponents relative to E-18.

Table 78:

Configured Value	Programmed Value (Displayed using the Show CLI command)	Pattern	
1, 2, 3,,10	0, 1, 2,.....,9	1<ConfiguredValue< 10, show command value = ConfiguredValue - 1	1->>>0.9999, displayed as 0 and so on
11,12,13,.....,99	1.0, 1.1, 1.2,...9.8	10<ConfiguredValue<99, show command value = ConfiguredValue - 0.1	
111,222,333...999	1.10, 2.21, 3.32	100<ConfiguredValue<999, show command value = ConfiguredValue - 0.01	
1111,1112,1113 upto 2047	1.110, 1.111, 1.112...	1000<ConfiguredValue< 2047 show command value = ConfiguredValue - 0.001	
2050, 12345, 23456,65432,...	2.0500, 1.2300, 2.3400,6.5400...	2047<ConfiguredValue<maximum- range show command value = first 3 digits appended by 0s	

Configuration Examples to Set FEC Alarm Threshold

Examples to configure FEC alarm threshold:

- [Configuring FED Alarm Thresholds, on page 359](#)
- [Configuring FDD Alarm Thresholds, on page 358](#)

Configuring FDD Alarm Thresholds

FDD Configuration Example

This example shows how to set FDD clear and raise alarm thresholds on the optics controller:

```
Router#config
Router(config)#controller optics 0/0/0/10
Router(config-Optics)#host fec-threshold detected-degrade clear 12000
Router(config-Optics)#host fec-threshold detected-degrade raise 22000
Router(config-Optics)#commit
Router(config-Optics)#end
```

Running Configuration

This example shows the running configuration on the optics controller:

```
Router#show running-config controller optics 0/0/0/10
Sat Feb  3 06:01:56.354 UTC
controller Optics0/0/0/10
host fec-threshold detected-degrade raise 22000
host fec-threshold detected-degrade clear 12000
!
!
```

Verification

This example shows how to verify the alarm threshold values on optics controllers:

```
Router#show controller optics 0/0/0/10 fec-thresholds
FEC Threshold Information
```

	Raise	Clear
Media FEC excess degrade	1.2600E-02	1.2100E-02
Media FEC detected degrade	1.1700E-02	1.1300E-02
Host FEC excess degrade	2.4000E-02	2.4000E-03
Host FEC detected degrade	2.2000E-14	1.1989E-14

Configuring FED Alarm Thresholds

FED Configuration Example

This example shows how to set FED raise and clear alarm thresholds on the optics controller:

```
Router#config
Router(config)#controller optics 0/0/0/12
Router(config-Optics)#host fec-threshold excess-degrade clear 14000
Router(config-Optics)#host fec-threshold excess-degrade raise 24000
Router(config-Optics)#commit
Router(config-Optics)#end
```

Running Configuration

This example shows the running configuration on the optics controller:

```
Router#show running-config controller optics 0/0/0/12
Sat Feb  3 06:02:00.153 UTC
controller Optics0/0/0/12
host fec-threshold excess-degrade raise 24000
host fec-threshold excess-degrade clear 14000
!
```

Verification

This example shows how to verify the alarm threshold values on optics controllers:

```
Router#show controller optics 0/0/0/12 fec-thresholds
FEC Threshold Information
```

	Raise	Clear
Media FEC excess degrade	1.2600E-02	1.2100E-02
Media FEC detected degrade	1.1700E-02	1.1300E-02
Host FEC excess degrade	2.3900E-14	1.3999E-14
Host FEC detected degrade	9.0000E-03	9.0000E-04

Enhanced SOP tolerance mode for ULH optics

An enhanced SOP tolerance mode is an optical module feature that

- increases resilience to rapid state of polarization (SOP) fluctuations,

- enables stable ultra long-haul (ULH) coherent signal transmission over challenging fiber environments, and
- actively compensates for polarization impairments using advanced DSP techniques.

Table 79: Feature History Table

Feature Name	Release info	Description
Enhanced SOP support for ULH optics	Release 25.2.1	<p>Introduced in this release on: Modular Systems (8800 [LC ASIC: Q100, Q200, P100])(select variants only*)</p> <p>This feature helps to improve the coherent optics' ability to handle fluctuations in the light's State of Polarization (SOP). It achieves this by enhancing the Digital Signal Processing (DSP) to track and compensate for rapid SOP rotation. The capability ultimately ensures more reliable data transmission, maintains high signal quality, and increases network stability, especially in long-distance and dynamic optical deployments.</p> <p>This feature is applicable on:</p> <ul style="list-style-type: none"> • 88-LC0-36FH • 88-LC0-36FH-M <p>CLI:</p> <p>The enh-sop-tol-mode keyword is added to the controller optics command.</p>

Benefits of SOP tolerance mode for ULH optics

Enhanced SOP tolerance mode provides several benefits.

- Improved performance: Digitally compensates for dynamic SOP changes to maintain high signal quality and network stability in challenging fiber conditions.
- Increased robustness: Offers greater resilience to polarization-related impairments, especially crucial for long-haul and subsea optical deployments.

Supported hardware variants

Enhanced SOP tolerance mode is supported on these line cards:

- 88-LC0-36FH
- 88-LC0-36FH-M

Configure enhanced SOP tolerance mode

Enable enhanced SOP tolerance mode on a coherent optics controller to improve performance under polarization fluctuations. By default, the enhanced SOP tolerance mode is disabled.

Use this task when optical signal quality is affected by polarization changes and enhanced tolerance is required.

Procedure

Step 1 Enable enhanced SOP tolerance mode.

Example:

```
Router# config
Router(config)# controller optics 0/0/0/14
Router(config-Optics)# enh-sop-tol-mode 1
Router(config-Optics)# commit
```

The possible values for enhanced SOP tolerance mode are *1* or *no enh-sop-tol-mode*.

Step 2 Verify that enhanced SOP tolerance mode is enabled.

Example:

```
Router# show run 0/0/0/14
!! Building configuration...
!! IOS XR Configuration 25.2.2.09I
!! Last configuration change at Mon Jul 28 18:56:00 2025 by cisco
!
hostname abc
username cisco
  group root-lr
  group cisco-support
  secret 10
$6$g8TGE1mb2o7/BE1.$X1wz3z0fTKp6E0utoKvS6H4CFT0B0mM8PGt6M5MndzrjnM/gJk.NwgOf5n3SZm2.qaPeRrZar7VQTtNnT16b1
!
line console
  exec-timeout 0 0
!
interface MgmtEth0/RP0/CPU0/0
  ipv4 address dhcp
!
controller Optics0/0/0/14
  enh-sop-tol-mode 1
!
interface FourHundredGigE0/0/0/0
  description AOC
  shutdown
```

Media Link-down PreFEC Degrade Enablement

Table 80: Feature History Table

Feature Name	Release Information	Description
Media Link-down PreFEC Degrade Enablement	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8712-MOD-M • 8011-4G24Y4H-I
Media Link-down PreFEC Degrade Enablement	Release 24.3.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: Q200, P100], 8700 [ASIC: P100]); Centralized Systems (8600 [ASIC: Q200]); Modular Systems (8800 [LC ASIC: Q100, Q200, P100])</p> <p>The Media Link-down PreFEC Degrade functionality can be used to protect the media side of the optical transceiver during transmission errors.</p> <p>By using this feature, you can proactively switch the traffic to standby path when the BER counter crosses the threshold value. This feature helps to avoid further traffic impact when the optical network reaches more noise or error.</p> <p>The feature introduces these changes:</p> <p>CLI:</p> <p>Modified the <code>controller optics</code> command by adding the <code>media link-down prefec-degrade</code> keyword.</p> <p>YANG Data Model:</p> <ul style="list-style-type: none"> • New XPaths for <code>Cisco-IOS-XR-controller-optics-oper.yang</code> • New XPaths for <code>Cisco-IOS-XR-um-cont-optics-fec-threshold-cfg.yang</code> <p>(see GitHub, YANG Data Models Navigator)</p>

The Media Link-down PreFEC Degrade functionality can be used to protect the media side of the optical transceiver during transmission errors, such as errors due to noise, or data transmission errors. This feature is disabled by default. You can enable this feature by using the `media link-down prefec-degrade` command.

Prerequisites for using Media Link-down PreFEC Degrade Functionality

To use the Media Link-down PreFEC Degrade functionality, you must configure the FEC Alarm Threshold. For information on configuring FEC alarms threshold, see [Configuring FEC Alarm Threshold](#).

About Media Link-down PreFEC Degrade Functionality

Prior to this release, the FEC Alarm Threshold functionality enabled you to configure the FEC alarms threshold values to control alarms (raise and clear FEC alarms) on media and host side of the optical transceiver. Using the FEC Alarm Threshold functionality, you can configure the FDD and FED alarm threshold values and set the **raise threshold value** and **clear threshold value** values to control alarms.

After you configure FEC Alarm Threshold and enable Media Link-down PreFEC Degrade functionality, you get the alarm notification when the average bit error rate (BER) exceeds the threshold value. This triggers link-down and enables switchover functionality automatically. The traffic is switched to standby path, and remains in the standby path until the alarm is cleared or based on the settings done by the network operator.



Note In Cisco IOS XR Release 24.3.1, the Link-down PreFEC Degrade feature is supported only on the media side of the optical transceiver.

Configure Media Link-down PreFEC Degrade

The purpose of this task is to enable the media link-down preFEC degrade functionality to proactively switch the traffic to standby path.

Procedure

Step 1 Execute the **media link-down prefec-degrade** command to configure link-down preFEC degrade on the media side of the optics controller.

Example:

```
Router#config
Router(config)#controller optics 0/2/0/16
Router(config-Optics)#media link-down prefec-degrade
Router(config-Optics)#commit
```

Step 2 Execute the **show running-config controller optics R/S/I/P** command to view the running configuration on the optics controller.

Example:

```
Router#show running-config controller optics 0/2/0/16
Thu May 13 20:18:55.957 UTC
controller Optics0/2/0/16
  media link-down prefec-degrade
!
```

Step 3 Execute the **show controller optics R/S/I/P** command to verify link-down preFEC degrade feature on optics controllers.

Example:

```
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: On
Media linkdown prefec degrade : Enabled
LED State: Green
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 ps²

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

Application select code provisioning

Application select code, also known as AppSel code, is a feature that:

- allows the host device to choose the operating mode of a QDD module
- allows you to configure the media code, and
- specifies how to configure the optical side of a module.

Table 81: Feature History Table

Feature Name	Release Information	Feature Description
Application select code provisioning	Release 25.2.1	<p>You can now configure application select codes directly on a QDD module by using a CLI. This simplifies provisioning by allowing the selection of advertised application modes such as 400ZR, OpenZR+ and others. The router activates the selected code to ensure compatibility and reduce configuration complexity.</p> <p>This feature introduces these changes:</p> <p>CLI:</p> <ul style="list-style-type: none"> • The appsel simple code keyword is introduced in the Controller optics command. • The appsel keyword is introduced in the Show controller optics command.

The Common Management Interface Specification (CMIS) specifications set rules for how QDD modules work and how a host device sets them up. CMIS provides a consistent way for host devices, such as routers or switches, to communicate with and control optical modules, regardless of the module manufacturer.

Each QDD module can work in different modes, and these modes are identified by a special code called an AppSel code. This code acts like a unique ID for each mode. Each mode has an application descriptor, which explains how the module handles data. It describes how signals are processed between the connections on the host side and the optical side. The optical side is also known as the media side of the module. The AppSel code also includes a media code, which tells the module how to set up its optical side. The host software uses

this media code to control the module's optical interface. The host software also sets up other components, like the physical layer (PHY), SerDes, and MacPort, to complete the data path.

Modules store a list of the AppSel codes they support in their EEPROM memory. This helps the host system or device know which settings the module can use. Some codes, like 400G-OIF-ZR and 400G-OpenZR+, are standard and follow industry rules. Other codes are custom, made by third-party vendors to give users more options with one module. However, these custom codes can cause problems. For example, Cisco routers may not work well with modules using custom codes if the host system doesn't recognize them.

This feature allows users to select application modes advertised by the optical module. You can choose all application modes a module supports.

AppSel supports operation modes based on these parameters:

- Data rate, such as 100GbE or 400GbE
- Signal type
- Signal processing between host side and media side
- Optical configuration, based on the media code, which is part of the AppSel code.

Benefits of AppSel code provisioning

These are the benefits of provisioning AppSel code:

- Enables you to choose any application mode supported by the optical module for greater flexibility.
- Enables direct configuration of the NPU, PHY, and optics to match the selected application's datapath.
- Removes the need for Cisco IOS XR software to implement new proprietary modes for each vendor, simplifying software requirements.
- Allows you to directly specify the desired AppSel code, enabling straightforward configuration without requiring Cisco IOS XR software to interpret the mode.
- Eliminates the extra step of mapping custom vendor codes, reducing delays in supporting new module vendors.

Configuration guidelines and restrictions for AppSel code provisioning

Configuration guidelines for AppSel code provisioning

These configurations apply for AppSel code provisioning:

- This feature is not enabled by default. You must configure it to take effect.
- If you configure a non-default application ID and later remove the configuration, the selected application ID falls back to the default.
- If you configure a non-default application ID and later overwrite it with an invalid application ID, the selected application ID falls back to default. The system raises an alarm indicating the invalid configuration.
- The maximum number of applications supported depends upon the module as advertised in the AppSel list.

Restrictions for AppSel code provisioning

These restrictions apply to AppSel code provisioning:

- The Cisco 8000 series routers do not support In Service Software Upgrade (ISSU) or In Service Software Downgrade (ISSD).
- Optical Transport Network (OTN) is not supported.

How AppSel code provisioning works

The key components that are involved in the AppSel code provisioning are:

- **Optical side:** The optical side or the media side refers to the part of a pluggable module, such as a QDD or other optical transceiver that connects to and communicates with the optical network. It is the interface responsible for converting electrical signals, from the host device, into optical signals for transmission over fiber optic cables, and vice versa for receiving signals.
- **Host side:** The host side refers to the part of a pluggable module, such as a QDD or other optical transceiver that interfaces with the host device. A host device is typically a network device like a router, switch, or server. It is responsible for handling and processing electrical signals exchanged between the module and the host device.
- **Optics Driver:** The entity that collects all supported application codes from the module. Users can see all supported application codes and select any one of the supported application codes. The optics driver programs the valid application code to the module.
- **Application Codes:** The unique codes that represent operational modes of the optical module.
- **Optics Management Agent (MA):** The entity that sends the default configuration to the optics driver if no user configuration is provided. It also manages fallback behavior when no application code is selected.

Workflow

These stages describe how AppSel code provisioning process works.

1. AppSel code identifies the optical module modes when the modules are plugged into a router.
2. AppSel code points to application descriptor. In this stage, the AppSel code acts as a sequence number for an application descriptor.
3. The application descriptor defines the configuration. In this stage, the application descriptor describes a functional transmission configuration, including signal processing between host lanes and media lanes.
4. The AppSel code that contains the media code configure the media side of the optical module.
5. The router software or the host software applies the media code to configure the optical interface of the optical module.
6. Based on the module's host side interface, you can configure PHY, NPU SerDes, and MacPort.
7. The host side and media side configurations complete, establishing the datapath.

Result

The router enables AppSel code provisioning as the datapath is complete between the host side and media side. This ensures that the optical module operates correctly and efficiently in the desired mode, with proper coordination between the host side interface and the media side.

Configure an AppSel code on an optical module

Configure the AppSel code to enable the optical module to operate in a specific application mode, such as 400ZR or OpenZR+.

AppSel codes are advertised by the module and must be validated before configuration. This ensures compatibility between the host and the module.

Before you begin

Ensure the router is running Cisco IOS XR Software Release 25.2.1 or later.

Procedure

Step 1 Identify the AppSel code that needs to be configured on a particular port from the list of available appcodes.

Example:

```
Router# show controllers optics 0/0/0/0 appsel advertised
Router# show controllers optics 0/0/0/0 appsel detailed
```

Step 2 Enter configuration mode on the router.

Example:

```
Router# conf
```

Step 3 (Optional) Identify the controller optics interface and configure breakout to match with the AppSel code that you want to configure.

Example:

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

Step 4 Configure the AppSel code.

Example:

```
Router(config)# controller optics 0/0/0/0
appsel simple code 4
!
```

Step 5 Verify the configuration.

Example:

```
Router# #show controllers optics 0/0/0/0 appsel advertised
```

```
Sun Feb  2 20:00:04.884 UTC
```

```
-----
App-ID | Host-ID | Media-ID | Standard |
Host   | Power  |          |          |
```

Supported	Consumption(W)						
1 Yes	17 n/a	ETH 400GAUI-8 C2M (Annex	62	OIF 400ZR, DWDM, amplifi	OIF		
2 Yes	13 n/a	ETH 100GAUI-2 C2M (Annex	62	OIF 400ZR, DWDM, amplifi	OIF		
3 Yes	17 n/a	ETH 400GAUI-8 C2M (Annex	70	OpenZR+ ZR400-OFEC-16QAM	OpenZR+		
4 Yes	13 n/a	ETH 100GAUI-2 C2M (Annex	70	OpenZR+ ZR400-OFEC-16QAM	OpenZR+		
5 Yes	17 n/a	ETH 400GAUI-8 C2M (Annex	199	0xC0-0xFE Vendor Specif	0xC0-0xFE		
6 Yes	15 n/a	ETH 200GAUI-4 C2M (Annex	199	0xC0-0xFE Vendor Specif	0xC0-0xFE		
7 Yes	13 n/a	ETH 100GAUI-2 C2M (Annex	199	0xC0-0xFE Vendor Specif	0xC0-0xFE		
8 Yes	17 n/a	ETH 400GAUI-8 C2M (Annex	196	0xC0-0xFE Vendor Specif	0xC0-0xFE		
9 Yes	15 n/a	ETH 200GAUI-4 C2M (Annex	196	0xC0-0xFE Vendor Specif	0xC0-0xFE		
10 Yes	13 n/a	ETH 100GAUI-2 C2M (Annex	196	0xC0-0xFE Vendor Specif	0xC0-0xFE		
11 Yes	17 n/a	ETH 400GAUI-8 C2M (Annex	200	0xC0-0xFE Vendor Specif	0xC0-0xFE		
12 Yes	15 n/a	ETH 200GAUI-4 C2M (Annex	200	0xC0-0xFE Vendor Specif	0xC0-0xFE		
13 Yes	13 n/a	ETH 100GAUI-2 C2M (Annex	200	0xC0-0xFE Vendor Specif	0xC0-0xFE		
14 Yes	17 n/a	ETH 400GAUI-8 C2M (Annex	83	OTN-ITU-T FOIC4.8-DO (G.	OTN-ITU-T		
15 Yes	254 n/a	0xC0-0xFE Vendor Specif	254	0xC0-0xFE Vendor Specif	0xC0-0xFE		

Router# show controllers optics 0/0/0/0 appsel detailed

Sun Feb 2 20:00:29.702 UTC

App-ID	Host-ID	Media-ID	Host Lane	Media Lane	Host Lane	Media Lane
Host			Count	Count	Assign	Assign
Supported						
1 Yes	17	62	8	1	1	1
2 Yes	13	62	2	1	85	1
3 Yes	17	70	8	1	1	1
4 Yes	13	70	2	1	85	1
5 Yes	17	199	8	1	1	1
6 Yes	15	199	4	1	17	1
7 Yes	13	199	2	1	85	1

Yes														
8		17		196		8		1		1		1		
Yes														
9		15		196		4		1		17		1		
Yes														
10		13		196		2		1		85		1		
Yes														
11		17		200		8		1		1		1		
Yes														
12		15		200		4		1		17		1		
Yes														
13		13		200		2		1		85		1		
Yes														
14		17		83		8		1		1		1		
Yes														
15		254		254		8		1		255		1		
Yes														

```
Router# show controllers optics 0/0/0/0 appsel active
```

```
Sun Feb 2 20:00:47.776 UTC
Instance      : 1
App-ID       : 3
Host-ID      : 17   ETH 400GAUI-8 C2M (Annex
Media-ID     : 70   OpenZR+ ZR400-OFEC-16QAM
Host Lane Count : 8
Media Lane Count : 1
Host Lane Assign : 0x1
Media Lane Assign : 0x1
```

The optical module operates in the selected application mode, ensuring compatibility and optimal performance.

What to do next

- Monitor the interface status and confirm the active AppSel code.
- Ensure alarms are cleared, and the interface is operational.

Alarms Troubleshooting

Table 82: Feature History Table

Feature Name	Release	Description
Enhanced Alarm Prioritization, Monitoring, and Management	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8712-MOD-M • 8011-4G24Y4H-I
Enhanced Alarm Prioritization, Monitoring, and Management	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-12TH24FH-E • 88-LC1-36EH+A8:B12 • 88-LC1-52Y8H-EM
Enhanced Alarm Prioritization, Monitoring, and Management	Release 24.1.1	<p>To improve alarm prioritization, precise monitoring, and streamlined management, ensuring a more efficient and responsive approach to network events, we have introduced the suppression of LOL (Loss of Line) alarms when the LOS-P (Loss of Signal-Payload) alarm is generated, the clearing of alarm static counters, and the suppression of warnings when the respective alarm is triggered.</p>

In this latest release, we bring forth advanced features to elevate your alarm management experience. The key highlights include:

- **LOL Alarm Suppression:** Prioritize the detection and handling of critical LOS-P (Loss of Signal-Payload) alarms by suppressing LOL (Loss of Line) alarms when LOS-P alarms are generated.

- **Clearing Static Counters:** Gain the ability to clear alarm static counters using the command `clear counters controller coherentDSP location`, facilitating focused monitoring within a specific time frame.
- **Warning Suppression:** Enhance your monitoring environment by suppressing warnings when the corresponding alarm is triggered, effectively preventing redundant or repetitive alerts.

These enhancements collectively contribute to improved alarm prioritization, precise monitoring, and streamlined management, ensuring a more efficient and responsive approach to network events.

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

Procedure

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

Procedure

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

Procedure

- Step 1** 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.

Step 2 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

Procedure

- Step 1** 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.

Step 2 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

Procedure

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

Procedure

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

Procedure

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

Procedure

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

Procedure

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

Procedure

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

Procedure

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

Procedure

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

Procedure

Step 1 Identify and correct the underlying cause of mis-alignment. The Flexo LOF (Loss of Frame) alarm is cleared

Step 2 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

Loss of lock (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

Procedure

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

Procedure

Step 1 Identify and correct the underlying cause of mis-alignment. The Flexo LOM alarm is cleared when good

Step 2 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

Procedure

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

Procedure

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

Procedure

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

Procedure

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

Procedure

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

Procedure

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

Procedure

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

Procedure

Step 1 Verify the value of the minimum acceptable OSNR value of Cisco 8000 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 to 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

Procedure

- 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.
- Step 7** Verify if the fiber is of single-mode type.
- Step 8** Clean the fiber connectors at both ends for a signal degrade.
- 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

Procedure

- Step 1** 1 Verify the wavelength configuration.
- Step 2** 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 16

Configuring 800G Optics

The Cisco 800G QSFP-DD modules deliver high-capacity 800 Gigabit Ethernet connectivity for data center and enterprise networks, supporting enhanced scalability and flexibility.

- [Configure 2x400G and 8x100G breakout modes, on page 385](#)
- [Support for DP08QSDD-ZRB-19B optics on 88-LC1-36EH line card, on page 389](#)

Configure 2x400G and 8x100G breakout modes

The Cisco QSFP-DD800 modules offer advanced high-density 800 Gigabit Ethernet connectivity for modern data centers and enterprise networks, featuring:

- Increased density and flexibility to support web customer deployments.
- Utilization of QSFP-DD800 optics technology for efficient, high-density ethernet connections.
- Enhanced scalability for seamless data center interconnection and network growth.

Table 83: Feature History Table

Feature Name	Release Information	Feature Description
Configure 2x400G and 8x100G breakout modes	Release 25.3.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: Q200, P100],(select variants only*); ; Modular Systems (8800 [LC ASIC: Q100, Q200, P100])(select variants only*)</p> <p>You can now configure 2x400G and 8x100G breakout modes on the 8212-32FH-M routers and 88-LC1-36EH line cards using the QSFP-DD800 optical modules.</p> <p>This feature introduces these changes:</p> <p>CLI:</p> <ul style="list-style-type: none"> • mode 800 keyword in hw-module port-range command • breakout 8X100 keyword in controller optics command. <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8212-32FH-M • 88-LC1-36EH

Guidelines for using Cisco QSFP-DD800 optical modules

Use these guidelines to operate Cisco QSFP-DD800 optical modules:

- When operating 800G optical modules in supported port pairs, only the first port in each pair (e.g., Port 0, 1, 4, 8, 9, 12, 13, 16) is active, while the second port in the pair (e.g., Port 2, 3, 6, 10, 11, 14, 15, 18) remains inactive.

For example, in the first port pair (ports 0 and 2), port 0 is enabled, whereas port 2 is disabled.

- The major alarm `hw_optics: OPTICS Module is not supported on this port` does not appear when a Cisco 800G optical module is installed in a port configured with `mode 800`.

Supported port pairings on Cisco 8212-48FH-M Router

800G optics are supported on these port pairings of the Cisco 8212-48FH-M Router:

Port 1	Port 2
0	2
1	3
4	6
8	10
9	11
12	14
13	15
16	18
17	19
20	22
21	23
24	26
25	27
28	30
29	31
32	33
34	35
36	37
38	39
40	41
42	43
44	45
46	47

Configure 800G optics

To support 800G modules, configure the mode as 800 on the designated port pairings. This setup activates the primary port for 800G module insertion while disabling the secondary port in each pair.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Router# configure terminal
```

Step 2 Configure **mode 800** using the **hw-module port-range mode 800** command for the desired port pairing. Insert the module into the *first* port number of the pair.

Example:

```
Router(config)# hw-module port-range 0 0 instance 0 location 0/RP0/CPU0 mode 800
```

Step 3 Commit the configuration.

Example:

```
Router(config)# commit
```

Configure 8x100G breakout mode

You can configure the Cisco QDD-8X100G-FR optical module for 8x100G breakout mode to operate as eight separate 100 Gigabit Ethernet interfaces. The Cisco QDD-2X400G-FR4 optical module operates by default in 2x400G mode and does not require a breakout configuration.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Router# configure terminal
```

Step 2 Access the controller optics configuration for the port where the QDD-8X100G-FR optical module is inserted.

Example:

```
Router(config)# controller optics 0/0/0/13
```

Step 3 Apply the breakout 8x100.

Example:

```
Router(config-Optics)# breakout 8x100
Router# commit
```

Step 4 Verify the installed 800 GbE modules and their configurations using the **show inventory | include 8x100**.

Example:

```
Router#show inventory | i 8x100
NAME: "EightHundredGigE0/0/0/13", DESCR: "Cisco QSFPDD 8x100G FR Pluggable Optics Module"
NAME: "EightHundredGigE0/0/0/38", DESCR: "Cisco QSFPDD 8x100G FR Pluggable Optics Module"
```

Support for DP08QSDD-ZRB-19B optics on 88-LC1-36EH line card

A DP08QSDD-ZRB-19B optic is a QSFP-DD800 coherent pluggable module that

- delivers 800-Gbps ZR/ZR+ transmission over amplified C-band DWDM networks
- enables high-capacity long-haul and data center interconnect (DCI) optical links directly from the router port, and
- eliminates the need for external transponders for long-distance DWDM connections.

Starting with Cisco IOS XR Release 25.4.1, the DP08QSDD-ZRB-19B optical module is supported in 88-LC1-36EH line cards.

Table 84: Feature History Table

Feature Name	Release Information	Feature Description
Support for DP08QSDD-ZRB-19B optics on 88-LC1-36EH line card	Release 25.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: Q200, P100], 8700 [ASIC: P100, K100], 8010 [ASIC: A100]); Centralized Systems (8600 [ASIC: Q200]); Modular Systems (8800 [LC ASIC: Q100, Q200, P100])</p> <p>We have introduced support for the DP08QSDD-ZRB-19B 800G-ZR+ C-band coherent QSFP-DD module on the Cisco 88-LC1-36EH line card. This module enables 800G ZR/ZR+ coherent transmission over amplified C-band DWDM systems, providing high-capacity routed optical networking directly from the router port.</p>



CHAPTER 17

Configuring Controllers

This chapter describes the Optics Controller for the 36-port QSFP56-DD 400 GbE and 48-port QSFP28 100 GbE Line Cards. This chapter also describes the procedures used to configure the controllers.



Note When two MACsec enabled Cisco 8000 Series Routers with Coherent Line Cards are connected, there is no compatibility between Coherent Line Cards of IOS XR Release.

- breakout - Configure breakout mode ('breakout 4x10' only.)
- clear - Clear the uncommitted configuration.
- commit - Commit the configuration changes to running.
- do - Run an exec command.
- end - Exit from configure mode.
- exit - Exit from this submode.
- ext-description - Set ext-description for this controller.
- no - Negate a command or set its defaults.
- pwd - Commands used to reach current submode.
- root - Exit to the global configuration mode.
- show - Show contents of configuration.

Following controller configuration options are supported on the router:

- [How to Configure Controllers, on page 392](#)
- [Diagnostic Parameters for Optical Transceivers, on page 394](#)
- [View Coherent Optical Transceiver Module Parameters , on page 401](#)
- [Display of Alarms for Coherent Optical Transceivers, on page 405](#)
- [Restrictions, on page 406](#)
- [Verification, on page 406](#)
- [Pseudo Random Binary Sequence, on page 408](#)
- [Loopback on Optical Transceivers, on page 411](#)
- [Media Side Input Loopback Configuration, on page 413](#)

- [Media Side Output Loopback, on page 414](#)
- [Host Side Input Loopback Configuration, on page 415](#)
- [Host Side Output Loopback Configuration, on page 417](#)

How to Configure Controllers

This section contains the following procedures:

Configuring Optics Controller

Configuring optics controller of breakout 4x10:

```
RP/0/RP0/CPU0:uut#configure
Fri Oct 11 16:22:31.222 UTC
RP/0/RP0/CPU0:uut(config)#controller optics 0/1/0/28
RP/0/RP0/CPU0:uut(config-Optics)#breakout 4x10
RP/0/RP0/CPU0:uut(config-Optics)#commit
Fri Oct 11 16:23:26.868 UTC
RP/0/RP0/CPU0:uut(config-Optics)#end
RP/0/RP0/CPU0:uut#
RP/0/RP0/CPU0:uut#show running-config controller optics 0/1/0/28
Fri Oct 11 16:23:41.273 UTC
controller Optics0/1/0/28
breakout 4x10
!
```

Disabling Optical Modules

This feature provides the ability to disable and re-enable an optical module through CLI, which simulates online insertion and removal (OIR) by disabling power to the transceiver port.

Typical troubleshooting procedures for optical modules can include performing OIR by removing and re-installing the module, which requires onsite personnel to physically reseal the optical module. The ability to remotely disable and enable an optical module can significantly reduce operational expenses.

Example

The following output shows a QSFP28 module powered on and in UP state:

```
Router# show controllers optics 0/0/0/0

Controller State: Up

Transport Admin State: In Service

Laser State: Off

LED State: Not Applicable

FEC State: FEC ENABLED

Optics Status

    Optics Type:  QSFP28 100G FR
    Wavelength = 1311.00 nm

    Alarm Status:
```

```

-----
Detected Alarms: None

LOS/LOL/Fault Status:

Laser Bias Current = 26.2 mA
Actual TX Power = 0.73 dBm
RX Power = -0.68 dBm

Performance Monitoring: Disable

THRESHOLD VALUES
-----

Parameter                High Alarm  Low Alarm  High Warning  Low Warning
-----
Rx Power Threshold(dBm)   7.4        -10.4     4.5           -6.3
Tx Power Threshold(dBm)   7.0        -6.3     4.0           -2.4
LBC Threshold(mA)        100.00     8.00     83.00         10.00
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 = 27.92 Celsius
Voltage = 3.24 V

```

Transceiver Vendor Details

```

Form Factor           : QSFP28
Optics type           : QSFP28 100G FR
Name                  : CISCO-CISCO
OUI Number            : 00.00.0c
Part Number           : 10-3248-01
Rev Number            : 01
Serial Number         : FBN2331A114
PID                   : QSFP-100G-FR-S
VID                   : ES0
Date Code(yy/mm/dd)  : 19/09/19

```

To disable the module, use the **transceiver disable** command in controller optics configuration mode:

```

Router(config)# controller optics 0/0/0/0
Router(config-Optics)# transceiver disable
Router(config-Optics)# commit
Router(config-Optics)# end

```

The following example shows the QSFP28 module disabled and powered down:

```

Router# show controllers optics 0/0/0/0

Controller State: Down

Transport Admin State: In Service

Laser State: Off

Optics Status

Optics Type: Unknown optics
Wavelength = 0.00 nm

```

```

Alarm Status:
-----
Detected Alarms: None

LOS/LOL/Fault Status:

TX Power = N/A
RX Power = N/A

Performance Monitoring: Disable

THRESHOLD VALUES
-----

Parameter                High Alarm  Low Alarm  High Warning  Low Warning
-----
Rx Power Threshold(dBm)   7.4        -10.4     4.5          -6.3
Tx Power Threshold(dBm)   7.0        -6.3     4.0          -2.4
LBC Threshold(mA)        100.00     8.00     83.00       10.00
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 = 0.00 Celsius
Voltage = 0.00 V

```

Transceiver Vendor Details

To re-enable the module, use the **no transceiver disable** command in controller optics configuration mode.

Diagnostic Parameters for Optical Transceivers

Table 85: Feature History Table

Feature Name	Release Information	Description
Diagnostic Parameters for Optical Transceivers	Release 25.1.1	Introduced in this release on: Fixed Systems (8010 [ASIC: A100])(select variants only*) *This feature is supported on Cisco 8011-4G24Y4H-I routers.

Feature Name	Release Information	Description
Diagnostic Parameters for Optical Transceivers	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100, K100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none">• 8212-48FH-M• 8711-32FH-M• 8712-MOD-M• 88-LC1-12TH24FH-E• 88-LC1-36EH+A8:B12• 88-LC1-52Y8H-EM

Feature Name	Release Information	Description
Diagnostic Parameters for Optical Transceivers	Release 7.11.1	

Feature Name	Release Information	Description
		<p>You can analyze the diagnostic parameters for optical transceivers installed on a network device and detect potential issues with the optical transceivers, such as excessive power levels, abnormal temperature readings, or degradation of the optical signal. Such analysis is possible because the show controllers optics command now displays the following diagnostic parameters:</p> <ul style="list-style-type: none"> • Effective Signal to Noise Ratio (eSNR) • Pulse Amplitude Modulation with Four Levels (PAM4) Level Transition Parameter (LTP) • Pre-Forward Error Correction (FEC) and Post-FEC Bit Error Rate (BER) • Frame Error Count (FERC) • Laser age • Thermoelectric Cooler (TEC) current • Laser frequency • Laser temperature <p>For additional information on VDM (Versatile Diagnostics Monitoring), see the Common Management Interface Specification.</p> <p>The feature introduces these changes:</p> <p>CLI:</p> <ul style="list-style-type: none"> • The observable-info keyword is added to the show controller optics command. <p>YANG Data Model:</p> <ul style="list-style-type: none"> • New XPath for <code>Cisco-IOS-XR-controller-optics-oper.yang</code>

Feature Name	Release Information	Description
		(see GitHub , YANG Data Models Navigator)

In order to monitor and report the performance of an optical transceiver and thereby enhancing the troubleshooting capabilities of the optical transceiver, the **observable-info** keyword is added to the **show controllers optics** command to display the diagnostics parameters. These parameters help in monitoring the health of the network when the optical transceiver heats up, when the link is down, when alarms are raised, or when there's traffic loss in the network. This improvement in the **show controllers optics** command now displays the following diagnostic parameters:

- Effective Signal to Noise Ratio (eSNR)
- Pulse Amplitude Modulation with Four Levels (PAM4) Level Transition Parameter (LTP)
- Pre-Forward Error Correction (FEC) and Post-FEC Bit Error Rate (BER)
- Frame Error Count (FERC)
- Laser age
- Thermoelectric Cooler (TEC) current
- Laser frequency
- Laser temperature



Note Not all optical transceivers support the **observable-info** keyword. Also, the parameters that are displayed depend on what the optical transceiver supports, that is, not all optical transceivers display the same parameters. For additional information on VDM (Versatile Diagnostics Monitoring), see the [Common Management Interface Specification](#).

Verification

The following **show controllers optics observable-info** command displays the monitoring parameters of the optical transceiver present in the 0/0/0/9 location ID. The 0/0/0/9 location ID represents rack/slot/instance/port. Based on the requirement, the network administrators can use the displayed values of this command for monitoring and troubleshooting.

```
Router#show controllers optics 0/0/0/9 observable-info
Observable Information

[eSNR Media Input]
Unit: dB
Id      Value      TCAWarn      LowThreshWarn      HighThresWarn      LowThreshAlarm
HighThreshAlarm      TCAWarn      TCAAlarm

          Low High      Low High
Lane0    21.30      n   n      0.00      0.00      0.00
0.00
Lane1    22.05      n   n      0.00      0.00      0.00
0.00
Lane2    22.62      n   n      0.00      0.00      0.00
0.00
Lane3    22.05      n   n      0.00      0.00      0.00
```

```

0.00          n  n  n  n

[PAM4 Level Transition Parameter Media Input]
Unit: dB
Id      Value          LowThreshWarn      HighThresWarn      LowThreshAlarm
HighThreshAlarm      TCASWarn      TCAAlarm

          Low High      Low High
Lane0    47.79          0.00              0.00              0.00
0.00          n  n  n  n
Lane1    54.70          0.00              0.00              0.00
0.00          n  n  n  n
Lane2    64.34          0.00              0.00              0.00
0.00          n  n  n  n
Lane3    59.64          0.00              0.00              0.00
0.00          n  n  n  n

[Pre-FEC BER Minimum Media Input]
Unit: n/a
Id      Value          LowThreshWarn      HighThresWarn      LowThreshAlarm
HighThreshAlarm      TCASWarn      TCAAlarm

          Low High      Low High
Module  0.000E+00      0.000E+00          0.000E+00          0.000E+00
0.000E+00      n  n  n  n

[Pre-FEC BER Minimum Host Input]
Unit: n/a
Id      Value          LowThreshWarn      HighThresWarn      LowThreshAlarm
HighThreshAlarm      TCASWarn      TCAAlarm

          Low High      Low High
Module  0.000E+00      0.000E+00          0.000E+00          0.000E+00
0.000E+00      n  n  n  n

[Pre-FEC BER Maximum Media Input]
Unit: n/a
Id      Value          LowThreshWarn      HighThresWarn      LowThreshAlarm
HighThreshAlarm      TCASWarn      TCAAlarm

          Low High      Low High
Module  0.000E+00      0.000E+00          0.000E+00          0.000E+00
0.000E+00      n  n  n  n

[Pre-FEC BER Maximum Host Input]
Unit: n/a
Id      Value          LowThreshWarn      HighThresWarn      LowThreshAlarm
HighThreshAlarm      TCASWarn      TCAAlarm

          Low High      Low High
Module  0.000E+00      0.000E+00          0.000E+00          0.000E+00
0.000E+00      n  n  n  n

[Pre-FEC BER Average Media Input]
Unit: n/a
Id      Value          LowThreshWarn      HighThresWarn      LowThreshAlarm
HighThreshAlarm      TCASWarn      TCAAlarm

          Low High      Low High
Module  0.000E+00      0.000E+00          0.000E+00          0.000E+00
0.000E+00      n  n  n  n

[Pre-FEC BER Average Host Input]
Unit: n/a

```

```

Id      Value      LowThreshWarn      HighThresWarn      LowThreshAlarm
HighThreshAlarm      TCAWarn      TCAAlarm

                                Low High      Low High
Module  0.000E+00      0.000E+00      0.000E+00      0.000E+00
0.000E+00      n  n      n  n

[Pre-FEC BER Current Media Input]
Unit: n/a
Id      Value      LowThreshWarn      HighThresWarn      LowThreshAlarm
HighThreshAlarm      TCAWarn      TCAAlarm

                                Low High      Low High
Module  0.000E+00      0.000E+00      0.000E+00      0.000E+00
0.000E+00      n  n      n  n

[Pre-FEC BER Current Host Input]
Unit: n/a
Id      Value      LowThreshWarn      HighThresWarn      LowThreshAlarm
HighThreshAlarm      TCAWarn      TCAAlarm

                                Low High      Low High
Module  0.000E+00      0.000E+00      0.000E+00      0.000E+00
0.000E+00      n  n      n  n

[FERC Minimum Media Input]
Unit: n/a
Id      Value      LowThreshWarn      HighThresWarn      LowThreshAlarm
HighThreshAlarm      TCAWarn      TCAAlarm

                                Low High      Low High
Module  0.000E+00      0.000E+00      0.000E+00      0.000E+00
0.000E+00      n  n      n  n

[FERC Minimum Host Input]
Unit: n/a
Id      Value      LowThreshWarn      HighThresWarn      LowThreshAlarm
HighThreshAlarm      TCAWarn      TCAAlarm

                                Low High      Low High
Module  0.000E+00      0.000E+00      0.000E+00      0.000E+00
0.000E+00      n  n      n  n

[FERC Maximum Media Input]
Unit: n/a
Id      Value      LowThreshWarn      HighThresWarn      LowThreshAlarm
HighThreshAlarm      TCAWarn      TCAAlarm

                                Low High      Low High
Module  0.000E+00      0.000E+00      0.000E+00      0.000E+00
0.000E+00      n  n      n  n

[FERC Maximum Host Input]
Unit: n/a
Id      Value      LowThreshWarn      HighThresWarn      LowThreshAlarm
HighThreshAlarm      TCAWarn      TCAAlarm

                                Low High      Low High
Module  0.000E+00      0.000E+00      0.000E+00      0.000E+00
0.000E+00      n  n      n  n

[FERC Average Media Input]
Unit: n/a
Id      Value      LowThreshWarn      HighThresWarn      LowThreshAlarm

```

```

HighThreshAlarm      TCAWarn  TCAAlarm

                        Low High  Low High
Module  0.000E+00    0.000E+00    0.000E+00    0.000E+00
0.000E+00    n   n     n   n

[FERC Average Host Input]
Unit: n/a
Id      Value
HighThreshAlarm  TCAWarn  TCAAlarm

                        Low High  Low High
Module  0.000E+00    0.000E+00    0.000E+00    0.000E+00
0.000E+00    n   n     n   n

[FERC Current Media Input]
Unit: n/a
Id      Value
HighThreshAlarm  TCAWarn  TCAAlarm

                        Low High  Low High
Module  0.000E+00    0.000E+00    0.000E+00    0.000E+00
0.000E+00    n   n     n   n

[FERC Current Host Input]
Unit: n/a
Id      Value
HighThreshAlarm  TCAWarn  TCAAlarm

                        Low High  Low High
Module  0.000E+00    0.000E+00    0.000E+00    0.000E+00
0.000E+00    n   n     n   n
    
```

View Coherent Optical Transceiver Module Parameters

Table 86: Feature History Table

Feature Name	Release Information	Description
View Coherent Optical Transceiver Module Parameters	Release 25.1.1	Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*) *This feature is supported on: <ul style="list-style-type: none"> • 8712-MOD-M • 8011-4G24Y4H-I

Feature Name	Release Information	Description
View Coherent Optical Transceiver Module Parameters	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-12TH24FH-E • 88-LC1-36EH • 88-LC1-52Y8H-EM
View Coherent Optical Transceiver Module Parameters	Release 24.1.1	<p>You can now monitor the overall functioning and status of the coherent optical transceiver. This is possible because you can now view the module state and data path state of the optical transceivers, which give you an insight into the current state of the optical transceiver.</p> <p>This feature modifies the output of the show controller optics command.</p>

The display of the [show controller optics](#) command is now improved to include information about the module state and datapath state of the optical transceivers. You can view these details using the **information all** and **information counters** keywords of the [show controller optics](#) command.

Module State Machine (MSM) defines host-module interactions and behavioral characteristics of the optical module, such as the initialization of the management interface and the module power mode. The Module State field provides the current status of the optical transceiver. The optical transceiver can be in any one of the following states:

- Low power
- Power up
- Ready
- Power down
- Fault

Data Path State Machine (DPSM) defines the host-module interactions and behavioral characteristics needed for the initialization of one particular data path, which represents the signal flow and signal processing of any one instance of one type of application.

The Datapath State field provides the current state of the data path on each host lane. It represents the initialization status of the resources associated with a data path in response to host configuration settings or commands.

By default, the data paths begin initializing when the module state is ready. The data path state is dependent on the module state. The host uses the activated data path to carry traffic.



Note For additional information on the module state and datapath state, see the [Common Management Interface Specification](#).

Verification

The **show controllers optics r/s/i/p information counters** command displays the parameters of the optical transceiver present in the 0/0/0/8 location ID. The 0/0/0/8 location ID represents rack/slot/instance/port. Based on the requirement, the network administrators can use the displayed values of this command for monitoring and troubleshooting. This example displays the details when the optical transceiver operates in the transponder mode.

```
Router#show controllers optics 0/0/0/8 information counters
Fri Feb 16 11:06:31.415 UTC
```

```
Module State : Ready
```

```
Datapath State [Client-0]: TX Turn On
```

```
Acquisition Counter:    INVALID
```

```
HOST SIDE ALARM COUNTERS
```

```
=====
Host-Intf-0-FDD-Alarm-Counter[0]                Host-Intf-0-FED-Alarm-Counter[0]
```

```
HOST SIDE FEC-BER FEC-FERC CURRENT VALUES
```

```
=====
Host-Intf-0-FEC-BER[0.00E+00]                   Host-Intf-0-FEC-FERC[0.00E+00]
```

```
Supported Loopback Types :
```

```
=====
[1.] Media Internal
[2.] Media Line
[3.] Host Line
[4.] Host Internal
[5.] Host Per Lane
[6.] Media Per Lane
[7.] Simultaneous Media Host
```

```
Unsupported Loopback Types :
```

```
=====
```

```
Media Configured Loopback : Media Loopback None
Media Applied Loopback    : Media Loopback None
```

```
Host Configured Loopback : Host Loopback None
Host Applied Loopback    : Host Loopback None
```

```
FW Upgrade Capability Mode:
=====
Supports Both Warm & Cold boot
Supports Cold boot only
```

This example displays the details when the optical transceiver operates in the muxponder mode.

```
Router#show controllers optics 0/0/0/29 information all
Fri Feb 16 11:06:31.415 UTC
```

Module State : Ready

Datapath State [Client-0]: Activated

Datapath State [Client-1]: Activated

Datapath State [Client-2]: Activated

Datapath State [Client-3]: Activated

Acquisition Counter: INVALID

HOST SIDE ALARM COUNTERS

```
=====
Host-Intf-0-FDD-Alarm-Counter[0]          Host-Intf-0-FED-Alarm-Counter[0]
Host-Intf-1-FDD-Alarm-Counter[0]          Host-Intf-0-FED-Alarm-Counter[0]
Host-Intf-2-FDD-Alarm-Counter[0]          Host-Intf-0-FED-Alarm-Counter[0]
Host-Intf-3-FDD-Alarm-Counter[0]          Host-Intf-0-FED-Alarm-Counter[0]
```

HOST SIDE FEC-BER FEC-FERC CURRENT VALUES

```
=====
Host-Intf-0-FEC-BER[0.00E+00]             Host-Intf-0-FEC-FERC[0.00E+00]
Host-Intf-1-FEC-BER[0.00E+00]             Host-Intf-1-FEC-FERC[0.00E+00]
Host-Intf-2-FEC-BER[0.00E+00]             Host-Intf-2-FEC-FERC[0.00E+00]
Host-Intf-3-FEC-BER[0.00E+00]             Host-Intf-3-FEC-FERC[0.00E+00]
```

Supported Loopback Types :

```
=====
[1.] Media Internal
[2.] Media Line
[3.] Host Line
[4.] Host Internal
[5.] Host Per Lane
[6.] Media Per Lane
[7.] Simultaneous Media Host
```

Display of Alarms for Coherent Optical Transceivers

Table 87: Feature History Table

Feature Name	Release Information	Description
View Additional Alarms for Coherent Optical Transceiver	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8712-MOD-M • 8011-4G24Y4H-I
View Additional Alarms for Coherent Optical Transceiver	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-12TH24FH-E • 88-LC1-36EH • 88-LC1-52Y8H-EM
View Additional Alarms for Coherent Optical Transceiver	Release 24.1.1	<p>You can now monitor the optical transceivers for proper functioning and identify the cause of any malfunction. This is made possible because the output of the show alarms command is now enhanced to display the additional media lane alarms for coherent optical transceivers.</p>

Alarms are raised when the optical transceiver malfunctions.

The following media lane alarms are now reported in the **show alarms** command:

- TX loss of alignment - indicates loss of signal alignment on the transmitter.
- TX out of alignment - indicates that the signal on the transmitter (TX) is out of alignment.

- TX CMU loss of lock - indicates that the transmitter (TX) has lost connection (locked) with the external clock (clock monitor unit).
- TX reference clock loss of lock - indicates that the transmitter (TX) has lost connection (locked) with the reference clock.
- TX deskew loss of lock - for traffic flow in the TX direction, this alarm indicates that the end receiver cannot align the physical lanes using alignment marker.
- TX FIFO error - indicates signal FIFO error on the transmitter.
- RX demodulator loss of lock - indicates that the media demodulator cannot achieve lock.
- RX CDC loss of lock - indicates that the receiver (RX) has lost connection (locked) with the external clock.
- RX loss of alignment - indicates that the signal alignment on the receiver (RX) is lost.
- RX out of alignment - indicates that the signal on the receiver (RX) is out of alignment.
- RX deskew loss of lock - for traffic flow in the RX direction, this alarm indicates that the end receiver cannot align the physical lanes using alignment marker.
- RX FIFO error - indicates signal FIFO error on the receiver.
- RX FEC excessive degrade - indicates that the signal has reached or exceeded FED threshold.
- RX FEC detected degrade - indicates that the signal has reached or exceeded FDD threshold.
- Remote degrade - indicates remote signal degradation.
- Local degrade - indicates local signal degradation.
- Remote Phy fault - indicates remote signal fault (RPF).

Restrictions

The optical transceivers may not display these alarms if:

- Optical transceiver is disabled and the optical transceiver is in the **secondary admin-state maintenance** mode.
- Higher priority alarms such as improper removal and loss of signal (LOS) are reported. In such instances, these alarms can be viewed using the **show alarms brief suppressed** command.

Verification

The following **show alarms** command displays the alarms for the coherent optical transceivers.

```
Router#show alarms brief system active
```

```
-----
Active Alarms
-----
Location          Severity          Group              Set Time           Description
```

```

-----
0/RP0/CPU0      Major      Software      03/30/2023 12:30:39 UTC      Communications
Failure With Cisco Licensing Cloud

0/RP0/CPU0      Minor      Software      04/02/2023 13:55:38 UTC      Optics0/0/0/31 -
hw_optics:  RX POWER LANE-0 HIGH WARNING

0/RP0/CPU0      Major      Software      04/02/2023 13:55:38 UTC      Optics0/0/0/31 -
hw_optics:  Optics media rx signal power high warning

0/RP0/CPU0      Major      Software      04/02/2023 14:00:01 UTC      Optics0/0/0/29 -
hw_optics:  Optics media tx fifo error

0/RP0/CPU0      Major      Software      04/02/2023 14:00:01 UTC      Optics0/0/0/29 -
hw_optics:  Optics media tx loss of alignment

0/RP0/CPU0      Major      Software      04/02/2023 14:00:01 UTC      Optics0/0/0/29 -
hw_optics:  Optics media tx out of alignment

0/RP0/CPU0      Major      Software      04/02/2023 14:00:01 UTC      Optics0/0/0/29 -
hw_optics:  Optics media tx CMU loss of lock

0/RP0/CPU0      Major      Software      04/02/2023 14:00:01 UTC      Optics0/0/0/29 -
hw_optics:  Optics media tx reference clock loss of lock

0/RP0/CPU0      Major      Software      04/02/2023 14:00:01 UTC      Optics0/0/0/29 -
hw_optics:  Optics media tx deskew loss of lock

0/RP0/CPU0      Major      Software      04/02/2023 14:00:01 UTC      Optics0/0/0/29 -
hw_optics:  Optics media rx loss of alignment

0/RP0/CPU0      Major      Software      04/02/2023 14:00:01 UTC      Optics0/0/0/29 -
hw_optics:  Optics media rx out of alignment

0/RP0/CPU0      Major      Software      04/02/2023 14:00:01 UTC      Optics0/0/0/29 -
hw_optics:  Optics media rx fifo error

0/RP0/CPU0      Major      Software      04/02/2023 14:00:01 UTC      Optics0/0/0/29 -
hw_optics:  Optics media rx demodulation loss of lock

```

Pseudo Random Binary Sequence

The Pseudo Random Binary Sequence (PRBS) feature provides a robust mechanism for performing data integrity checks on router interfaces by generating and analyzing pseudo-random bit patterns. It operates simultaneously on both the transmit (Tx) and receive (Rx) directions without generating actual client traffic, ensuring that the optical links and interfaces maintain a good state and are error-free.

Feature Name	Release Information	Feature Description
PRBS support for high-speed optical interfaces	Release 25.4.1	<p>Introduced in this release (8800 [ASIC: Q200]).</p> <p>The Pseudo Random Binary Sequence (PRBS) feature provides data integrity checks on router interfaces by generating and analyzing pseudo-random bit patterns. It operates simultaneously on both the transmit (Tx) and receive (Rx) directions without generating actual client traffic, ensuring that the optical links and interfaces maintain a good state and are error-free.</p> <p>CLI:</p> <p>This feature introduces the following commands:</p> <ul style="list-style-type: none"> • controller Optics <code>prbs-mode <source/sink/source-sink></code> <code>pattern <pn15/pn31></code> <code><line/system></code> • show controller Optics <code>prbs-details</code>

The feature supports PRBS15 and PRBS31 patterns, which correspond to pseudo-random sequences of lengths $2^{15}-1$ and $2^{31}-1$ bits respectively.

PRBS is typically employed during system installations, or when replacing line cards or nodes, especially in Ethernet-based lossless fabric environments such as simple two-tier leaf and spine network designs.

PRBS operational modes

PRBS testing uses three primary operational modes to check data integrity on links without generating live client traffic

You can configure PRBS in one of the following modes on the optics controller.

- **Source mode** — The device continuously generates the PRBS signal on the line according to the configured pattern.
- **Sink mode** — The device locks onto the incoming PRBS signal, analyzes it, and reports any errors detected.
- **Source-Sink mode** — The device acts as both transmitter and receiver, generating the PRBS pattern and simultaneously verifying the incoming signal for errors.

Limitations for PRBS on optics controller

- The PRBS support is limited to modules that comply with the QSFP-DD Common Management Interface Specification (CMIS).
- User-defined pattern is not supported.
- Configuration is supported only on the main controller and is not supported on breakout controllers.

Configure PRBS on optics controller

Before you begin

Before enabling PRBS, ensure that the interfaces are unshut.

Ensure that optics controller is set in the maintenance mode using the `sec-admin-state maintenance` configuration.

Procedure

Step 1 Check the capabilities of the optical module.

Example:

```
RP/0/RP0/CPU0:ios#show controllers optics 0/2/0/28 prbs-capability-info
Wed Sep  3 16:26:42.376 UTC
```

```
Supported:
Mode: Source | Sink | Source-Sink
Pattern: PN9Q | PN13 | PN13Q | PN15 | PN15Q
Direction: Line | System
Error-Inject: None
```

```
Unsupported Combinations:
```

```
None
RP/0/RP0/CPU0:ios#
```

Step 2 Configure PRBS on the optics controller:

Example:

```
controller Optics0/0/0/18
 prbs mode source-sink pattern pn15q direction line
 sec-admin-state maintenance
!
```

Step 3 Verify that PRBS is configured on the optics controller. Verify that PRBS mode, pattern, and direction are configured using the `show controllers optics R/S/I/P prbs-info` command:

```
RP/0/0/CPU0:ios#show controllers Optics 0/0/0/18 prbs-info
PRBS -----
Port:      Optics0/0/0/31
```

```
-----PRBS details-----
```

```

PRBS Test      : Enable
PRBS Mode     : Source-Sink
PRBS Pattern  : PN31Q
PRBS Status   : Locked
PRBS Direction : Line
PRBS Configured Time : 18 Aug 01:40:23 (1755506423 seconds)
PRBS First Lock Established Time: 18 Aug 01:40:26 (1755506426 seconds)
Counter last updated timestamp : 18 Aug 01:47:16 (1755506836 seconds)

```

Lane	SNR (dBm) MaxBER	MaxSNR (dBm)	ErrorCount	TotalBits	BER
1	22.78 0.000e+00	23.11	11071	43558738601216	0.000e+00
2	22.04 0.000e+00	22.32	132435	43546486903424	0.000e+00
3	22.62 0.000e+00	22.78	53296	43589169331456	0.000e+00
4	22.18 0.000e+00	22.32	307982	43580449870336	0.000e+00

Lane	LockStatus LockLostTimestamp	LostCount	FoundCount	LockTime (secs)
1	Locked	0	1	417
2	Locked	0	1	417
3	Locked	0	1	417
4	Locked	0	1	417

Configure PRBS on coherent DSP controller

Before you begin

Before enabling PRBS, ensure that the interfaces are unshut.

Procedure

Step 1 Configure PRBS on the coherent DSP controller:

Example:

```

controller CoherentDsp0/0/0/14
  prbs mode source-sink pattern pn15q direction line
  sec-admin-state maintenance

```

Step 2 Verify that PRBS is configured on the coherent DSP controller. Verify the current performance monitoring parameters of the coherent DSP controller in 30 second intervals using the **show controllers optics R/S/I/P pm current 30-sec prbs** command:

```

RP/0/RP0/CPU0:ios#show controllers coherentDSP 0/1/0/14 pm current 30-sec prbs
Thu Oct 24 14:07:45.189 UTC

```

```

PRBS in the current interval [14:07:30 - 14:07:45 Thu Oct 24 2024]

```

```

PRBS current bucket type : Valid

EBC          : 0          Threshold : 0          TCA(enable) : NO
FOUND-COUNT  : 0          Threshold : 0          TCA(enable) : NO
LOST-COUNT   : 0          Threshold : 0          TCA(enable) : NO

FOUND-AT-TS  : NULL
LOST-AT-TS   : NULL

CONFIG-PTRN  : PRBS_PATTERN_PN31
STATUS       : LOCKED

Last clearing of "show controllers OTU" counters never
RP/0/RP0/CPU0:ios#

```

Loopback on Optical Transceivers

Table 88: Feature History Table

Feature Name	Release Information	Description
Loopback on Optical Transceivers	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8712-MOD-M • 8011-4G24Y4H-I
Loopback on Optical Transceivers	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-12TH24FH-E • 88-LC1-36EH • 88-LC1-52Y8H-EM

Feature Name	Release Information	Description
Loopback on Optical Transceivers	Release 7.11.1	<p>You can now easily detect link failures between the optical transceiver and an external device such as a router by creating a loopback within the transceiver itself. Enabling loopback detects the fault in the physical or network connections, such as, traffic loss or a faulty optical transceiver.</p> <p>The loopback configuration allows incoming traffic within the transceiver to be redirected back to its source. By analyzing the loopback signals received at the source, it becomes possible to detect physical connectivity failures or network issues, such as packet loss or a malfunctioning transceiver.</p> <p>The feature introduces these changes:</p> <p>CLI:</p> <p>Modified the controller optics command by adding the following keywords:</p> <ul style="list-style-type: none"> • host loopback internal • host loopback line • loopback internal • loopback line <p>The information loopback keyword is added to the show controller optics command.</p> <p>YANG Data Model:</p> <ul style="list-style-type: none"> • New XPath for <code>Cisco-IOS-XR-controller-optics-cfg.yang</code> <p>(see GitHub, YANG Data Models Navigator)</p>

You can now enable loopback functionality on the optical transceivers. Loopback is the process of redirecting inbound traffic or data signals from an optical transceiver back to the module itself. Re-routing traffic to its source enables utilization of the received data for diagnostic purposes, particularly in the identification and

resolution of physical connectivity issues or network-related problems, such as traffic loss or a faulty optical transceiver.

The optical transceiver is divided into two sides, the host side, which is positioned towards the router, and the media side, which is positioned towards the wire or cable media. It is possible to enable loopback on both the host side and media side of the optical transceiver.



Note Loopback can be performed only when the controller state is active (UP) and in the maintenance mode.

There are four types of loopback:

- Loopback Internal or Media Side Output Loopback
- Loopback Line or Media Side Input Loopback
- Host Loopback Internal or Host Side Input Loopback
- Host Loopback Line or Host Side Output Loopback

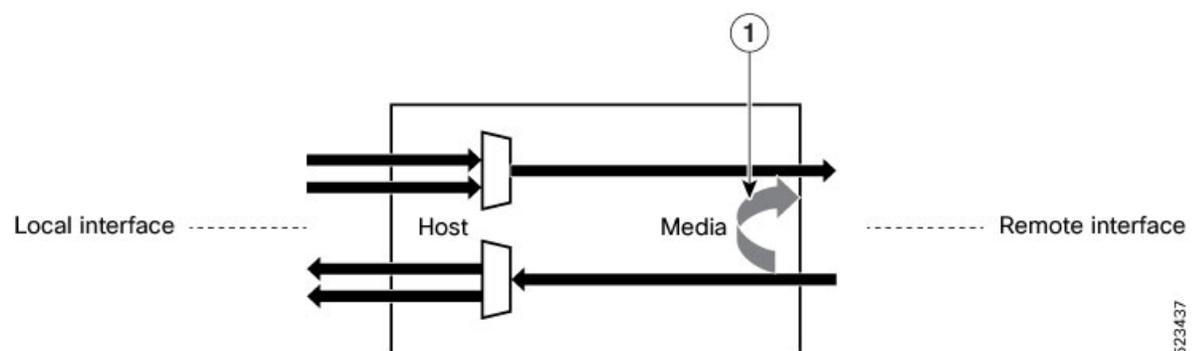


Note Configuring the internal loopback brings up the host interface and configuring the line loopback brings up the remote interface.

Media Side Input Loopback Configuration

In loopback line or media side input loopback, the signals received at the media side are looped back to the media side, indicating that the received data on the media is transmitted back to the media, that is, towards the remote interface. This is indicated by the arrow labeled as 1 in the illustration.

Figure 17: Media Side Input Loopback on the Optical Transceiver



Configuration Example

This example shows how to enable media side input loopback on the optical transceiver:

```
Router#config
Router(config)#controller optics 0/0/0/9
Router(config-Optics)#secondary-admin-state maintenance
Router(config-Optics)#loopback line
```

```

Loopback is a traffic-affecting operation
Router(config-Optics)#commit
Router(config-Optics)#end

```

Running Configuration

This example shows the running configuration of the media side input loopback on the optical transceiver:

```

Router#show run controller optics 0/0/0/9
controller Optics0/0/0/9
  secondary-admin-state maintenance
  loopback line
!
```

Verification

This example shows how to verify the media side input loopback configuration on the optical transceiver:

```

Router#show controller optics 0/0/0/9
Controller State: Up
Transport Admin State: Maintenance
Laser State: On
LED State: Green
FEC State: FEC ENABLED
Optics Status:

Optics Type: QSFPDD 400G FR4
Wavelength: 1301.00 nm
Loopback Host: None
Loopback Media: Line

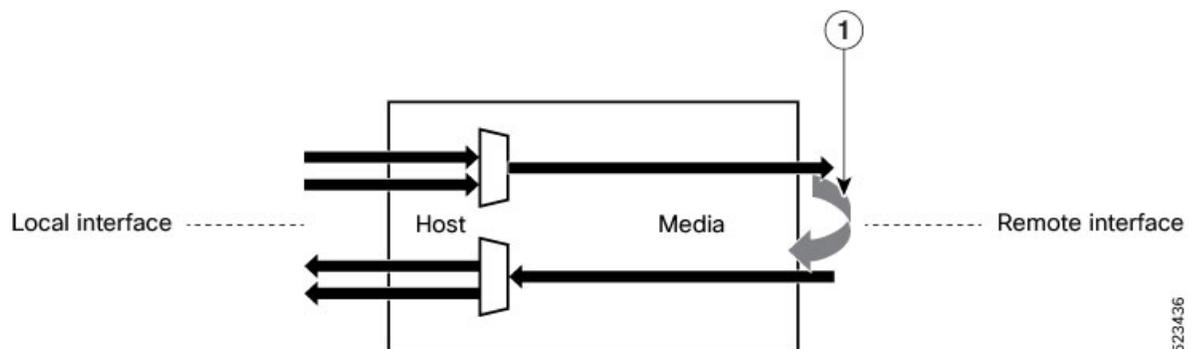
Alarm Status:
-----
Detected Alarms: None
LOS/LOL/Fault Status:
Performance Monitoring: Disable

```

Media Side Output Loopback

In loopback internal or media side output loopback, the loopback signal originating from the NPU is looped back to the same NPU on the media or line side, towards the remote interface. This is indicated by the arrow labeled as 1 in the illustration.

Figure 18: Media Side Output Loopback on the Optical Transceiver



Configuration Example

This example shows how to enable media side output loopback on the optical transceiver:

```
Router#config
Router(config)#controller optics 0/0/0/9
Router(config-Optics)#secondary-admin-state maintenance
Router(config-Optics)#loopback internal
Loopback is a traffic-affecting operation
Router(config-Optics)#commit
Router(config-Optics)#end
```

Running Configuration

This example shows the running configuration of the media side output loopback on the optical transceiver:

```
Router#show run controller optics 0/0/0/9
controller Optics0/0/0/9
  secondary-admin-state maintenance
  loopback internal
!
```

Verification

This example shows how to verify the media side output loopback configuration on the optical transceiver:

```
Router#show controller optics 0/0/0/9
Controller State: Up
Transport Admin State: Maintenance
Laser State: On
LED State: Green
FEC State: FEC ENABLED
Optics Status:

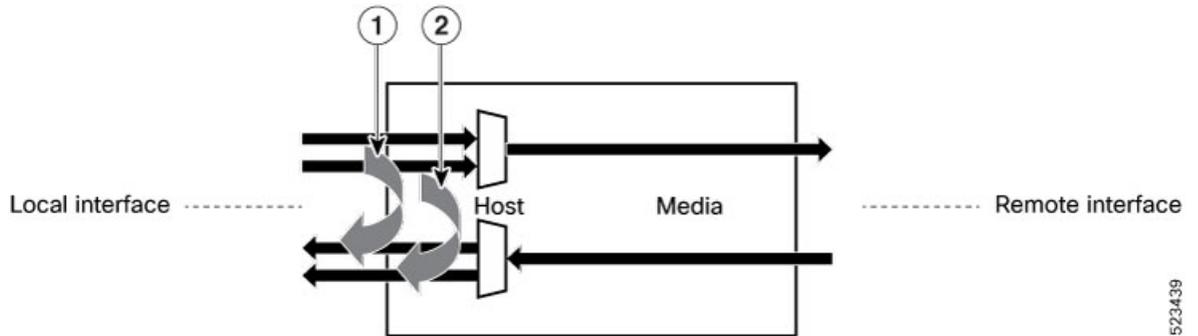
Optics Type: QSPDD 400G FR4
Wavelength: 1301.00 nm
Loopback Host: None
Loopback Media: Internal

Alarm Status:
-----
Detected Alarms: None
LOS/LOL/Fault Status:
Performance Monitoring: Disable
```

Host Side Input Loopback Configuration

In host loopback internal or host side input loopback, the loopback signal coming from the NPU is looped back to the NPU on the host, that is, towards the local interface. This is indicated by the arrows labeled as 1 and 2 in the illustration.

Figure 19: Host Side Input Loopback on the Optical Transceiver



523439

Configuration Example

This example shows how to enable host side input loopback on the optical transceiver:

```
Router#config
Router(config)#controller optics 0/0/0/9
Router(config-Optics)#secondary-admin-state maintenance
Router(config-Optics)#host loopback line
Loopback host is a traffic-affecting operation
Router(config-Optics)#commit
Router(config-Optics)#end
```

Running Configuration

This example shows the running configuration of the host side input loopback on the optical transceiver:

```
Router#show run controller optics 0/0/0/9
controller Optics0/0/0/9
  secondary-admin-state maintenance
  host loopback line
!
```

Verification

This example shows how to verify the host side input loopback configuration on the optical transceiver:

```
Router#show controller optics 0/0/0/9
Controller State: Up
Transport Admin State: Maintenance
Laser State: On
LED State: Green
FEC State: FEC ENABLED
Optics Status:

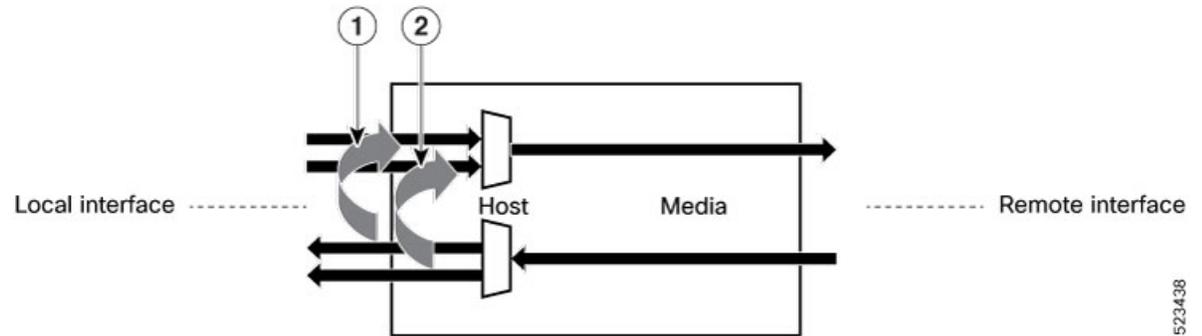
Optics Type: QSFPDD 400G FR4
Wavelength: 1301.00 nm
Loopback Host: Line
Loopback Media: None

Alarm Status:
-----
Detected Alarms: None
LOS/LOL/Fault Status:
Performance Monitoring: Disable
```

Host Side Output Loopback Configuration

In host loopback line or host side output loopback, the signals received at the host side are looped back to the host side, indicating that the received data on the host is transmitted back to the host, that is, towards the local interface. This is indicated by the arrows labeled as 1 and 2 in the illustration.

Figure 20: Host Side Output Loopback on the Optical Transceiver



Configuration Example

This example shows how to enable host side output loopback on the optical transceiver:

```
Router#config
Router(config)#controller optics 0/0/0/9
Router(config-Optics)#secondary-admin-state maintenance
Router(config-Optics)#host loopback internal
Loopback host is a traffic-affecting operation
Router(config-Optics)#commit
Router(config-Optics)#end
```

Running Configuration

This example shows the running configuration on the optical transceiver:

```
Router#show run controller optics 0/0/0/9
controller Optics0/0/0/9
  secondary-admin-state maintenance
  host loopback internal
!
```

Verification

This example shows how to verify the host side output loopback configuration on the optical transceiver:

```
Router#show controller optics 0/0/0/9
Controller State: Up
Transport Admin State: Maintenance
Laser State: On
LED State: Green
FEC State: FEC ENABLED
Optics Status:

Optics Type: QSFPDD 400G FR4
Wavelength: 1301.00 nm
Loopback Host: Internal
Loopback Media: None

Alarm Status:
```

```
-----  
Detected Alarms: None  
LOS/LOL/Fault Status:  
Performance Monitoring: Disable
```



CHAPTER 18

Configuring QDD Optical Line System

This chapter describes the QDD Optical Line System (OLS) and its supported configurations.

- [Overview of QDD optical line system, on page 420](#)
- [Supported Wavelength or Frequency Configuration, on page 422](#)
- [Functional Description of QDD OLS, on page 423](#)
- [QDD OLS Configurations, on page 424](#)
- [Use Case for QDD OLS pluggable, on page 434](#)
- [OLS Alarms Troubleshooting, on page 435](#)

Overview of QDD optical line system

Table 89: Feature History Table

Feature Name	Release Information	Description
QDD optical line system	Release 24.1.1	

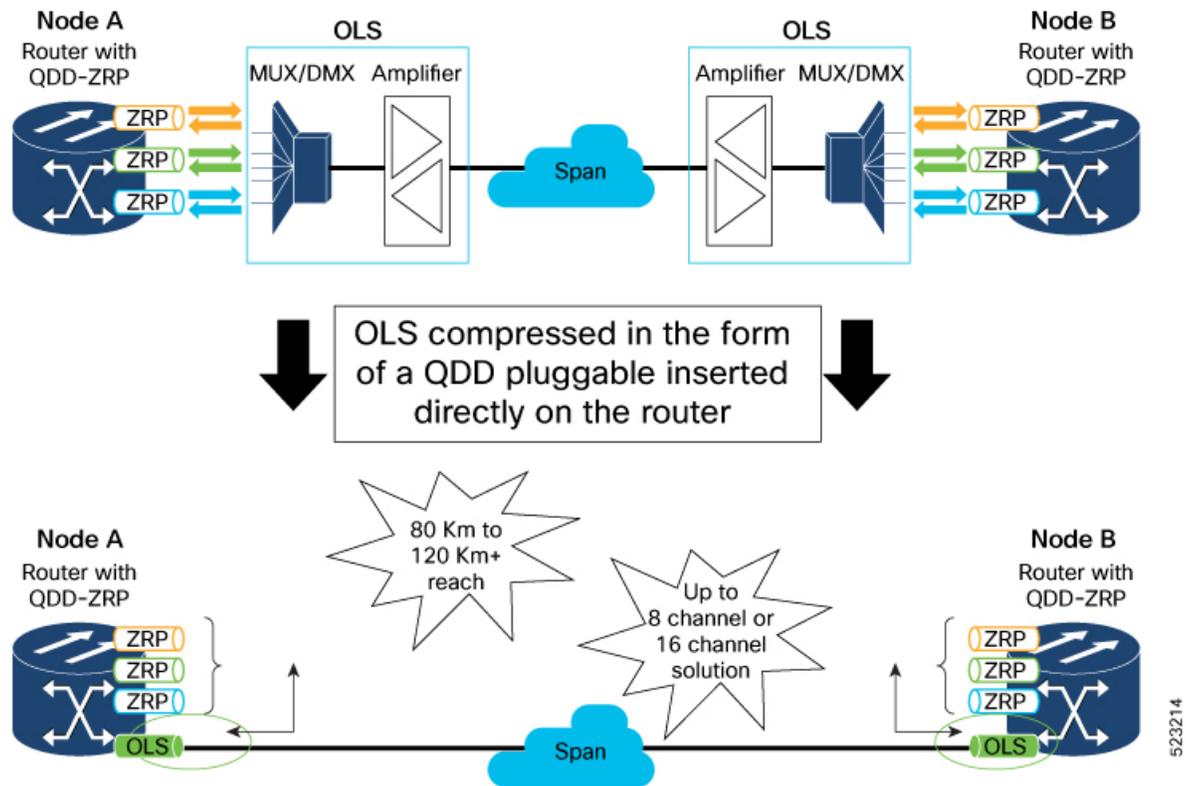
Feature Name	Release Information	Description
		<p>The QDD Optical Line System (OLS) is a new pluggable optical amplifier that interconnects two routers or switches for transmitting traffic on a limited number of coherent optical channels over a single span point-to-point link. With the QDD OLS pluggable, it's now possible to obtain the functionality of amplification into a QSFP-DD module that can be plugged into a port of the router or switch. The QDD OLS is supported on Cisco 8201, 8201-32FH, and 8201-24H8FH routers.</p> <p>The feature introduces these changes:</p> <ul style="list-style-type: none"> • CLI: <ul style="list-style-type: none"> • controller ots (QDD OLS) • rx-low-threshold • tx-low-threshold • ampli-control-mode • egress-ampli-gain • egress-ampli-power • egress-ampli-safety-control-mode • egress-ampli-osri • show controllers ots (QDD OLS) • YANG Data Model: <ul style="list-style-type: none"> • Cisco-IOS-XR-controller-ots-oper.yang • Cisco-IOS-XR-controller-ots-cfg.yang • Cisco-IOS-XR-pengine-oper.yang • Cisco-IOS-XR-pengine-cfg.yang • Cisco-IOS-XR-pengine-clear-act.yang

The QDD OLS is a pluggable optical amplifier that interconnects two routers or switches for transporting a limited number of coherent optical channels over a single span point-to-point link.

Currently, an optical line system (OLS) requires a separate Cisco Network Convergence System 1000 Series or Cisco Network Convergence System 2000 Series optical system with dedicated 48 or 64 channels MUX/DMX units and amplifiers. The new QDD OLS module provides amplification functionality in a QSFP module, while the passive cables provide MUX/DMX functionality. With this solution, you can compress a point-to-point DWDM system directly into the routing or switching platform.

The new optical line system can now transport 8 or 16 optical channels without any additional optical hardware unit. With the use of external passive MUX/DMX units (examples for these) along with this setup, it's also possible to obtain up to 32 optical channels.

Figure 21: QDD Optical Line System



523214

Supported Wavelength or Frequency Configuration

For each channel supported through ONS-BRK-CS-8LC or ONS-BRK-CS-16LC passive/mux cable, the wavelength or the frequency must be configured according to the table below.

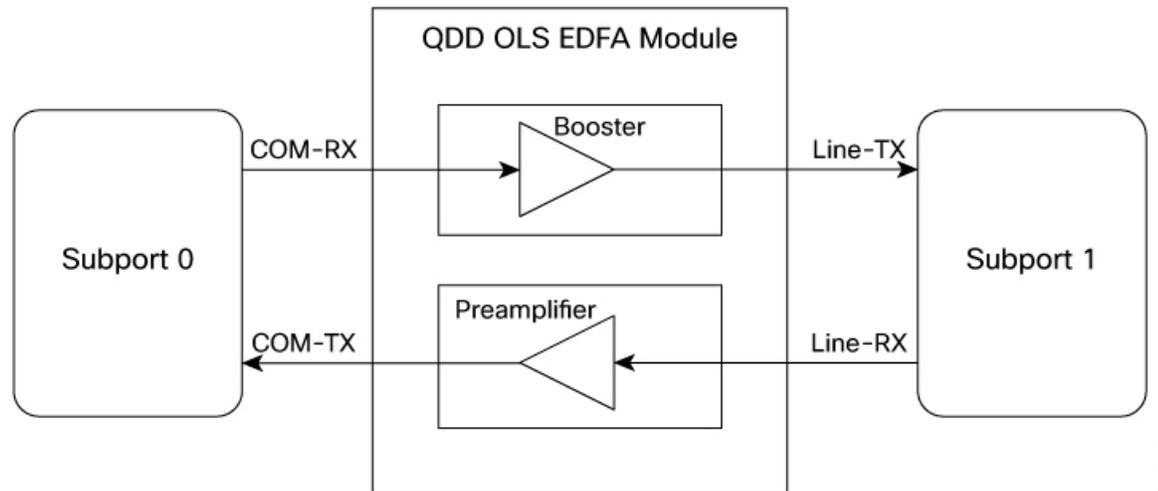
Table 90: QDD OLS Operating Signal Wavelength Range

Channel Spacing	Total Bandwidth	Wavelength		Frequency	
		Start	End	Start	End
8 channels - 200 GHz spaced	19.2 nm 2.4 THz	1539.1 nm	1558.4 nm	192.375 THz	194.775 THz
16 channels - 100 GHz spaced					

Functional Description of QDD OLS

The QDD OLS pluggable contains the COM side and the Line side as shown in the figure below:

Figure 22: Functional Description of QDD OLS



Each physical port of the QDD OLS pluggable is represented as two ots controllers (subport 0 and subport 1). COM port is subport 0 and Line port is subport 1.

The Gain of the Booster is associated to subport 1 while the gain of the Preamplifier is associated to subport 0.

Controller	Optical Ports
ots R/S/I/P/0	COM-RX (booster input)
	COM-TX (preamplifier output)
ots R/S/I/P/1	LINE-RX (preamplifier input)
	LINE-TX (booster output)

QDD OLS Configurations

The following section contains the QDD OLS configuration details.

Configuring the Operational Mode, Amplifier Gain, and Amplifier Output Power

You can configure the mode of operation of the OLS pluggable to either gain control or power control mode.

In the gain control mode, you can configure the desired gain value of the OLS pluggable.

In the power control mode, you can configure the desired output power (TX) of the OLS pluggable.

Gain Control Operational Mode and Amplifier Gain Configuration Example

The following example shows how to configure the gain control operational mode and the amplifier gain of the OLS pluggable:

```
Router#config
Router(config)#controller ots 0/0/2/1/0
Router(config-Ots)#ampli-control-mode manual
Router(config-Ots)#egress-ampli-gain 150
Router(config-Ots)#commit
Router(config-Ots)#exit
Router(config)#exit
```

Running Configuration

This example shows the running configuration for the OLS pluggable:

```
Router#show run controller ots 0/0/2/1/0
controller Ots0/0/2/1/0
  ampli-control-mode manual
  egress-ampli-gain 150
!
```

Verification

This example shows how to verify the configured gain control operational mode and amplifier gain of the OLS pluggable:

```
Router#show controllers ots 0/0/2/1/0
Thu Mar 23 21:33:49.862 UTC

Controller State: Up

Transport Admin State: In Service

LED State: Green

Alarm Status:
-----
Detected Alarms: None

Alarm Statistics:
-----
RX-LOS-P = 4
RX-LOC = 0
TX-POWER-FAIL-LOW = 1
INGRESS-AUTO-LASER-SHUT = 0
INGRESS-AUTO-POW-RED = 0
```

```

INGRESS-AMPLI-GAIN-LOW = 0
INGRESS-AMPLI-GAIN-HIGH = 0
EGRESS-AUTO-LASER-SHUT = 0
EGRESS-AUTO-POW-RED = 0
EGRESS-AMPLI-GAIN-LOW = 4
EGRESS-AMPLI-GAIN-HIGH = 1
HIGH-TX-BR-PWR = 0
HIGH-RX-BR-PWR = 0
SPAN-TOO-SHORT-TX = 0
SPAN-TOO-SHORT-RX = 0

```

Parameter Statistics:

```

-----
Total Tx Power = 16.72 dBm
Rx Signal Power = -22.29 dBm
Tx Signal Power = 16.53 dBm
Egress Ampli Gain = 14.7 dB
Egress Ampli OSRI = OFF
Egress Ampli Force APR = OFF

```

Configured Parameters:

```

-----
Egress Ampli Gain = 15.0 dB
Egress Ampli Power = 4.0 dBm
Egress Ampli OSRI = OFF
Egress Ampli Force APR = OFF
Ampli Control mode = Manual
Rx Low Threshold = -30.0 dBm
Tx Low Threshold = -5.0 dBm

```

```

Temperature = 27.92 Celsius
Voltage = 3.33 V

```

Optical Module Details

```

Optics type           : QDD DUAL EDFA
Name                  : CISCO-ACCELINK
OUI Number            : 00.00.00
Part Number           : EDFA-211917-QDD
Rev Number            : 26
Serial Number         : ACW2650Z004
PID                   : ONS-QDD-OLS
VID                   : ves1
Firmware Version      : 2.06
Date Code (yy/mm/dd) : 22/12/31
Fiber Connector Type  : CS

```

Power Control Operational Mode and Amplifier Output Power Configuration Example

The following example shows how to configure the power control operational mode and the amplifier output power of the OLS pluggable :

```

Router#config
Router(config)#controller ots 0/0/2/1/0
Router(config-Ots)#ampli-control-mode powermode
Router(config-Ots)#egress-ampli-power 50
Router(config-Ots)#commit
Router(config-Ots)#exit
Router(config)#exit

```

Running Configuration

This example shows the running configuration for the OLS pluggable:

```
Router(controller-Ots)#show run controller ots 0/0/2/1/0
controller Ots0/0/2/1/0
  ampli-control-mode powermode
  egress-ampli-power 50
!
```

Verification

This example shows how to verify the configured power control operational mode and amplifier output power of the OLS pluggable:

```
Router#show controllers ots 0/0/2/1/0
Thu Jun 1 08:56:37.236 UTC
```

```
Controller State: Up
```

```
Transport Admin State: In Service
```

```
LED State: Green
```

```
Alarm Status:
-----
Detected Alarms: None

Alarm Statistics:
-----
RX-LOS-P = 4
RX-LOC = 0
TX-POWER-FAIL-LOW = 1
INGRESS-AUTO-LASER-SHUT = 0
INGRESS-AUTO-POW-RED = 0
INGRESS-AMPLI-GAIN-LOW = 0
INGRESS-AMPLI-GAIN-HIGH = 0
EGRESS-AUTO-LASER-SHUT = 0
EGRESS-AUTO-POW-RED = 0
EGRESS-AMPLI-GAIN-LOW = 4
EGRESS-AMPLI-GAIN-HIGH = 1
HIGH-TX-BR-PWR = 0
HIGH-RX-BR-PWR = 0
SPAN-TOO-SHORT-TX = 0
SPAN-TOO-SHORT-RX = 0

Parameter Statistics:
-----
Total Tx Power = 5.00 dBm
Rx Signal Power = -22.29 dBm
Tx Signal Power = 4.99 dBm
Egress Ampli Gain = 3.2 dB
Egress Ampli OSRI = OFF
Egress Ampli Force APR = OFF

Configured Parameters:
-----
Egress Ampli Gain = 15.0 dB
Egress Ampli Power = 5.0 dBm
Egress Ampli OSRI = OFF
Egress Ampli Force APR = OFF
Ampli Control mode = Power
Rx Low Threshold = -30.0 dBm
Tx Low Threshold = -5.0 dBm

Temperature = 29.33 Celsius
Voltage = 3.34 V
```

```
Optical Module Details
```

```

Optics type           : QDD DUAL EDFA
Name                  : CISCO-ACCELINK
OUI Number            : 00.00.00
Part Number           : EDFA-211917-QDD
Rev Number            : 26
Serial Number         : ACW2650Z004
PID                   : ONS-QDD-OLS
VID                   : ves1
Firmware Version      : 2.06
Date Code (yy/mm/dd) : 22/12/31
Fiber Connector Type  : CS

```

Configuring the Low-Threshold Power

You can set the low-threshold power of the optical signal (RX and TX) that can be received or transmitted from the OLS pluggable. The TX and RX low-threshold power values are used for raising and clearing the TX and RX power fail alarms.

Low-Threshold Power Configuration Example

The following example shows how to configure the optical receive (RX) low-threshold power on the OLS pluggable:

```

Router#config
Router(config)#controller ots 0/0/2/1/0
Router(config-Ots)#rx-low-threshold -200
Router(config-Ots)#commit
Router(config-Ots)#exit
Router(config)#exit

```



Note To configure the optical transmit (TX) low-threshold power on the OLS pluggable, use the **tx-low-threshold tx-low** command.

Running Configuration

This example shows the running configuration for the OLS pluggable:

```

Router#show run controller ots 0/0/2/1/0
controller Ots0/0/2/1/0
  rx-low-threshold -200
!
!

```

Verification

This example shows how to verify the configured optical receive (RX) low-threshold power value for the OLS pluggable:

```

Router#show controllers ots 0/0/2/1/0

Controller State: Up

Transport Admin State: In Service

LED State: Yellow

Alarm Status:
-----

```

Detected Alarms:
RX-LOS-P

Alarm Statistics:

```
-----
RX-LOS-P = 0
RX-LOC = 0
TX-POWER-FAIL-LOW = 0
INGRESS-AUTO-LASER-SHUT = 0
INGRESS-AUTO-POW-RED = 0
INGRESS-AMPLI-GAIN-LOW = 0
INGRESS-AMPLI-GAIN-HIGH = 0
EGRESS-AUTO-LASER-SHUT = 0
EGRESS-AUTO-POW-RED = 0
EGRESS-AMPLI-GAIN-LOW = 0
EGRESS-AMPLI-GAIN-HIGH = 0
HIGH-TX-BR-PWR = 0
HIGH-RX-BR-PWR = 0
SPAN-TOO-SHORT-TX = 0
SPAN-TOO-SHORT-RX = 0
```

Parameter Statistics:

```
-----
Total Tx Power = -50.00 dBm
Rx Signal Power = -50.00 dBm
Tx Signal Power = -50.00 dBm
Egress Ampli Gain = 0.0 dB
Egress Ampli OSRI = OFF
Egress Ampli Force APR = OFF
```

Configured Parameters:

```
-----
Egress Ampli Gain = 20.0 dB
Egress Ampli Power = 8.0 dBm
Egress Ampli OSRI = OFF
Egress Ampli Force APR = OFF
Ampli Control mode = Manual
Rx Low Threshold = -20.0 dBm
Tx Low Threshold = -5.0 dBm
```

Temperature = 14.29 Celsius
Voltage = 3.37 V

Optical Module Details

```
Optics type       : QDD DUAL EDFA
Name              : CISCO-ACCELINK
OUI Number       : 00.00.00
Part Number      : EDFA-211917-QDD
Rev Number       : 26
Serial Number    : ACW2650Z004
PID              : ONS-QDD-OLS
VID              : ves1
Firmware Version : 2.06
Date Code (yy/mm/dd) : 22/12/31
Fiber Connector Type : CS
```

Configuring the Optical Safety Remote Interlock (OSRI)

To shut down the amplifier, use the Optical Safety Remote Interlock (OSRI) configuration. This configuration is used during the maintenance of the pluggable, debugging scenarios, and when the OLS pluggable isn't in use. With this configuration enabled, the output power can still be a maximum of -15dBm based on the input power.

OSRI Configuration Example

The following example shows how to configure the Optical Safety Remote Interlock (OSRI) on the OLS pluggable:

```
Router#config
Router(config)#controller ots 0/0/2/1/0
Router(config-Ots)#egress-ampli-osri on
Router(config-Ots)#commit
Router(config-Ots)#exit
Router(config)#exit
```

Running Configuration

This example shows the running configuration for the OLS pluggable:

```
Router#show run controller optics 0/0/2/1/0
controller Ots0/0/2/1/0
  egress-ampli-osri on
!
```

Verification

This example shows how to verify the configured OSRI for the OLS pluggable:

```
Router#show controllers ots 0/0/2/1/0

Thu Jun  1 09:04:10.335 UTC

Controller State: Up

Transport Admin State: In Service

LED State: Green

Alarm Status:
-----
Detected Alarms: None

Alarm Statistics:
-----
RX-LOS-P = 4
RX-LOC = 0
TX-POWER-FAIL-LOW = 1
INGRESS-AUTO-LASER-SHUT = 0
INGRESS-AUTO-POW-RED = 0
INGRESS-AMPLI-GAIN-LOW = 0
INGRESS-AMPLI-GAIN-HIGH = 0
EGRESS-AUTO-LASER-SHUT = 0
EGRESS-AUTO-POW-RED = 0
EGRESS-AMPLI-GAIN-LOW = 4
EGRESS-AMPLI-GAIN-HIGH = 1
HIGH-TX-BR-PWR = 0
HIGH-RX-BR-PWR = 0
SPAN-TOO-SHORT-TX = 0
SPAN-TOO-SHORT-RX = 0
```

```

Parameter Statistics:
-----
Total Tx Power = -50.00 dBm
Rx Signal Power = -22.36 dBm
Tx Signal Power = -50.00 dBm
Egress Ampli Gain = 0.0 dB
Egress Ampli OSRI = ON
Egress Ampli Force APR = OFF

Configured Parameters:
-----
Egress Ampli Gain = 15.0 dB
Egress Ampli Power = 5.0 dBm
Egress Ampli OSRI = ON
Egress Ampli Force APR = OFF
Ampli Control mode = Power
Rx Low Threshold = -30.0 dBm
Tx Low Threshold = -5.0 dBm

Temperature = 27.90 Celsius
Voltage = 3.34 V

Optical Module Details

Optics type           : QDD DUAL EDFA
Name                  : CISCO-ACCELINK
OUI Number            : 00.00.00
Part Number           : EDFA-211917-QDD
Rev Number            : 26
Serial Number         : ACW2650Z004
PID                   : ONS-QDD-OLS
VID                   : ves1
Firmware Version     : 2.06
Date Code (yy/mm/dd) : 22/12/31
Fiber Connector Type  : CS

```

Configuring Safety Control Mode

You can enable safety control mode only on subport 1.

With `safety-control-mode` set as **auto** and if LOS is detected on the line RX, the line TX normalizes the signal output power to 8 dBm and the ALS (automatic laser shutdown) and APR (automatic power reduction) alarms are raised.

Safety Control Configuration Example

The following example shows how to enable safety control mode on the OLS pluggable (on subport 1):

```

Router#config
Router (config)#controller ots 0/0/2/1/1
Router (config-Ots)#egress-ampli-safety-control-mode auto
Router (config-Ots)#commit
Router (config-Ots)#exit
Router (config)#exit

```

Running Configuration

This example shows the running configuration for the OLS pluggable:

```

Router#show run controller ots 0/0/2/1/1
controller Ots0/0/2/1/1
  ampli-control-mode manual

```

```

egress-ampli-gain 230
egress-ampli-safety-control-mode auto
!

```

Verification

This example shows how to verify the configured safety control mode:

```

Router#show controllers ots 0/0/2/1/1

Thu Jun  1 09:04:17.550 UTC

Controller State: Down

Transport Admin State: In Service

LED State: Yellow

Alarm Status:
-----
Detected Alarms:
                RX-LOS-P
                EGRESS-AUTO-LASER-SHUT
                EGRESS-AUTO-POW-RED
                EGRESS-AMPLI-GAIN-HIGH

Alarm Statistics:
-----
RX-LOS-P = 12
RX-LOC = 0
TX-POWER-FAIL-LOW = 1
INGRESS-AUTO-LASER-SHUT = 0
INGRESS-AUTO-POW-RED = 0
INGRESS-AMPLI-GAIN-LOW = 0
INGRESS-AMPLI-GAIN-HIGH = 0
EGRESS-AUTO-LASER-SHUT = 13
EGRESS-AUTO-POW-RED = 13
EGRESS-AMPLI-GAIN-LOW = 2
EGRESS-AMPLI-GAIN-HIGH = 12
HIGH-TX-BR-PWR = 0
HIGH-RX-BR-PWR = 0
SPAN-TOO-SHORT-TX = 0
SPAN-TOO-SHORT-RX = 0

Parameter Statistics:
-----
Total Tx Power = 8.08 dBm
Rx Signal Power = -50.00 dBm
Tx Signal Power = 5.61 dBm
Egress Ampli Gain = 28.9 dB
Egress Ampli Safety Control mode = auto
Egress Ampli OSRI = OFF
Egress Ampli Force APR = OFF

Configured Parameters:
-----
Egress Ampli Gain = 23.0 dB
Egress Ampli Power = 3.0 dBm
Egress Ampli Safety Control mode = auto
Egress Ampli OSRI = OFF
Egress Ampli Force APR = OFF
Ampli Control mode = Manual
Rx Low Threshold = -30.0 dBm

```

```
Tx Low Threshold = -5.0 dBm
```

```
Temperature = 23.00 Celsius
Voltage = 3.36 V
```

Optical Module Details

```
Optics type           : QDD DUAL EDFA
Name                  : CISCO-ACCELINK
OUI Number            : 00.00.00
Part Number           : EDFA-211917-QDD
Rev Number            : 26
Serial Number         : ACW2650Z004
PID                   : ONS-QDD-OLS
VID                   : ves1
Firmware Version      : 2.06
Date Code (yy/mm/dd) : 22/12/31
Fiber Connector Type  : CS
```

Configuring Force APR

Automatic Power Reduction (APR) is an eye-safe output power level of the OLS pluggable when you restart the pluggable (amplifier).

If the OLS pluggable is in the APR state, then the default value for APR power is 8 dBm and APR timer is 9 seconds. When **Force APR** is configured or enabled, the OLS pluggable continues to remain in the APR state. If force APR is not configured or disabled, then the OLS pluggable remains in the Gain mode or Power mode, based on the **ampli-control-mode** user configuration.

Force APR Configuration Example

The following example shows how to configure force APR on the OLS pluggable :

```
Router#config
Router(config)#controller ots 0/0/2/1/1
Router(config-Ots)#egress-ampli-force-apr on
Router(config-Ots)#commit
Router(config-Ots)#exit
Router(config)#exit
```

Running Configuration

This example shows the running configuration for the OLS pluggable:

```
Router#show run controller ots 0/0/2/1/1
controller Ots0/0/2/1/1
  egress-ampli-force-apr on
  ampli-control-mode manual
  egress-ampli-gain 230
  egress-ampli-force-apr on
  egress-ampli-safety-control-mode manual
!
```

Verification

This example shows how to verify the force APR configured:

```
Router#show controllers ots 0/0/2/1/1
```

```
Mon Jan  1 12:37:17.536 UTC
```

```

Controller State: Up
Transport Admin State: In Service
LED State: Green

```

```

Alarm Status:
-----
Detected Alarms: None

```

```

Alarm Statistics:
-----
RX-LOS-P = 1
RX-LOC = 0
TX-POWER-FAIL-LOW = 0
INGRESS-AUTO-LASER-SHUT = 0
INGRESS-AUTO-POW-RED = 0
INGRESS-AMPLI-GAIN-LOW = 0
INGRESS-AMPLI-GAIN-HIGH = 0
EGRESS-AUTO-LASER-SHUT = 0
EGRESS-AUTO-POW-RED = 0
EGRESS-AMPLI-GAIN-LOW = 5
EGRESS-AMPLI-GAIN-HIGH = 0
HIGH-TX-BR-PWR = 0
HIGH-RX-BR-PWR = 0
SPAN-TOO-SHORT-TX = 0
SPAN-TOO-SHORT-RX = 0

```

```

Parameter Statistics:
-----
Total Tx Power = 17.49 dBm
Rx Signal Power = -19.91 dBm
Tx Signal Power = 17.47 dBm
Egress Ampli Gain = 17.0 dB
Egress Ampli OSRI = OFF
Amplifier Control Mode = Manual
Egress Ampli Force APR = ON

```

```

Configured Parameters:
-----
Egress Ampli Gain = 20.0 dB
Egress Ampli Power = 8.0 dBm
Egress Ampli OSRI = OFF
Egress Ampli Force APR = OFF
Ampli Control mode = Manual
Egress Ampli Force APR = ON
Rx Low Threshold = -30.0 dBm
Tx Low Threshold = -5.0 dBm

```

```

Temperature = 33.40 Celsius
Voltage = 3.25 V

```

Optical Module Details

```

Optics type           : QDD DUAL EDFA
Name                  : CISCO-II-VI
OUI Number            : 00.90.65
Part Number           : 60P310001
Rev Number            : 01
Serial Number         : IIF26410004
PID                   : ONS-QDD-OLS
VID                   : VES1

```

```

Firmware Version      : 1.00
Date Code (yy/mm/dd) : 22/10/10
Fiber Connector Type  : CS

```

Use Case for QDD OLS pluggable

The QDD OLS pluggable can transport 8 or 16 coherent optical channels from the DWDM optical modules that are plugged into the router.

The optical modules are interconnected with the QDD OLS amplifiers using the following cables:

- ONS-BRK-CS-8LC: dual fanout 1x8 cable-assembly with embedded passive splitter and coupler
- ONS-BRK-CS-16LC: dual fanout 1x16 cable-assembly with embedded passive splitter and coupler
- ONS-CAB-CS-LC-5: dual adapter patch-cord CS-connector to LC-connector

The following section explains the 8-channel Optical Line System (OLS) that is achieved by using the QDD OLS pluggable and QDD-400G-ZRP-S modules. With this 8-channel Optical Line System (OLS) set-up it's now possible to obtain 28 dB/112 kilometer span reach. Also, the fiber bandwidth is increased by 8 times.

8-Channel Optical Line System

The following section explains the 8-channel Optical Line System (OLS) that is achieved by using the QDD OLS pluggable and QDD-400G-ZR-S or QDD-400G-ZRP-S module. With this 8-channel Optical Line System (OLS) set up it's now possible to obtain 28 dB/112 kilometer span reach. Also, the fiber bandwidth is increased by 8 times.

This section explains the 8-channel optical line system (OLS) that is achieved by using the following:

- Four Cisco 8000 series router (represented as Node A, Node B, Node C, and Node D)
- Sixteen QDD-400G-ZR-S or QDD-400G-ZRP-S modules
- Two QDD OLS (ONS-QDD-OLS) pluggables
- Two ONS-BRK-CS-8LC breakout cables
- Two ONS-CAB-CS-LC-5 fiber optic cable

Connections on Node A

Four QDD-400G-ZR-S or QDD-400G-ZRP-S modules are inserted into port 0, port 1, port 2, and port 3 of the Cisco 8000 series router. The QDD OLS (ONS-QDD-OLS) pluggable is inserted into port 4 of the Cisco 8000 series router.

Connections on Node B

Four QDD-400G-ZR-S or QDD-400G-ZRP-S modules are inserted into port 0, port 1, port 2, and port 3 of the Cisco 8000 series router.

Connections between Node A and Node B

Using the ONS-BRK-CS-8LC breakout cable connect eight QDD-400G-ZR-S or QDD-400G-ZRP-S modules (four each on Node A and Node B) and the QDD OLS (ONS-QDD-OLS) pluggable (port 3 of Node A).

Connections on Node C

Four QDD-400G-ZR-S or QDD-400G-ZRP-S modules are inserted into port 0, port 1, port 2, and port 3 of the Cisco 8000 series router.

Connections on Node D

Four QDD-400G-ZR-S or QDD-400G-ZRP-S modules are inserted into port 0, port 1, port 2, and port 3 of the Cisco 8000 series router. The QDD OLS (ONS-QDD-OLS) pluggable is inserted into port 4 of the Cisco 8000 series router.

Connections between Node C and Node D

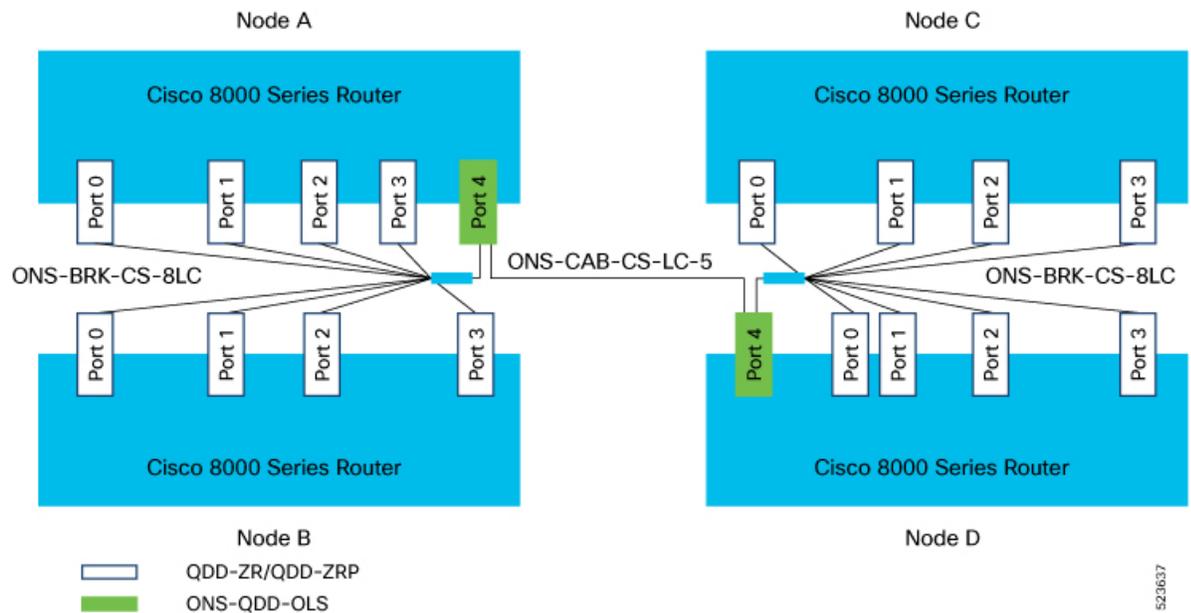
Using the ONS-BRK-CS-8LC breakout cable connect eight QDD-400G-ZR-S or QDD-400G-ZRP-S modules (four each on Node A and Node B) and the QDD OLS (ONS-QDD-OLS) pluggable (port 3 of Node A).

Connections between Node A and Node D

Using the ONS-CAB-CS-LC-5 fiber optic cable connect both the QDD OLS (ONS-QDD-OLS) pluggables that are present in Node A and Node D.

The representation of these interconnections between Node A/Node B and Node C/Node D are depicted in the block diagram below:

Figure 23: 8-Channel Optical Line System



OLS Alarms Troubleshooting

This section contains the procedures for troubleshooting alarms.

RX-LOS-P

Default Severity: Critical

Logical Object: Controller

The RX-LOS-P alarm is raised when there is loss of signal.

Clear the RX-LOS-P Alarm

1. Verify the transmission (TX) at the peer end.
2. Check the fiber connections.

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).

RX-POWER-FAIL-LOW

Default Severity: Minor (MN), Non-Service-Affecting (NSA)

Logical Object: Controller

The RX-POWER-FAIL-LOW alarm is raised when the RX power is below the configured low threshold values.

Clear the RX-POWER-FAIL-LOW Alarm

1. Verify the transmission (TX) at the peer end.
2. Check the fiber connections.
3. Increase the peer end gain or transmit-power value to obtain the RX power above the 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).

TX-POWER-FAIL-LOW

Default Severity: Critical

Logical Object: Controller

The TX-POWER-FAIL-LOW alarm is raised when the TX power is below the configured low threshold values.

Clear the TX-POWER-FAIL-LOW Alarm

1. Increase the gain or power configuration value to obtain the TX power above the 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).

EGRESS-AMPLI-GAIN-LOW

Default Severity: Minor (MN), Non-Service-Affecting (NSA)

Logical Object: Controller

The EGRESS-AMPLI-GAIN-LOW alarm is raised when the actual gain of the OLS pluggable is lower than the configured gain value.

Clear the EGRESS-AMPLI-GAIN-LOW Alarm

1. Configure the gain value within the optimum 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).

EGRESS-AMPLI-GAIN-HIGH

Default Severity: Minor (MN), Non-Service-Affecting (NSA)

Logical Object: Controller

The EGRESS-AMPLI-GAIN-HIGH alarm is raised when the actual gain of the OLS pluggable is higher than the configured gain value.

Clear the EGRESS-AMPLI-GAIN-HIGH Alarm

1. Verify the RX and TX values and adjust the gain within the optimum working 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).

EGRESS-AUTO-LASER-SHUT

Default Severity: Not-Alarmed

Logical Object: Controller

The EGRESS-AUTO-LASER-SHUT alarm is raised when there is loss of signal (LOS) on the OTS line side (subport 1)

Clear the EGRESS-AUTO-LASER-SHUT Alarm

1. Verify the fiber connections on the line side of the OLS pluggable.
2. Verify the gain or power on the line side of the peer end.

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).

EGRESS-AUTO-POW-RED

Default Severity: Not-Alarmed

Logical Object: Controller

The EGRESS-AUTO-POW-RED alarm is raised when there is loss of signal (LOS) on the OTS line side (subport 1)

Clear the EGRESS-AUTO-POW-RED Alarm

1. Verify the fiber connections on the line side of the OLS pluggable.
2. Verify the gain or power on the line side of the peer end.

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 19

Managing Router Hardware

This chapter describes the concepts and tasks used to manage and configure the hardware components of a router running the Cisco IOS XR software.

This module contains the following topics:

- [MPA Reload, on page 439](#)
- [RP Redundancy and Switchover, on page 440](#)
- [NPU Power Optimization, on page 445](#)
- [Dynamic Power Management, on page 449](#)
- [Ability to Set Maximum Power Limit for the Router , on page 466](#)
- [Configuring the Compatibility Mode for Various NPU Types, on page 468](#)
- [Storage Media Sanitization, on page 473](#)
- [Excluding Sensitive Information in Show Running Configurations Output, on page 481](#)
- [Fabric Link Management for Uncorrectable Errors, on page 483](#)
- [Fault recovery handling, on page 487](#)
- [Periodic syslog messages for shutdowns due to fault-recovery failures, on page 490](#)
- [Machine check error notifications, on page 491](#)
- [Guidelines for Online Insertion and Removal on Cisco 8700 Series routers, on page 494](#)

MPA Reload

A Modular Port Adapter (MPA) is a hardware component used in networking equipment, such as routers and switches, to provide flexible and scalable port configurations.

A data path power-on timer is used during the power-on sequence of a network device to manage the initialization, stabilization, and diagnostic processes of the data path components. If an MPACard doesn't come up within 20 minutes, the data path power-on timer expires, and the MPA goes for another reload to attempt recovery.



Note When a router enters an undefined state and disrupts the traffic due to the data path power-on timer expiry (timer associated with a data path has expired), reload the router using the [reload location](#) command.

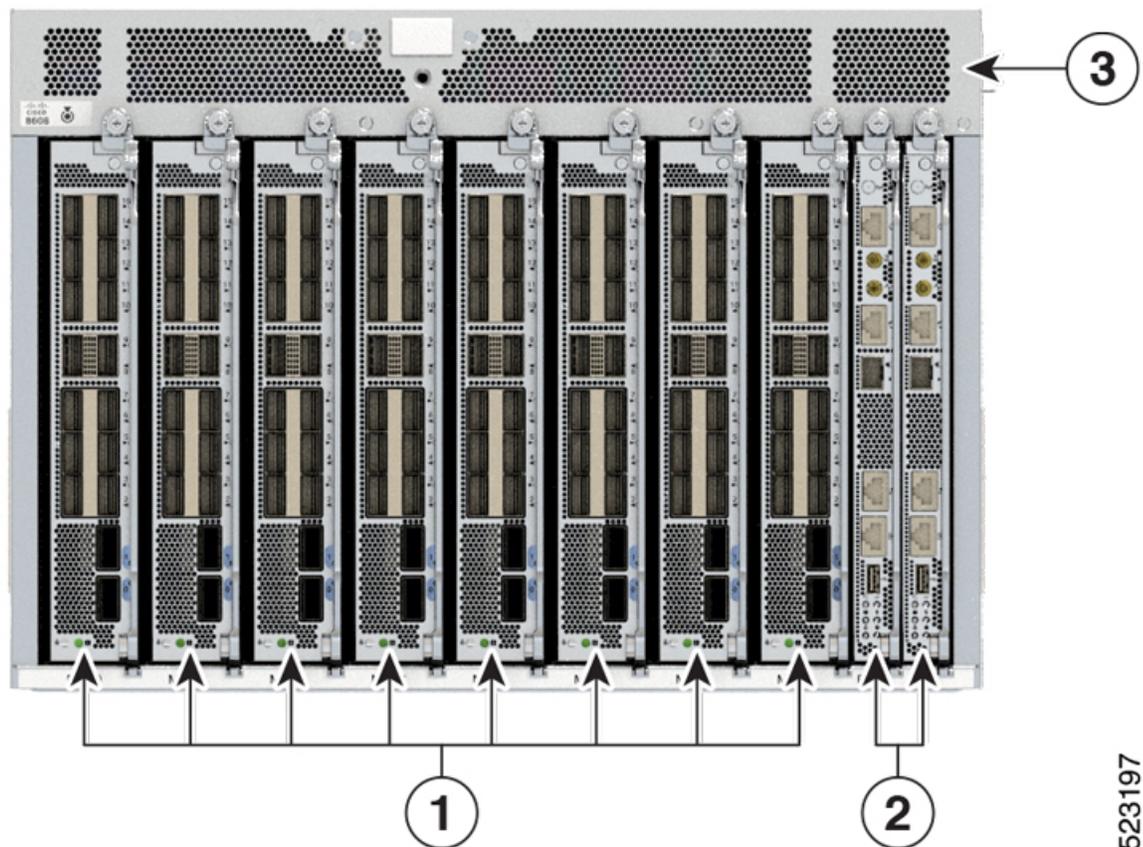
RP Redundancy and Switchover

This section describes RP redundancy and switchover commands and issues.

Establishing RP Redundancy

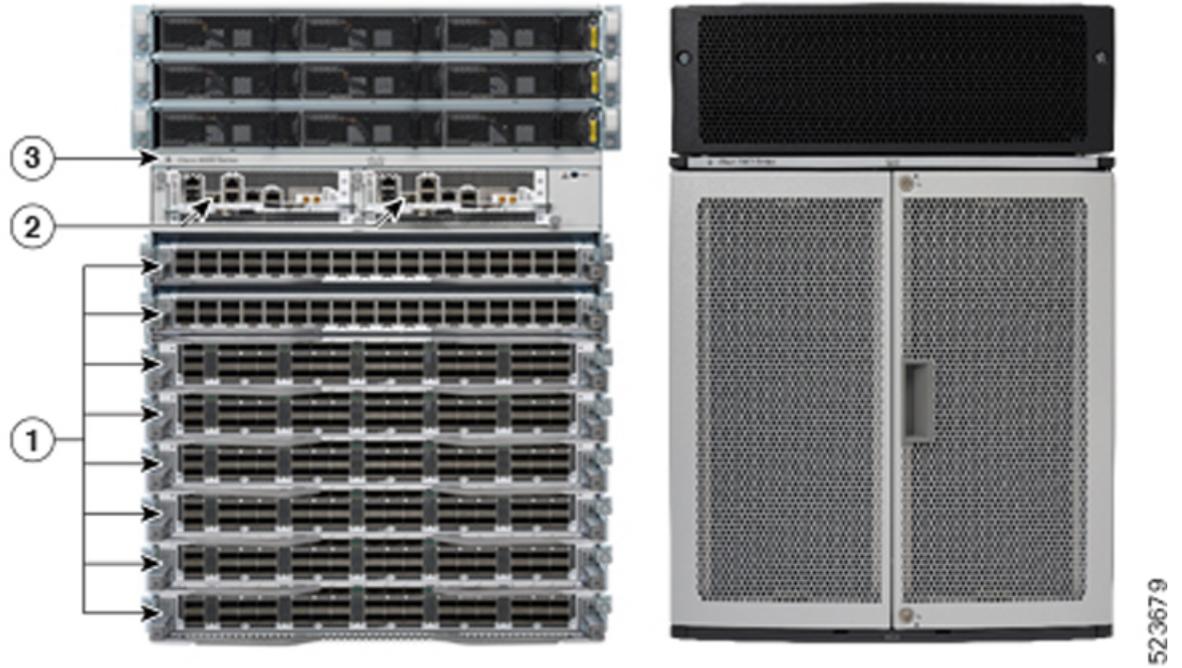
Your router has two slots for RPs: RP0 and RP1 (see [Figure 24: Redundant Set of RP Installed in Slots RP0 and RP1 in an Cisco 8608 8-Slot Centralized Chassis](#), on page 440 and [Figure 25: Redundant Set of RP Installed in Slots RP0 and RP1 in an Cisco 8808 8-Slot Distributed Chassis](#), on page 441). RP0 is the slot on the left, facing the front of the chassis, and RP1 is the slot on right. These slots are configured for redundancy by default, and the redundancy cannot be eliminated. To establish RP redundancy, install RP into both slots.

Figure 24: Redundant Set of RP Installed in Slots RP0 and RP1 in an Cisco 8608 8-Slot Centralized Chassis



523197

Figure 25: Redundant Set of RP Installed in Slots RP0 and RP1 in an Cisco 8808 8-Slot Distributed Chassis



1	Modular Port Adaptors (MPAs)
2	Route Processors (RPs)
3	Chassis

Determining the Active RP in a Redundant Pair

During system startup, one RP in each redundant pair becomes the active RP. You can tell which RP is the active RP in the following ways:

- The active RP can be identified by the green Active LED on the faceplate of the card. When the Active LED turns on, it indicates that the RP is active and when it turns off, it indicates that the RP is in standby.
- The slot of the active RP is indicated in the CLI prompt. For example:

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

In this example, the prompt indicates that you are communicating with the active RP in slot RP1.

- Enter the **show redundancy** command in EXEC mode to display a summary of the active and standby RP status. For example:

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

This node (0/RP0/CPU0) is in ACTIVE role
Partner node (0/RP1/CPU0) is in STANDBY role
Standby node in 0/RP1/CPU0 is ready
```

```

Reload and boot info
-----
RP reloaded Fri Apr  9 03:44:28 2004: 16 hours, 51 minutes ago
This node booted Fri Apr  9 06:19:05 2004: 14 hours, 16 minutes ago
Last switch-over Fri Apr  9 06:53:18 2004: 13 hours, 42 minutes ago
Standby node boot Fri Apr  9 06:54:25 2004: 13 hours, 41 minutes ago
Standby node last not ready Fri Apr  9 20:35:23 2004: 0 minutes ago
Standby node last ready Fri Apr  9 20:35:23 2004: 0 minutes ago
There have been 2 switch-overs since reload

```

Role of the Standby RP

The second RP to boot in a redundant pair automatically becomes the standby RP. While the active RP manages the system and communicates with the user interface, the standby RP maintains a complete backup of the software and configurations for all cards in the system. If the active RP fails or goes off line for any reason, the standby RP immediately takes control of the system.

Summary of Redundancy Commands

RP redundancy is enabled by default in the Cisco IOS XR software, but you can use the commands described in [Table 91: RP Redundancy Commands, on page 442](#) to display the redundancy status of the cards or force a manual switchover.

Table 91: RP Redundancy Commands

Command	Description
show redundancy	Displays the redundancy status of the RP. This command also displays the boot and switch-over history for the RP.
redundancy switchover	Forces a manual switchover to the standby RP. This command works only if the standby RP is installed and in the “ready” state.
show platform	Displays the status for node, including the redundancy status of the RP cards. In EXEC mode, this command displays status for the nodes assigned to the SDR. In administration EXEC mode, this command displays status for all nodes in the system.

Automatic Switchover

Automatic switchover from the active RP to the standby RP occurs only if the active RP encounters a serious system error, such as the loss of a mandatory process or a hardware failure. When an automatic switchover occurs, the RPs respond as follows:

- If a standby RP is installed and “ready” for switchover, the standby RP becomes the active RP. The original active RP attempts to reboot.
- If the standby RP is not in “ready” state, then both RPs reboot. The first RP to boot successfully assumes the role of active RP.

RP Redundancy During RP Reload

The **reload** command causes the active RP to reload the Cisco IOS XR software. When an RP reload occurs, the RPs respond as follows:

- If a standby RP is installed and “ready” for switchover, the standby RP becomes the active RP. The original active RP reboots and becomes the standby RP.
- If the standby RP is not in the “ready” state, then both RPs reboot. The first RP to boot successfully assumes the role of active RP.

Manual Switchover

If a standby RP is installed and ready for switchover, you can force a manual switchover using the **redundancy switchover** command or reloading the active RP using the **reload** command.

Manual Switchover Using the Reload Command

You can force a manual switchover from the active RP to the standby RP by reloading the active RP using the **reload** command. As active RP reboots, the current standby RP becomes active RP, and rebooting RP switches to standby RP.

```
RP/0/RP0/CPU0:router# reload
RP/0/RP1/CPU0:router#
```

Manual Switchover Using the Redundancy Switchover Command

You can force a manual switchover from the active RP to the standby RP using the **redundancy switchover** command.

If a standby RP is installed and ready for switchover, the standby RP becomes the active RP. The original active RP becomes the standby RP. In the following example, partial output for a successful redundancy switchover operation is shown:

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

This node (0/RP0/CPU0) is in ACTIVE role
Partner node (0/RP1/CPU0) is in STANDBY role
Standby node in 0/RP1/CPU0 is ready

RP/0/RP0/CPU0:router# redundancy switchover
Updating Commit Database. Please wait...[OK]
Proceed with switchover 0/RP0/CPU0 -> 0/RP1/CPU0? [confirm]
Initiating switch-over.
RP/0/RP0/CPU0:router#

<Your 'TELNET' connection has terminated>
```

In the preceding example, the Telnet connection is lost when the previously active RP resets. To continue management of the router, you must connect to the newly activated RP as shown in the following example:

```
User Access Verification

Username: xxxxx
```

```

Password: xxxxx
Last switch-over Sat Apr 15 12:26:47 2009: 1 minute ago

RP/0/RP1/CPU0:router#

```

If the standby RP is not in “ready” state, the switchover operation is not allowed. In the following example, partial output for a failed redundancy switchover attempt is shown:

```

RP/0/RP0/CPU0:router# show redundancy

Redundancy information for node 0/RP1/CPU0:
=====
Node 0/RP0/CPU0 is in ACTIVE role
Partner node (0/RP1/CPU0) is in UNKNOWN role

Reload and boot info
-----
RP reloaded Wed Mar 29 17:22:08 2009: 2 weeks, 2 days, 19 hours, 14 minutes ago
Active node booted Sat Apr 15 12:27:58 2009: 8 minutes ago
Last switch-over Sat Apr 15 12:35:42 2009: 1 minute ago
There have been 4 switch-overs since reload

RP/0/RP0/CPU0:router# redundancy switchover

Switchover disallowed: Standby node is not ready.

```

Communicating with a Standby RP

The active RP automatically synchronizes all system software, settings, and configurations with the standby RP.

If you connect to the standby RP through the console port, you can view the status messages for the standby RP. The standby RP does not display a CLI prompt, so you cannot manage the standby card while it is in standby mode.

If you connect to the standby RP through the management Ethernet port, the prompt that appears is for the active RP, and you can manage the router the same as if you had connected through the management Ethernet port on the active RP.

NPU Power Optimization

Table 92: Feature History Table

Feature Name	Release Information	Description
NPU Power Optimization	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8011-4G24Y4H-I • 8712-MOD-M
NPU Power Optimization	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-36EH • 88-LC1-12TH24FH-E • 88-LC1-52Y8H-EM
NPU Power Optimization	Release 7.3.15	<p>This feature lets you choose a predefined NPU power mode based on your network's individual requirements, and consequently reducing NPU power consumption.</p> <p>The hw-module npu-power-profile command is introduced for this feature.</p>

Cisco 8000 series routers are powered by Cisco Silicon One Q200 and Q100 series processors. Cisco Silicon One processors offer high performance, flexible, and power-efficient routing silicon in the market.

NPU Power Optimization feature helps to reduce NPU power consumption by running a processor in a predefined mode. There are three NPU power modes—high, medium, and low. Based on your network traffic and power consumption requirements, you can choose to run the processor in any one of the three NPU power modes.

- High: The router will use the maximum amount of power, resulting in the best possible performance.
- Medium: The router power consumption and performance levels are both average.
- Low: The router operates with optimal energy efficiency while providing a modest level of performance.



Note We recommend that you work with your Cisco account representatives before implementing this feature in your network.

On a Q200-based Cisco 8200 series chassis, you can configure an NPU power mode on the entire router.

On a Q200-based Cisco 8800 series chassis, you can configure an NPU power mode only on fabric cards and line cards.

The following table lists the supported hardware, and their default NPU power mode:

Table 93: Supported Hardware and Default Modes

Supported Hardware	Default NPU Power Mode
Cisco 8200 32x400 GE 1RU fixed chassis (8201-32FH)	High
88-LC0-36FH without MACSec, based on Q200 Silicon Chip	Medium
88-LC0-36FH-M with MACSec, based on Q200 Silicon Chip	Medium
8808-FC0 Fabric Card, based on Q200 Silicon Chip	Low
8818-FC0 Fabric Card, based on Q200 Silicon Chip	Medium



Caution We recommend that you use the default NPU power mode on your router.

Limitations

The NPU power optimization is not supported on the Q100-based systems.

The NPU Power Profile mode is not supported on the following Q200-based line cards:

Table 94: Limitation on Hardware and Power Profile Modes

Hardware	Power Profile Mode
88-LC0-36FH-M	High
88-LC0-34H14FH	High

Configuring NPU Power Mode

Configuring NPU power mode on a fixed chassis:

The following example shows how to configure an NPU power mode on a fixed chassis:

```
RP/0/RP0/CPU0:ios(config)#hw-module npu-power-profile high
RP/0/RP0/CPU0:ios(config)#commit

RP/0/RP0/CPU0:ios(config)#reload
```



Note Note: Reload the chassis for the configurations changes to take effect.

Verifying NPU power mode configuration on a fixed chassis:

Use the **show controllers npu driver** command to verify the NPU power mode configuration:

```
RP/0/RP0/CPU0:ios#show controllers npu driver location 0/RP0/CPU0
Mon Aug 24 23:29:34.302 UTC
=====
NPU Driver Information
=====
Driver Version: 1
SDK Version: 1.32.0.1
Functional role: Active,      Rack: 8203, Type: lcc, Node: 0
Driver ready      : Yes
NPU first started : Mon Aug 24 23:07:41 2020
Fabric Mode:
NPU Power profile: High
Driver Scope: Node
Respawn count    : 1
Availablity masks :
                card: 0x1,      asic: 0x1,      exp asic: 0x1
...
```

Configuring NPU power mode on a modular chassis

The following example shows how to configure an NPU power mode on a fabric card and a line card:

```
RP/0/RP0/CPU0:ios(config)#hw-module npu-power-profile card-type FC high
RP/0/RP0/CPU0:ios(config)#hw-module npu-power-profile card-type LC low location 0/1/cpu0
RP/0/RP0/CPU0:ios(config)#commit
```



Note For the configurations to take effect, you must:

- Reload a line card if the configuration is applied on the line card.
 - Reload a router if the configuration is applied on a fabric card.
-

Verifying the NPU power mode configuration on a modular chassis

Use the **show controllers npu driver location** command to verify the NPU power mode configuration:

```
RP/0/RP0/CPU0:ios#show controllers npu driver location 0/1/CPU0

Functional role: Active,      Rack: 8808, Type: lcc, Node: 0/RP0/CPU0
Driver ready      : Yes
NPU first started : Mon Apr 12 09:57:27 2021
Fabric Mode: FABRIC/8FC
NPU Power profile: High
```

```

Driver Scope: Rack
Respawn count      : 1
Availability masks :
    card: 0xba,    asic: 0xcfcc,    exp asic: 0xcfcc
Weight distribution:
    Unicast: 80,   Multicast: 20
    
```

Process / Lib	Connection status	Registration status	Connection requests	DLL registration
FSDB	Active	Active	1	n/a
FGID	Active	Active	1	n/a
AEL	n/a	n/a	n/a	Yes
SM	n/a	n/a	n/a	Yes

```

Asics :
HP - HotPlug event, PON - Power On reset
HR - Hard Reset,   WB - Warm Boot
    
```

Asic inst. (R/S/A)	fap id	HP	Slice state	Asic type	Admin state	Oper state	Asic state	Last init	PON (#)	HR (#)	FW Rev
0/FC1/2	202	1	UP	s123	UP	UP	NRML	PON	1	0	0x0000
0/FC1/3	203	1	UP	s123	UP	UP	NRML	PON	1	0	0x0000
0/FC3/6	206	1	UP	s123	UP	UP	NRML	PON	1	0	0x0000
0/FC3/7	207	1	UP	s123	UP	UP	NRML	PON	1	0	0x0000
0/FC4/8	208	1	UP	s123	UP	UP	NRML	PON	1	0	0x0000
0/FC4/9	209	1	UP	s123	UP	UP	NRML	PON	1	0	0x0000
0/FC5/10	210	1	UP	s123	UP	UP	NRML	PON	1	0	0x0000
0/FC5/11	211	1	UP	s123	UP	UP	NRML	PON	1	0	0x0000
0/FC7/14	214	1	UP	s123	UP	UP	NRML	PON	1	0	0x0000
0/FC7/15	215	1	UP	s123	UP	UP	NRML	PON	1	0	0x0000

SI Info :

Card	Board HW Version	SI Board Version	SI Param Version	Retimer Board Version	Retimer SI Param Version	Front Panel PHY
FC1	0.22	1	6	NA	NA	NA
FC3	0.21	1	6	NA	NA	NA
FC4	0.21	1	6	NA	NA	NA
FC5	0.21	1	6	NA	NA	NA
FC7	0.21	1	6	NA	NA	NA

```

Functional role: Active, Rack: 8808, Type: lcc, Node: 0/1/CPU0
Driver ready      : Yes
NPU first started : Mon Apr 12 09:58:10 2021
Fabric Mode: FABRIC/8FC
NPU Power profile: Low
Driver Scope: Node
Respawn count      : 1
Availability masks :
    card: 0x1,    asic: 0x7,    exp asic: 0x7
Weight distribution:
    Unicast: 80,   Multicast: 20
    
```

```

+-----+
| Process | Connection | Registration | Connection | DLL |
| /Lib    | status    | status      | requests   | registration |
+-----+
| FSDB    | Active    | Active      |            | 1 | n/a |
| FGID    | Inactive  | Inactive    |            | 0 | n/a |
| AEL     | n/a      | n/a        |            | n/a | Yes |
| SM      | n/a      | n/a        |            | n/a | Yes |
+-----+
    
```

Asics :
 HP - HotPlug event, PON - Power On reset
 HR - Hard Reset, WB - Warm Boot

```

+-----+
| Asic inst. | fap|HP|Slice|Asic|Admin|Oper | Asic state | Last |PON|HR | FW |
| (R/S/A)    | id | |state|type|state|state|           | init |(#)|(#)| Rev |
+-----+
| 0/2/0      | 8 | 1 | UP  |npu | UP  | UP  |NRML      |PON | 1 | 0|0x0000|
| 0/2/1      | 9 | 1 | UP  |npu | UP  | UP  |NRML      |PON | 1 | 0|0x0000|
| 0/2/2      |10 | 1 | UP  |npu | UP  | UP  |NRML      |PON | 1 | 0|0x0000|
+-----+
    
```

SI Info :

```

+-----+
| Card | Board | SI Board | SI Param | Retimer SI | Retimer SI | Front Panel |
|      |      |          |          |            |            |             |
|      | HW Version | Version | Version | Board Version | Param Version | PHY |
+-----+
| LC2  | 0.41  | 1       | 9       | NA          | NA          | DEFAULT |
+-----+
    
```

Dynamic Power Management

Table 95: Feature History Table

Feature Name	Release Information	Description
Dynamic Power Management	Release 25.1.1	Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*) *This feature is supported on: <ul style="list-style-type: none"> • 8011-4G24Y4H-I • 8712-MOD-M

Feature Name	Release Information	Description
Dynamic Power Management	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-36EH • 88-LC1-12TH24FH-E • 88-LC1-52Y8H-EM
Dynamic Power Management	Release 7.3.15	<p>The Dynamic Power Management feature considers certain dynamic factors before allocating power to the fabric and line cards.</p> <p>This feature has the following benefits:</p> <ul style="list-style-type: none"> • Reduces number of PSUs required by accurately representing the maximum power consumption • Improves PSU efficiency by providing more accurate power allocation <p>This feature thus optimizes power allocation and avoids overprovisioning power to a router.</p>
Dynamic Power Management	Release 7.3.2	<p>Previously available for fabric and line cards, this feature that helps avoid excess power allocation by considering dynamic factors before allocating power to them is now available for optical modules.</p> <p>To view the power allocation on a per port basis, a new command "show environment power allocated [details]" is introduced.</p>
Dynamic Power Management	Release 7.3.3	<p>The Dynamic Power Management feature is now supported on the following Cisco 8100 and 8200 series routers:</p> <ul style="list-style-type: none"> • Cisco 8201 • Cisco 8202 • Cisco 8201-32-FH • Cisco 8101-32-FH

Feature Name	Release Information	Description
Dynamic Power Management	Release 7.5.2	The Cisco 8202-32FH-M router will now consider dynamic factors, such as optical modules, NPU power profile, and MACsec mode to enable improved power allocation and utilization.

Prior to Cisco IOS XR Release 7.3.15, when Cisco 8000 series routers were powered on or reloaded, the power management feature reserved power to fabric cards and allocated maximum power to line cards. The power management feature wouldn't consider dynamic factors, such as the type of fabric or line cards in the chassis, or whether a fabric or line card was present in a slot.

The Dynamic Power Management feature considers such dynamic factors before allocating power to the fabric and line cards.

This feature has the following benefits:

- Reduces number of PSUs required by accurately representing the maximum power consumption
- Improves PSU efficiency by providing more accurate power allocation

This feature thus optimizes power allocation and avoids overprovisioning power to a router.

This feature is supported on the following Cisco 8000 series routers:

- Cisco 8804, 8808, 8812, and 8818 routers
- Cisco 8201, 8202, 8201-32-FH, and 8202-32FH-M routers
- Cisco 8101-32-FH

By default, this feature is enabled on the router.

The Dynamic Power Management feature allocates the total power to a router and its fabric card or line card based on the following parameters:

- Number and type of fabric cards installed on the router
- Fabric cards operating modes (5FC or 8FC)
- Number and type of line cards installed on the router
- Combination of line card and fabric card types installed
- NPU power mode configured on a fabric card
- Number and type of optics installed (supported in Cisco IOS XR Software Release 7.3.2 and later)
- MACSec-enabled ports (supported from Cisco IOS XR Software Release 7.3.3 and later)

For details, see *Dynamic Power Management for MACSec-Enabled Ports* section in the *Configuring MACSec* chapter in the *System Security Configuration Guide for Cisco 8000 Series Routers*.

On 8202-32FH-M router, the Dynamic Power Management feature allocates the total power to a router based on the following parameters:

- Optical modules installed.

- NPU power profile. To identify the mode on which the router is operating, use the `hw-module npu-power-profile` command.
- MACSec mode. By default, MACSec mode is disabled on 8202-32FH-M router.



Note We recommend you work with your Cisco account representatives to calculate power requirements for the Cisco 8000 series router.

Power Allocation to Empty Card Slot

This feature allocates a minimum required power for all empty LC or FC slots. This minimum power is required to boot the CPU and FPGAs immediately when a card is inserted. The feature doesn't control booting up the CPU and FPGAs. Also, the minimum power is required to detect the card type before the feature decides if there's enough power to power up the data path.

For example, the following **show environment power** command output displays various LC or FC card statuses, and also shows allocated and used power.



Note The allocated power capacity shown in the following **show** command output isn't standard capacity. The allocated power capacity varies depending on various other factors.

```
Router# show environment power
Thu Apr 22 12:03:06.754 UTC
=====
CHASSIS LEVEL POWER INFO: 0
=====
Total output power capacity (N + 1)      : 9600W + 6300W
Total output power required              : 9241W
Total power input                        : 6146W
Total power output                       : 5826W
=====

Power      Supply      -----Input-----  -----Output---   Status
Module     Type                Volts A/B  Amps A/B  Volts  Amps
=====
0/PT0-PM0  PSU6.3KW-HV         245.5/245.7  5.1/5.0   54.7   43.1   OK
0/PT0-PM1  PSU6.3KW-HV         0.0/245.2   0.0/7.4   54.3   31.7   OK
0/PT0-PM2  PSU6.3KW-HV         0.0/246.9   0.0/7.5   54.1   32.3   OK

Total of Power Modules:                6146W/25.0A                5826W/107.1A
=====

Location   Card Type                Power      Power      Status
           Type                Allocated  Used
           Type                Watts      Watts
=====
0/RP0/CPU0  8800-RP                  95         69         ON
0/RP1/CPU0  -                          95         -          RESERVED
0/0/CPU0    88-LC0-36FH           796      430      ON
0/1/CPU0    -                          102        -          RESERVED
0/2/CPU0    88-LC0-36FH             796        430       ON
0/3/CPU0    -                          102     -          RESERVED
0/4/CPU0    -                          102        -          RESERVED
0/5/CPU0    -                          102        -          RESERVED
```

0/6/CPU0	-	102	-	RESERVED
0/7/CPU0	-	102	-	RESERVED
0/8/CPU0	-	102	-	RESERVED
0/9/CPU0	88-LC0-36FH	102	-	OFF
0/10/CPU0	-	102	-	RESERVED
0/11/CPU0	-	102	-	RESERVED
0/FC0	-	26	-	RESERVED
0/FC1	-	26	-	RESERVED
0/FC2	-	26	-	RESERVED
0/FC3	8812-FC	784	509	ON
0/FC4	8812-FC	784	503	ON
0/FC5	8812-FC	26	-	OFF
0/FC6	8812-FC	26	-	OFF
0/FC7	8812-FC	26	-	OFF
0/FT0	8812-FAN	1072	1000	ON
0/FT1	8812-FAN	1072	1012	ON
0/FT2	8812-FAN	1072	861	ON
0/FT3	8812-FAN	1072	1033	ON

This table describes the card slot statuses:

Table 96: Router Card Slot Status

Status	Description
RESERVED	When a slot is empty
OFF	When a card is inserted in a slot but power isn't allocated to the card
ON	When a card is allocated power and the card is in operational state

Low-Power Condition

When you insert an LC or FC in a card slot at the time when the router doesn't have enough power available to allocate to the new card, the dynamic power management feature doesn't provision power to the card. It raises the *ev_power_budget_not_ok* alarm, and gracefully shuts down the card.

In the following **show** command output, an FC inserted in the card slot location 0/FC6 is gracefully shut down due to lack of power:

```
Router# show shelfmgr history events location 0/FC6
Thu Apr 22 12:03:11.763 UTC
NODE NAME      : 0/FC6
CURRENT STATE  : CARD_SHUT_POWERED_OFF
TIME STAMP     : Apr 20 2021 16:49:52
-----
DATE           TIME (UTC)  EVENT                               STATE
-----
Apr 20 2021 16:49:52  ev_powered_off                       CARD_SHUT_POWERED_OFF
Apr 20 2021 16:49:52  ev_device_offline                     STATE_NOT_CHANGED
Apr 20 2021 16:49:52  ev_unmapped_event                     STATE_NOT_CHANGED
Apr 20 2021 16:49:48  transient_condition                   CARD_SHUTDOWN
Apr 20 2021 16:49:48  ev_check_card_down_reaso              CHECKING_DOWN_REASON
Apr 20 2021 16:49:48  ev_timer_expiry                       CARD_SHUTDOWN_IN_PROGRESS
Apr 20 2021 16:48:46  ev_power_budget_not_ok                CARD_SHUTDOWN_IN_PROGRESS
Apr 20 2021 16:48:45  transient_condition                   POWER_BUDGET_CHECK
Apr 20 2021 16:48:45  ev_fpd_upgrade_not_reqd              CARD_STATUS_CHECK_COMPLETE
Apr 20 2021 16:47:45  ev_card_status_check                  CARD_STATUS_CHECK
```

```

Apr 20 2021 16:47:45   ev_card_info_rcvd      CARD_INFO_RCVD
Apr 20 2021 16:47:44   ev_device_online       DEVICE_ONLINE
Apr 20 2021 16:47:43   ev_timer_expiry        CARD_POWERED_ON
Apr 20 2021 16:47:33   ev_powered_on          CARD_POWERED_ON
Apr 20 2021 16:47:33   init                   CARD_DISCOVERED
-----

```

However, after an LC, FC, or chassis reload, the dynamic power management feature can't ensure that the same LCs, FCs, optics, or interfaces, which were operational earlier (before the reload), would become active again.



Note During a low-power condition, this feature doesn't borrow power from a redundant power supply.

Power Allocation to Optics

From Cisco IOS XR Release 7.3.2 onwards, power requirement for optics is also considered before allocating power to them.

To identify the power allocated for a particular interface, use the **show environment power allocated [details] location location** command.

When the optical modules are inserted, power is automatically allocated for that interface. If power has been allocated to the interface, then use the “**no shut**” command to enable the interface.

```

Router# show environment power allocated location 0/3/CPU0
Thu Oct 7 22:27:35.732 UTC
-----

```

Location	Components	Power Allocated Watts
0/3/CPU0	Data-path	772
	OPTICS	138
Total		910

```

Router# show environment power allocated details location 0/3/CPU0
Thu Oct 7 22:27:42.221 UTC
-----

```

Location	Components	Power Allocated Watts
0/3/CPU0	Data-path	772
	0/3/0/0	3
	0/3/0/1	3
	0/3/0/2	3
	0/3/0/3	3
	0/3/0/4	3
	0/3/0/5	3
	0/3/0/6	3
	0/3/0/7	3
	0/3/0/8	3
	0/3/0/9	3
	0/3/0/10	3
	0/3/0/11	3
0/3/0/12	3	

```

0/3/0/13          3
0/3/0/14          3
0/3/0/15          3
0/3/0/16          3
0/3/0/17          3
0/3/0/18          3
0/3/0/19          3
0/3/0/20          3
0/3/0/21          3
0/3/0/22          3
0/3/0/23          3
0/3/0/24          3
0/3/0/25          3
0/3/0/26          3
0/3/0/27          3
0/3/0/28          3
0/3/0/29          3
0/3/0/30          3
0/3/0/31          3
0/3/0/32          3
0/3/0/33          3
0/3/0/34          3
0/3/0/35          3
0/3/0/36          3
0/3/0/37          3
0/3/0/38          3
0/3/0/39          3
0/3/0/40          3
0/3/0/41          3
0/3/0/42          3
0/3/0/43          3
0/3/0/44          3
0/3/0/46          3

```

```

=====
Total                910

```

When the power is not allocated to the interface, the following syslog error and alarms are displayed

```

!<--Syslog Error-->!
#LC/0/3/CPU0:Oct  7 22:46:48.114 UTC: optics_driver[165]: %PKT_INFRA-FM-3-FAULT_MAJOR :
ALARM_MAJOR :POWER ALLOCATION FAIL :DECLARE :0/3/CPU0: Optics0/3/0/44
LC/0/3/CPU0:Oct  7 22:46:48.114 UTC: optics_driver[165]:
%L2-OPTICS-2-QSFP_POWER_ALLOCATION_FAILURE : Not enough power available to enable Optics
0/3/0/44

```

```

!<--Alarm-->!
Router#show alarms brief system active
Thu Oct  7 22:47:19.569 UTC

```

```

-----
Active Alarms
-----

```

Location	Severity	Group	Set Time	Description
----------	----------	-------	----------	-------------

0/3/CPU0	Major	Software	10/07/2021 22:46:48 UTC	Optics0/3/0/44 - hw_optics: Lack of available power to enable the optical module
0/3/CPU0	Major	Software	10/07/2021 22:47:06 UTC	Optics0/3/0/46 - hw_optics: Lack of available power to enable the optical module

If power is not allocated to an interface and you attempt to enable that interface using the “**no shut**” command, the following syslog error is displayed:

```
LC/0/2/CPU0:Aug 30 18:01:14.930 UTC: eth_intf_ea[262]: %PLATFORM-VEEA-1-PORT_NOT_ENABLED :
  Power not allocated to enable the interface HundredGigE0_2_0_6.
```

Power Allocation to Fixed-Port Routers

The following **show environment power** command output displays power information for fixed-port routers and components.

```
Router# show environment power
Wed Feb 16 21:05:10.001 UTC
=====
CHASSIS LEVEL POWER INFO: 0
=====
  Total output power capacity (Group 0 + Group 1) :    1400W +    1400W
  Total output power required                      :    1033W
  Total power input                               :     390W
  Total power output                              :     255W

Power Group 0:
=====
  Power      Supply      -----Input-----  -----Output---  Status
  Module     Type                Volts    Amps    Volts    Amps
=====
  0/PM0      PSU1.4KW-ACPE        244.5    0.8    12.0    11.1    OK

Total of Group 0:                195W/0.8A                133W/11.1A

Power Group 1:
=====
  Power      Supply      -----Input-----  -----Output---  Status
  Module     Type                Volts    Amps    Volts    Amps
=====
  0/PM1      PSU1.4KW-ACPE        244.2    0.8    12.0    10.2    OK

Total of Group 1:                195W/0.8A                122W/10.2A

=====
  Location    Card Type                Power      Power      Status
                        Allocated  Used
                        Watts      Watts
=====
  0/RP0/CPU0  8201                    893        -          ON
  0/FT0       FAN-1RU-PE              28         -          ON
  0/FT1       FAN-1RU-PE              28         -          ON
  0/FT2       FAN-1RU-PE              28         -          ON
  0/FT3       FAN-1RU-PE              28         -          ON
  0/FT4       FAN-1RU-PE              28         -          ON
```

To identify the power allocated for a particular interface, use the **show environment power allocated [details] location location** command.

```
Router# show environment power allocated location 0/RP0/CPU0
Wed Feb 16 21:05:21.360 UTC
=====
  Location    Components                Power
                        Allocated
                        Watts
=====
  0/RP0/CPU0  Data-path                858
                OPTICS                    35
```

```

=====
Total                               893

Router# show environment power allocated details location 0/RP0/CPU0
Wed Feb 16 21:05:36.142 UTC
=====
Location      Components                Power
              Components                Allocated
              Components                Watts
=====
0/RP0/CPU0    Data-path                 858
              0/0/0/19                  21
              0/0/0/18                  14
=====
Total                               893

```

Disabling Dynamic Power Management

By default, the dynamic power management is enabled on a router. The following example shows how to disable dynamic power management:

```

RP/0/RP0/CPU0:ios(config)#power-mgmt action disable
RP/0/RP0/CPU0:ios(config)#commit

```



Caution After disabling the dynamic power management feature, you must manage the router power on your own. So, use this command with caution.



Note To reenable dynamic power management, use the **no power-mgmt action disable** command.

On-demand transfer of Redundant Power Modules to Power Reservation Pool

Table 97: Feature History Table

Feature Name	Release Information	Feature Description
On-demand transfer of Redundant Power Modules to Power Reservation Pool	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8712-MOD-M • 8011-4G24Y4H-I

Feature Name	Release Information	Feature Description
On-demand transfer of Redundant Power Modules to Power Reservation Pool	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-12TH24FH-E • 88-LC1-36EH • 88-LC1-52Y8H-EM
On-demand transfer of Redundant Power Modules to Power Reservation Pool	Release 7.11.1	<p>The Cisco 8800 Series Modular Routers now have a functionality that allows them to transfer their redundant Power Supply Units (PSUs) to the power reservation pool when there is inadequate power supply. This capability helps prevent the router from shutting down hardware components due to a lack of power in the reservation pool, which used to occur due to the router prioritizing redundancy over power availability in the power reservation pool. Consequently, the router now raises an alarm indicating redundancy loss when it transfers PSUs to the power reservation pool. This feature ensures that the router components reserve the necessary power, even when redundancy is enabled.</p>

The Cisco 8000 Series Modular Routers offer redundancy while managing Power Supply Units (PSUs), providing continuous operation if there is PSU failure. By default, the router operates in N+1 redundancy, where N represents the number of PSUs allotted to the power reservation pool for powering the router components, and 1 indicates the backup PSU. You can use the `power-mgmt redundancy-num-pms number` command in XR Config mode mode to configure the PSU redundancy from N+1 to N+x, where x is the number of redundant PSUs required. The total number of functioning PSUs must be at least x more than the number of PSUs required to support the power demanded by all the components in the system for optimal router functionality. The range of values assigned to x is 0–11, where 0 implies no power redundancy. The router uses the redundant PSUs only when there is a PSU failure. But, if the power requirement of the router

increases than the available power offered by PSUs, the router prioritizes maintaining PSU redundancy overpowering the components.

Starting from Cisco IOS XR Release 7.11.1, the Cisco 8800 Modular Routers prioritize powering the router components over preserving redundancy. The router transfers the redundant PSUs to a power reservation pool to power the router components on demand. The router utilizes the redundant PSUs to increase the power capacity in the power reservation pool rather than maintaining redundancy. For example, consider a scenario with 18900W (3 6300W PSUs) available power. Initially, the router reserves 12600W (using 2 PSUs) in the power reservation pool and retains 6300W (one PSU) as a backup to maintain N+1 redundancy. Suppose the router needs to reserve power for any components to power up and needs more power than is available in the reservation pool. In that case, the router uses the entire 18900W with all three PSUs to power the components by transferring the redundant PSU to the power reservation pool. The router then triggers a redundancy loss alarm with such an assignment. However, if any further actions result in reduced power consumption in the router, the system automatically restores redundancy and clears the redundancy lost alarm.

On redundancy loss, the router raises a **Critical** severity **Power Module redundancy lost** alarm. You can use the **show alarms brief** command to view the redundancy lost alarm.

Syslog messages for transforming redundant PSU into borrowable resource:

Syslog message created while redundancy loss (transforming redundant PSU to functional PSU):

```
RP/0/RP0/CPU0:Jul 24 11:49:01.316 UTC: envmon[214]: %PKT_INFRA-FM-3-FAULT_MAJOR : ALARM_MAJOR
:Power Module redundancy lost :DECLARE :0:
```

Syslog message created while restoring redundancy:

```
RP/0/RP0/CPU0:Jul 24 11:49:11.375 UTC: envmon[214]: %PKT_INFRA-FM-3-FAULT_MAJOR : ALARM_MAJOR
:Power Module redundancy lost :CLEAR :0:
```

You can also use the **show environment** view the redundancy status of the PSUs in the router.

The following section details the commands to verify the redundancy status in the router:

Router with N+1 redundancy:

```
Router:ios# show environment power
```

```
=====
CHASSIS LEVEL POWER INFO: 0
=====

Total output power capacity (N + 1)      : 12600W +    6300W
Total output power required              : 11545W
Total power input                        : 3302W
Total power output                       : 3004W

=====

Power      Supply      -----Input-----  -----Output---  Status
Module     Type                Volts A/B  Amps A/B  Volts  Amps
=====

0/PT5-PM0  PSU6.3KW-HV         240.5/241.3  2.2/2.4   55.1   18.3   OK
0/PT5-PM1  PSU6.3KW-HV         240.5/240.8  2.1/2.3   54.8   17.3   OK
0/PT5-PM2  PSU6.3KW-HV         242.2/241.1  2.3/2.4   54.9   19.1   OK

Total of Power Modules:                3302W/13.7A                3004W/54.7A

=====

Location   Card Type                Power      Power      Status
Allocated  Used

=====
```

Watts	Watts			
0/RP0/CPU0	8800-RP	105	78	ON
0/RP1/CPU0	-	105	-	RESERVED
0/0/CPU0	8800-LC-36FH	1097	513	ON
0/1/CPU0	-	102	-	RESERVED
0/2/CPU0	88-LC0-36FH	102	0	OFF
0/3/CPU0	-	102	-	RESERVED
0/4/CPU0	-	102	-	RESERVED
0/5/CPU0	-	102	-	RESERVED
0/6/CPU0	-	102	-	RESERVED
0/7/CPU0	-	102	-	RESERVED
0/8/CPU0	-	102	-	RESERVED
0/9/CPU0	-	102	-	RESERVED
0/10/CPU0	-	102	-	RESERVED
0/11/CPU0	-	102	-	RESERVED
0/12/CPU0	-	102	-	RESERVED
0/13/CPU0	-	102	-	RESERVED
0/14/CPU0	-	102	-	RESERVED
0/15/CPU0	-	102	-	RESERVED
0/16/CPU0	-	102	-	RESERVED
0/17/CPU0	-	102	-	RESERVED
0/FC0	-	32	-	RESERVED
0/FC1	-	32	-	RESERVED
0/FC2	8818-FC0	584	475	ON
0/FC3	-	32	-	RESERVED
0/FC4	8818-FC0	584	472	ON
0/FC5	-	32	-	RESERVED
0/FC6	-	32	-	RESERVED
0/FC7	-	32	-	RESERVED
0/FT0	8818-FAN	1786	237	ON
0/FT1	8818-FAN	1786	228	ON
0/FT2	8818-FAN	1786	234	ON
0/FT3	8818-FAN	1786	228	ON

Router with redundancy loss:

Router:ios# **sh env power**

```

=====
CHASSIS LEVEL POWER INFO: 0
=====
Total output power capacity (N + 1)      : 18900W +      OW
Total output power required              : 12689W
Total power input                        : 3302W
Total power output                       : 3004W
=====

```

Power Module	Supply Type	-----Input----- Volts A/B	Amps A/B	-----Output----- Volts	Amps	Status
0/PT5-PM0	PSU6.3KW-HV	240.5/241.3	2.2/2.4	55.1	18.3	OK
0/PT5-PM1	PSU6.3KW-HV	240.5/240.8	2.1/2.3	54.8	17.3	OK
0/PT5-PM2	PSU6.3KW-HV	242.2/241.1	2.3/2.4	54.9	19.1	OK
Total of Power Modules:		3302W/13.7A		3004W/54.7A		

Location Allocated	Card Type Used	Power	Power	Status

	Watts	Watts			
0/RP0/CPU0	8800-RP	105	78	ON	
0/RP1/CPU0	-	105	-	RESERVED	
0/0/CPU0	8800-LC-36FH	1097	513	ON	
0/1/CPU0	-	102	-	RESERVED	
0/2/CPU0	88-LC0-36FH	916	510	ON	
0/3/CPU0	-	102	-	RESERVED	
0/4/CPU0	-	102	-	RESERVED	
0/5/CPU0	-	102	-	RESERVED	
0/6/CPU0	-	102	-	RESERVED	
0/7/CPU0	-	102	-	RESERVED	
0/8/CPU0	-	102	-	RESERVED	
0/9/CPU0	-	102	-	RESERVED	
0/10/CPU0	-	102	-	RESERVED	
0/11/CPU0	-	102	-	RESERVED	
0/12/CPU0	-	102	-	RESERVED	
0/13/CPU0	-	102	-	RESERVED	
0/14/CPU0	-	102	-	RESERVED	
0/15/CPU0	-	102	-	RESERVED	
0/16/CPU0	-	102	-	RESERVED	
0/17/CPU0	-	102	-	RESERVED	
0/FC0	-	32	-	RESERVED	
0/FC1	-	32	-	RESERVED	
0/FC2	8818-FC0	749	475	ON	
0/FC3	-	32	-	RESERVED	
0/FC4	8818-FC0	749	472	ON	
0/FC5	-	32	-	RESERVED	
0/FC6	-	32	-	RESERVED	
0/FC7	-	32	-	RESERVED	
0/FT0	8818-FAN	1786	237	ON	
0/FT1	8818-FAN	1786	225	ON	
0/FT2	8818-FAN	1786	234	ON	
0/FT3	8818-FAN	1786	228	ON	

Router:ios# sh alarms brief system active

Active Alarms

Description	Location	Severity	Group	Set Time	
Redundancy Partner Not Present	0/RP0/CPU0	Critical	Software	10/27/2023 00:22:08 UTC	
Module redundancy lost	0	Major	Environ	10/27/2023 00:23:48 UTC	Power
Plane-0 status	0/RP0/CPU0	Minor	Fabric	10/27/2023 00:22:39 UTC	Fabric
Plane-1 status	0/RP0/CPU0	Minor	Fabric	10/27/2023 00:22:39 UTC	Fabric
Plane-3 status	0/RP0/CPU0	Minor	Fabric	10/27/2023 00:22:39 UTC	Fabric

0/RP0/CPU0 Plane-5 status	Minor	Fabric	10/27/2023 00:22:39 UTC	Fabric
0/RP0/CPU0 Plane-6 status	Minor	Fabric	10/27/2023 00:22:39 UTC	Fabric
0/RP0/CPU0 Plane-7 status	Minor	Fabric	10/27/2023 00:22:39 UTC	Fabric
0/RP0/CPU0 Communications Failure With Cisco Licensing Cloud	Major	Software	10/27/2023 00:22:59 UTC	
0 Module redundancy lost	Major	Environ	10/27/2023 00:23:48 UTC	Power

Power Redundancy Protection

Table 98: Feature History Table

Feature Name	Release Information	Feature Description
Power Redundancy Protection	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8011-4G24Y4H-I • 8712-MOD-M
Power Redundancy Protection	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-36EH • 88-LC1-12TH24FH-E • 88-LC1-52Y8H-EM

Feature Name	Release Information	Feature Description
Power Redundancy Protection	Release 24.1.1	<p>You can now prevent power module exhaustion or failure due to power redundancy issues in the power feeds with the help of alarms that warn that the total output power required by the router exceeds the total feed redundancy capacity. You can configure either single-fault protection or dual fault protection, depending on whether you want to trigger alarms during redundancy failures in the power supply feed, PSU redundancy, or both.</p> <p>The feature introduces these changes:</p> <p>CLI:</p> <ul style="list-style-type: none"> • power-mgmt feed-redundancy • The <code>Total feed redundancy capacity</code> field is added to the show environment command.

The Cisco 8000 Series Modular Routers have two redundancy mechanisms to ensure the router continues functioning even during power supply failures:

- The PSU redundancy involves having extra power supplies that can take over if one fails, ensuring continuous operation.
- The power feed redundancy divides the input power into A and B feeds. When both feeds are functioning normally, they share the power load equally. However, if one of the feeds fails, the other feed scales up to its maximum capacity or the power supply unit (PSU) will operate with reduced input to ensure that the power supply to the router is uninterrupted.

These power redundancy options provide a high level of reliability and minimize the risk of network downtime due to power supply failures.

The routers now have power redundancy protection that triggers alarms for PSU and feed redundancy failures when the total output power required by the router exceeds its total feed redundancy capacity. You can configure the total feed redundancy capacity in two modes- single fault protection and dual fault protection.

The **single fault protection** mode monitors the router against a **power supply feed or PSU** redundancy failure. Meanwhile, the **dual fault protection** monitors the router against a **power supply feed and PSU** redundancy failure simultaneously. You can also customize the PSU single feed capacity in the router. Each PSU has a default power range for the single feed; you can configure a value within the range to meet your specific infrastructure requirements.

The feed redundancy alarm is triggered when the total output power required exceeds the total feed redundancy capacity. The router's total feed capacity is determined by the least of two factors: feed redundancy capacity

and PSU redundancy capacity. The PSU redundancy capacity is the number of power supply units minus the redundant ones (N) multiplied by a dual feed capacity. On the other hand, the feed redundancy capacity is the total number of PSUs multiplied by a single feed capacity. In single-fault protection, the PSU refers to the router's total number of power supply units (N+1). In dual-fault protection, the PSU refers to the number of power supply units minus the redundant ones (N).

For example, consider a router that has a total of 9 PSUs with a default N + 1 power redundancy configuration. The PSU feed capacity with dual feed is 4800 W and the single feed capacity value is set 3200 W, then the total feed redundancy capacity would be:

Power Redundancy Protection	Total Number of PSUs	PSU redundancy	Number of PSUs minus the redundant ones (N)	Dual Feed Capacity	Single Feed Capacity	Feed Redundancy Capacity	PSU Redundancy Capacity	Total Feed Redundancy Capacity
Single fault protection	9	N+1	8	4800 W	3200 W	28800 W	38400 W	28800 W
Dual fault protection	9	N+1	8	4800 W	3200 W	25600 W	38400 W	25600 W

Guidelines and Restrictions for Power Redundancy Protection

- By default, the router doesn't enable Power Redundancy Protection.
- The Power Redundancy Protection feature doesn't impact the power budgeting in the routers.
- For maximum power redundancy protection, use the dual fault protection.
- For total feed redundancy capacity calculations, the router considers only the PSUs with A and B inputs. Both A and B inputs must be within the operating range in healthy conditions. If either feed is unavailable, the router excludes such PSUs from the calculations.
- The router considers all PSUs, including redundant PSUs with two feeds (within the operating range in healthy condition) for feed redundancy capacity in single fault protection. However, the router excludes the redundant PSUs for feed redundancy capacity in dual fault protection. If the router has 8 PSUs and N+3 redundancy, single fault protection calculation considers all eight PSUs, whereas dual fault protection considers just 5 PSUs.

Configure Power Redundancy Protection

To configure the power redundancy protection mode and PSU single feed capacity, you can use the [power-mgmt feed-redundancy](#) command.

Single fault protection with PSU single feed capacity set to 2400 Watts

Configuration:

```
Router# config
Router(config)# power-mgmt feed-redundancy single-fault-protection capacity 2400
Router(config)# commit
```

Running Configuration:

```
Router# show run power
...
power-mgmt feed-redundancy single-fault-protection capacity 2400
...
```

Verification:

```
Router# show env power
=====
CHASSIS LEVEL POWER INFO: 0
=====
Total output power capacity (N + 1)      : 28800W + 4800W
Total output power required          : 6679W >>>> 1
Total power input                        : 2394W
Total power output                       : 2066W
Total feed redundancy capacity (Single Fault) : 16800W >>>> 2
//*The router triggers feed redundancy loss alarm when 1 > 2.**//
=====
Power      Supply      -----Input-----      -----Output---      Status
Module     Type                Volts A/B    Amps A/B    Volts      Amps
=====
0/PT0-PM0  PSU4.8KW-DC100     62.8/62.7   2.6/2.5    55.2       5.3      OK
0/PT0-PM1  PSU4.8KW-DC100     62.7/62.7   2.7/2.6    55.3       5.3      OK
0/PT0-PM3  PSU4.8KW-DC100     61.0/62.7   2.6/2.5    55.2       4.8      OK
0/PT1-PM0  PSU4.8KW-DC100     67.3/67.3   2.7/2.5    55.3       5.2      OK
0/PT1-PM1  PSU4.8KW-DC100     67.3/67.2   2.8/2.7    55.3       5.7      OK
0/PT1-PM2  PSU4.8KW-DC100     67.3/67.4   2.7/2.7    55.2       5.6      OK
0/PT1-PM3  PSU4.8KW-DC100     67.3/67.3   2.6/2.5    55.3       5.5      OK
Total of Power Modules:                2394W/36.7A                2066W/37.4A
```

Dual fault protection with PSU single feed capacity set to 2400 Watts

Configuration:

```
Router# config
Router(config)# power-mgmt feed-redundancy dual-fault-protection capacity 2400
Router(config)# commit
```

Running Configuration:

```
Router# show run power
...
power-mgmt feed-redundancy dual-fault-protection capacity 2400
...
```

Verification:

```
Router# show env power
=====
CHASSIS LEVEL POWER INFO: 0
=====
Total output power capacity (N + 1)      : 28800W + 4800W
Total output power required          : 6679W >>>> 1
Total power input                        : 2394W
Total power output                       : 2066W
Total feed redundancy capacity (Dual Fault) : 14400W >>>> 2
//*The router triggers feed redundancy loss alarm when 1 > 2.**//
=====
Power      Supply      -----Input-----      -----Output---      Status
Module     Type                Volts A/B    Amps A/B    Volts      Amps
=====
0/PT0-PM0  PSU4.8KW-DC100     62.8/62.7   2.6/2.5    55.2       5.3      OK
0/PT0-PM1  PSU4.8KW-DC100     62.7/62.7   2.7/2.6    55.3       5.3      OK
0/PT0-PM3  PSU4.8KW-DC100     61.0/62.7   2.6/2.5    55.2       4.8      OK
```

```

0/PT1-PM0  PSU4.8KW-DC100  67.3/67.3  2.7/2.5  55.3  5.2  OK
0/PT1-PM1  PSU4.8KW-DC100  67.3/67.2  2.8/2.7  55.3  5.7  OK
0/PT1-PM2  PSU4.8KW-DC100  67.3/67.4  2.7/2.7  55.2  5.6  OK
0/PT1-PM3  PSU4.8KW-DC100  67.3/67.3  2.6/2.5  55.3  5.5  OK

Total of Power Modules:      2394W/36.7A      2066W/37.4A

```

Alarms for power redundancy loss

You can use the `show alarms brief` command to view the power redundancy alarm:



Note The router triggers the Power Module redundancy feed mode lost alarm only when **Total output power required** exceeds **Total feed redundancy capacity**.

```
Router# show alarms brief system active
```

```
-----
Active Alarms
```

```
-----
Location          Severity      Group          Set Time          Description
```

```
-----
0                Major        Environ        11/27/2023 12:55:08 UTC  Power Module redundancy
feed mode lost
```

System Log messages for power redundancy loss

Syslog message created while power redundancy loss (total output power exceeds total feed redundancy capacity):

```
RP/0/RP0/CPU0:Dec 15 10:24:29.489 UTC: envmon [123]: %PKT_INFRA-FM-3-FAULT_MAJOR : ALARM_MAJOR
:Power Feed redundancy lost :DECLARE :0
```

Ability to Set Maximum Power Limit for the Router

Table 99: Feature History Table

Feature Name	Release Information	Feature Description
Ability to Set Maximum Power Limit for the Router	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8011-4G24Y4H-I • 8712-MOD-M

Feature Name	Release Information	Feature Description
Ability to Set Maximum Power Limit for the Router	Release 24.4.1	<p>Introduced in this release on: Fixed Systems(8200, 8700); Modular Systems (8800 [LC ASIC: P100]) (select variants only*).</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-32FH-M • 8711-32FH-M • 88-LC1-12TH24FH-E
Ability to Set Maximum Power Limit for the Router	Release 7.11.1	<p>We are introducing functionality to set the maximum power limit for a router to improve power management and distribution in the PSUs. It prevents a router from using more than the configured power and also gives the ability to limit the reservation pool regardless of how many power supplies are present. In the previous releases, the ability to prevent a router from using more than a configured amount of power was unavailable.</p> <p>This feature introduces the following change:</p> <p>CLI</p> <ul style="list-style-type: none"> • power-mgmt configured-power-capacity

In the earlier releases, there was no mechanism to limit the power a router consumed. Routers could draw more than the infrastructure could handle. Over power consumption could result in system brownout.

With the Cisco IOS XR Software Release 7.11.1, you can allocate system power based on max power capacity configuration. This prevents the router from allocating more power than the infrastructure can handle. It also gives you the ability to limit power to a router according to your infrastructure requirements. The max power capacity parameter doesn't allow power consumed by the hardware to cross the configured amount.

The criteria to set maximum power limit is that the value must be set between the current allocated power and the available maximum power at time of configuration.

This feature is not applicable for fixed routers.

A new command **power-mgmt configured-power-capacity** has been introduced with this feature.

A new alarm **PKT_INFRA-FM-3-FAULT_MAJOR : ALARM_MAJOR :Power reservation exceeds configured power** is introduced to be raised when the max power capacity is crossed.



Note This alarm is extremely rare and is raised only when the power reservation exceeds configured power. This can only happen when hardware is inserted, it is granted power without a request, such as a fan tray.

Configuring the Compatibility Mode for Various NPU Types

Table 100: Feature History Table

Feature Name	Release Information	Description
Configure Compatibility Mode for P200-based Line Cards	Release 25.4.1	<p>You can now configure the compatibility settings for line cards installed in a router to operate in P200 mode. When P200 mode is enabled, only F100-based Fabric Cards (FCs) and P200-based line cards are supported.</p> <p>To enable the P200 NPU mode, use the hw-module profile npu-compatibility command.</p>
Optimizing NPU Mode Compatibility for Route Processor Upgrades	Release 24.1.1	<p>When installing Route Processor (RP) cards from different NPU modes or NPU families, the system prioritizes newer generations over older generations. Upgrading to a newer RP, like the 8800-RP2, maintains performance by allowing the use of the Q200 NPU mode without needing to revert to Q100 NPU mode.</p> <p>You can switch to a different NPU mode by using the hw-module profile npu-compatibility command.</p>

Feature Name	Release Information	Description
Configure Compatibility Mode for Q100 and Q200-based Line Cards	Release 7.7.1	<p>You can now configure the compatibility behavior of line cards to operate in Q100 mode (default behavior) or in Q200 mode when you have a mix of Q100-based line cards and Q200-based line cards that are installed in a router.</p> <p>In earlier releases, in a mixed mode combination, if multiple generations of line cards were installed on a distributed chassis, the second-generation line cards interoperated with the first-generation line cards. As a result, the NPUs set lower resource limits for the newer generation line cards to ensure backward compatibility. The router couldn't fully utilize the improved scale, higher capacity, and enhanced features of the newer generation line cards.</p> <p>This compatibility feature now enables you to select if you want the line cards to operate in Q100 or Q200 NPU mode.</p> <p>The hw-module profile npu-compatibility command is introduced for this feature.</p>

This table details the old and the new behavior when a mix of line cards from different NPUs is installed on a router.

Scenario	If..	Then..	Example
Old behavior	you install a mix of Q100-based line cards and Q200-based line cards	the Q200-based line cards operate in a scaled-down (Q100) mode by default.	If a router has a Q100 NPU-based line card and you add a line card from the Q200 NPU-based line card, the Q200 NPU line card operates in a scaled down mode to work with the Q100 line cards.
New behavior	you want line cards to operate in Q100 (default behavior), Q200, or P100 mode	you can select the mode.	If you select Q200 mode, the router boots only the Q200-based line cards and gracefully shuts down the Q100-based line cards.

FAQs About the Compatibility Modes for Various NPU Types

- **Can the line cards still be used in scaled down mode, like in the previous scenario?**
Yes, you can still switch to the previous implementation, if you may, to the scaled down mode.
- **What all ASICs can participate in the compatibility mode implementation?**

P200, P100, Q200, Q100.

- **Is there any default ASIC set by the system?**

The ASIC default is based on the Fabric Cards (FCs) and route processor cards used in a distributed chassis. However, you can choose to change the ASIC mode to Q200, Q100, P100, or P200.

- **Do I need to reboot the router after implementing a new NPU mode?**

Yes, reboot the router for the new NPU mode to take effect.

- **What defines an NPU mode?**

NPU mode is determined by the Route Processor (RP) and the Fabric Card (FC). During the router's boot-up process, it initially identifies the RP and the FC, setting the corresponding NPU mode regardless of the line cards present in the router.

Guidelines for configuring compatibility mode

These guidelines apply when you configure the line cards from different ASIC families:

- By default, a mix of Q100 and Q200 line cards results in the Q200 line cards operating in Q100 (scaled-down) mode. Configuring Q100 mode results in the same (default) behavior. Similarly, a mix of P100 and Q200 line cards results in the Q200 line cards operating in P100 (scaled-down) mode. Configuring P100 mode results in the same (default) behavior.
- To use the improved scale, higher capacity, and feature-rich capabilities of the Q200-based line cards, use the `hw-module profile npu-compatibility` command and set it to operate in the Q200 mode. Else, the Q200-based line cards scale down to the Q100 mode, which is the default behavior. The same behavior applies to the P100-based line cards.
- Reboot the router for the compatibility mode to take effect. If the system detects a noncompatible line card, it shuts down that line card. For example, in Q200 mode, the router boots only the Q200-based line cards and gracefully shuts down the Q100-based line cards.
- For 8800-RP, the default NPU mode is Q100. For 8800-RP2, the default NPU mode is Q200.
- For the various fabric card types available, the following scenarios may be applicable:
 - 8800-RP Route Processor Card - if the router boots up with an 8800-RP route processor card without any fabric card, then the default mode is set to Q100.
 - 8800-RP2 Route Processor Card - if the router boots up with a 8800-RP2 route processor card without any fabric card, then the router sets the default mode to P100. If you insert a Q200 fabric card, then router reload is required.
 - Swapping Fabric Cards - if the router initially boots with Q200 fabric cards and you later replace them with F100 fabric cards, a router reload is necessary.

This table lists the Q100, Q200, P100-based, and P200 line cards that support the compatibility mode:

ASIC Family	Line Card
Q100-based line cards	8800-LC-48H
	8800-LC-36FH

ASIC Family	Line Card
Q200-based line cards	88-LC0-34H14FH
	88-LC0-36FH
	88-LC0-36FH-M
P100-based line cards	88-LC1-36EH
	88-LC1-12TH24FH-E
	88-LC1-52Y8H-EM

Restrictions for configuring compatibility mode

These restrictions apply when you configure the line cards from different ASIC families:

- The `hw-module profile npu-compatibility` command isn't configurable on the Cisco 8200 Series fixed router and Cisco 8608 router.
- Q100-based ASIC is not supported with 8800-RP2-S.
- P200-based ASIC

Route Processor Card Behavior with NPUs

A newer generation Route Processor (RP) card takes precedence over an older generation RP card when installed from different NPU modes. The precedence followed by the system is: P200 > P100 > Q200 > Q100.

If you have Q200-based line cards and an older generation RP card (8800-RP) installed on your router, the router boots with Q100 ASIC mode for the line cards. However, you can change the ASIC mode from Q100 to Q200 by using the `hw-module profile npu-compatibility` command. Setting the ASIC mode to a newer generation ASIC allows you to utilize their improved scale, higher capacity, and feature-rich capabilities when you replace your RPs with a newer generation RP.

For instance, if your router is equipped with an 8800-RP route processor card set to ASIC mode as Q200, upgrading to an 8800-RP2 RP card won't require changing the ASIC mode from Q100 to Q200.

Line Card Behavior with NPUs

If you have various line cards installed from different NPU families, the newer generation line cards take precedence over an older generation line card. The precedence followed by the system is: P200 > P100 > Q200 > Q100.

Configuring NPU compatibility for Line Cards

To configure a router for handling line cards of different NPU-based line cards, use the `hw-module profile npu-compatibility` command. To go back to the default mode, use the `no` form of this command.

The following are the options available in command and their descriptions:

<code>npu-compatibility</code>	Allows you to make a router compatible with a NPU family.
<code>mode-name</code>	Allows you to set the mode, such as Q100, Q200, P100, or P200.

The following is a configuration example:

```
Router:ios(config)#hw-module profile npu-compatibility q200
Tue Dec 7 15:06:53.697 UTC
Chassis mode will be activated after a manual reload of chassis/all line cards
Router:ios(config)#commit
Tue Dec 7 15:06:54.646 UTC
LC/0/1/CPU0:Dec 7 15:06:54.796 UTC: npu_drvr292:
%FABRIC-NPU_DRV-3-HW_MODULE_PROFILE_NPU_COMPATIBILITY_CHASSIS_CFG_CHANGED : Please reload
chassis for the configuration to take effect
end
Router:ios(config)#end
Router:ios#
```

Running Configuration

```
RP/0/RP0/CPU0:ios# show ver
Mon Jun 27 19:25:52.947 UTC
Cisco IOS XR Software, Version 7.7.1.27I LNT
Copyright (c) 2013-2022 by Cisco Systems, Inc.
```

Build Information:

```
Built By      : ingunawa
Built On      : Wed Jun 01 23:50:09 UTC 2022
Build Host    : iox-ucs-060
Workspace     : /auto/iox-ucs-060-san1/prod/7.7.1.27I.SIT_IMAGE/8000/ws
Version       : 7.7.1.27I
Label        : 7.7.1.27I
```

```
cisco 8000 (VXR)
cisco 8808 (VXR) processor with 32GB of memory
ios uptime is 3 minutes
Cisco 8808 8-slot Chassis
```

```
RP/0/RP0/CPU0:ios#
```

```
RP/0/RP0/CPU0:ios# conf
Mon Jun 27 19:24:40.621 UTCRP/0/RP0/CPU0:ios(config)#hw-module profile npu-compatibility ?
```

```
P100 Use P100 for Chassis mode
Q100 Use Q100 for Chassis mode
Q200 Use Q200 for Chassis mode
```

Verification

```
RP/0/RP0/CPU0:ios# show hw-module profile npu-compatibility matrix
Wed Nov 17 02:00:28.652 UTC
```

Node	Card Type	NPU Type		
0/0/CPU0	88-LC0-36FH	Q200		
0/1/CPU0	88-LC1-36EH	P100		
0/2/CPU0	88-LC1-36EH	P100		
0/3/CPU0	88-LC1-36EH	P100		

NPU Type	Compatibility		Compatibility		Compatibility
	Mode Q100	Mode Q200	Mode G100	Mode P100	
Q100	Compatible	Not Compatible	Not Compatible	Not Compatible	
Q200	Compatible	Compatible	Not Compatible	Not Compatible	

```

Not Compatible      Not Compatible      Not Compatible
G100      Not Compatible      Compatible      Compatible      Not Compatible
Not Compatible      Not Compatible      Not Compatible
P100      Not Compatible      Not Compatible      Not Compatible      Not Compatible
Not Compatible      Not Compatible      Not Compatible
A100      Not Compatible      Not Compatible      Not Compatible      Not Compatible
Not Compatible      Not Compatible      Not Compatible
K100      Not Compatible      Not Compatible      Not Compatible      Not Compatible
Not Compatible      Not Compatible      Not Compatible
F100      Not Compatible      Not Compatible      Not Compatible      Not Compatible
Not Compatible      Not Compatible      Not Compatible
Default mode : P100
RP/0/RP0/CPU0:ios#

```

Storage Media Sanitization

Table 101: Feature History Table

Feature Name	Release Information	Feature Description
Storage Media Sanitization	Release 7.3.4	<p>To comply with NIST SP 800-88 guidelines for Media Sanitization, it is important that your organization ensures that no easily reconstructible data is stored in the router and associated devices after it has left the control of your organization or is no longer protected by confidentiality categorization.</p> <p>With this feature, you can erase and overwrite any sensitive data, configuration, or keys present in the route processor or line card, ensuring media sanitization and preventing unauthorized data retrieval.</p>

When you identify an RP or line card for RMA, or you require to ship it outside your organization, a service personnel may not be available on-site to remove the card immediately. However, you can reset your RP or line card to erase customer-sensitive data and let the RP or line card remain in the slot.

Factory reset of routers to remove SSD data

Factory reset of routers is a data security feature that wipes out the data from the solid state drive (SSD) and restores the routers to their original state. This feature is beneficial for users who want to restore routers to their original factory settings, often for troubleshooting or for re-purposing.

You can choose to reboot the router with the current software version or shut it down post-reset. After performing factory reset operation, the router OS does not automatically revert to the original Cisco IOS XR Software version with which it was shipped.

From Cisco IOS XR Software Release 25.1.1, the factory reset functionality is enhanced to securely remove data from the entire hard disk of the router, except the disaster recovery partition on the active RP.

Table 102: Feature History Table

Feature Name	Release Information	Feature Description
Factory reset of routers to remove SSD data	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: Q200, P100], 8010 [ASIC: A100]); Centralized Systems (8600 [ASIC: Q200]); Modular Systems (8800 [LC ASIC: Q100, Q200, P100])</p> <p>You can now securely reset your router to its original factory settings for troubleshooting or re-purposing. We have enhanced the existing factory reset functionality to remove data from the entire hard disk of the router, except the disaster recovery partition on the active RP.</p>

Benefits of factory reset functionality

The factory reset functionality on Cisco IOS XR routers provides these benefits:

- It restores the router to its original state by wiping out the data on the SSD.
- It facilitates troubleshooting by providing a fresh start for the router.
- You can initiate factory reset for specific nodes such as RP or LC, or for the entire chassis.

Guidelines and restrictions for factory reset functionality

These guidelines and restrictions apply to factory reset functionality on routers:

- You cannot initiate factory reset if the entire system is down or if no active RP is booted to IOS XR OS.
- We recommend using **factory-reset** without performing **commit replace** for securely removing the files in the misc/config folder. This guideline is applicable only until Cisco IOS XR Software Release 25.1.1.
- The RP or line card shuts down automatically if the factory reset takes more than 30 minutes, you can perform the factory reset again. The console displays this log message during automatic shutdown:

```
[ TIME ] Timed out starting Power-Off.
[ !! ] Forcibly powering off as result of failure.
```

This behavior is applicable only until Cisco IOS XR Software Release 25.1.1.

- If your router has dual RPs, and to perform the factory reset on both the RPs, first reset the standby RP from the active RP. After the reset is complete, you can then reset the active RP.
- From Cisco IOS XR Software Release 25.1.1, we support the *zero_fill* option from *gNOI factory_reset.proto*.

- The factory reset operation does not completely wipe out the data on the hard disk of the active RP because the disaster recovery partitioning is not removed.

You can perform a secure erase operation if you need to wipe out the data on the disaster recovery partition as well. For details, see [Secure erase of router SSD data, on page 478](#).

Perform factory reset on a router

Factory reset functionality supports these scenarios:

- Reload option: resets the router and reboots it
- Shutdown option: resets the router and shuts it down
- Location option: applies the reset operation to specific locations such as individual line card (LC) or route processor (RP)

Use the **factory-reset** command for erasing these folders of RP or LC (this is applicable only until Cisco IOS XR Software Release 25.1.1):

- /misc/disk1
- /misc/scratch
- /var/log
- /misc/config

Before you begin

- Device must be operational and booted to IOS XR OS to initiate factory reset.
- Ensure that there is no immediate requirement for the router after the operation, as it involves complete data removal and shutdown.
- Take a backup of the router data as a precautionary measure.

Procedure

Step 1 Initiate factory reset process on the router CLI.

- Reload option:

```
Router#factory-reset reload location 0/RP1/CPU0
Tue Mar 11 11:18:43.222 UTC
Performing factory-reset may affect the stability of the system. Re-imaging maybe required to
recover. Continue?
[confirm]
```

- Shutdown option:

```
Router#factory-reset shutdown location 0/RP1/CPU0
Tue Mar 11 11:18:43.222 UTC
Performing factory-reset may affect the stability of the system. Re-imaging maybe required to
```

```
recover. Continue?
[confirm]
```

The factory reset command with the **location** *location-id* option erases customer-sensitive data in the specified location.

Step 2 Check the system logs to confirm that the factory reset process is completed.

Example:

System logs for factory reset process with **reload** option:

```
RP/0/RP0/CPU0:Mar 18 09:44:00.573 UTC: shelfmgr_disk_erase_cli[69035]:
%PLATFORM-SHELFMGR-4-FACTORY_RESET : User cisco requested 'factory reset reload' of 0/RP1/CPU0
RP/0/RP0/CPU0:Mar 18 09:44:12.507 UTC: shelfmgr[420]:
%PLATFORM-CPA_INTF_SHELFMGR-4-CARD_REIMAGE_CFG_DONE : Successfully configured card 0/RP1/CPU0 for
reimage operation, boot mode: IPXE_INTERNAL
RP/0/RP0/CPU0:Mar 18 09:45:11.499 UTC: shelfmgr[420]: %PLATFORM-CPA_INTF_SHELFMGR-6-BIOS_INFO :
0/RP1/CPU0: BIOS banner - (Cisco 8000(R) Series BIOS Ver 1.38 Primary).
RP/0/RP0/CPU0:Mar 18 09:45:58.696 UTC: shelfmgr[420]: %PLATFORM-SHELFMGR-4-DISK_ERASE_START : Started
disk erase operation on 0/RP1/CPU0
RP/0/RP0/CPU0:Mar 18 09:46:00.362 UTC: shelfmgr[420]: %PLATFORM-SHELFMGR-4-DISK_ERASE_IN_PROGRESS :
[bash(950)] Performing NIST recommended purge sanitization method on /dev/sda on 0/RP1/CPU0
RP/0/RP0/CPU0:Mar 18 09:59:06.628 UTC: shelfmgr[420]: %PLATFORM-SHELFMGR-4-DISK_ERASE_DONE : Disk
erase operation finished successfully on 0/RP1/CPU0
RP/0/RP0/CPU0:Mar 18 10:07:00.805 UTC: shelfmgr[420]: %PLATFORM-SHELFMGR-6-INFO_LOG : 0/RP1/CPU0 is
operational
```

System logs for factory reset process with **shutdown** option:

```
RP/0/RP0/CPU0:Mar 18 10:30:04.999 UTC: shelfmgr_disk_erase_cli[66157]:
%PLATFORM-SHELFMGR-4-FACTORY_RESET : User cisco requested 'factory reset shutdown' of 0/RP1/CPU0
RP/0/RP0/CPU0:Mar 18 10:30:16.351 UTC: shelfmgr[420]:
%PLATFORM-CPA_INTF_SHELFMGR-4-CARD_REIMAGE_CFG_DONE : Successfully configured card 0/RP1/CPU0 for
reimage operation, boot mode: IPXE_INTERNAL
RP/0/RP0/CPU0:Mar 18 10:31:15.373 UTC: shelfmgr[420]: %PLATFORM-CPA_INTF_SHELFMGR-6-BIOS_INFO :
0/RP1/CPU0: BIOS banner - (Cisco 8000(R) Series BIOS Ver 1.38 Primary).
RP/0/RP0/CPU0:Mar 18 10:32:01.946 UTC: shelfmgr[420]: %PLATFORM-SHELFMGR-4-DISK_ERASE_START : Started
disk erase operation on 0/RP1/CPU0
RP/0/RP0/CPU0:Mar 18 10:32:03.605 UTC: shelfmgr[420]: %PLATFORM-SHELFMGR-4-DISK_ERASE_IN_PROGRESS :
[bash(948)] Performing NIST recommended purge sanitization method on /dev/sda on 0/RP1/CPU0
RP/0/RP0/CPU0:Mar 18 10:44:52.577 UTC: shelfmgr[420]: %PLATFORM-SHELFMGR-4-DISK_ERASE_DONE : Disk
erase operation finished successfully on 0/RP1/CPU0
RP/0/RP0/CPU0:Mar 18 10:44:58.325 UTC: shelfmgr[420]: %PLATFORM-SHELFMGR-6-INFO_LOG : 0/RP1/CPU0 is
shutdown
```

The logs are displayed on the console port of the node where the reset is performed.

Step 3 Verify that the factory reset process is completed.

Example:

This example shows how to verify the factory reset process that is performed with the **shutdown** option:

```
Router#show shelfmgr history events location 0/RP1/CPU0
Tue Mar 18 10:45:05.257 UTC
NODE NAME      : 0/RP1/CPU0
CURRENT STATE  : CARD_SHUT_POWERED_OFF
TIME STAMP     : Mar 18 2025 10:44:58
-----
DATE           TIME (UTC)  EVENT                STATE
-----
Mar 18 2025 10:45:03  ev_powered_off      STATE_NOT_CHANGED
Mar 18 2025 10:44:59  ev_unmapped_event   STATE_NOT_CHANGED
```

```

Mar 18 2025 10:44:58 ev_fault_fatal_powered_o CARD_SHUT_POWERED_OFF
Mar 18 2025 10:44:52 ev_factory_reset_done FACTORY_RESET_DONE
Mar 18 2025 10:32:03 ev_factory_reset_in_prog FACTORY_RESET_IN_PROGRESS
Mar 18 2025 10:32:02 ev_factory_reset_in_prog FACTORY_RESET_IN_PROGRESS
Mar 18 2025 10:32:01 ev_factory_reset_started FACTORY_RESET_IN_PROGRESS
Mar 18 2025 10:31:59 ev_install_image INSTALLING_IMAGE
Mar 18 2025 10:31:50 ev_kernel_booting KERNEL_BOOTING
Mar 18 2025 10:31:50 ev_unmapped_event STATE_NOT_CHANGED
Mar 18 2025 10:31:36 ev_unmapped_event STATE_NOT_CHANGED
Mar 18 2025 10:31:35 ev_unmapped_event STATE_NOT_CHANGED
Mar 18 2025 10:31:16 ev_ipxe_download DOWNLOADING_IMAGE
Mar 18 2025 10:31:13 ev_bios_ready BIOS_READY
Mar 18 2025 10:30:33 ev_powered_on STATE_NOT_CHANGED
Mar 18 2025 10:30:25 ev_powered_on CARD_POWERED_ON
Mar 18 2025 10:30:25 ev_unmapped_event STATE_NOT_CHANGED
Mar 18 2025 10:30:20 ev_powered_off CARD_POWERED_OFF
Mar 18 2025 10:30:16 ev_unmapped_event STATE_NOT_CHANGED
Mar 18 2025 10:30:16 transient_condition CARD_RESETTING
Mar 18 2025 10:30:16 ev_check_card_down_reaso CHECKING_DOWN_REASON
Mar 18 2025 10:30:16 ev_os_halted OS_HALTED
Mar 18 2025 10:30:07 ev_os_halting OS_HALT_IN_PROGRESS
Mar 18 2025 10:30:06 ev_xr_shut START_OS_HALT
Mar 18 2025 10:30:05 ev_ack_ok STATE_NOT_CHANGED
Mar 18 2025 10:30:05 ev_graceful_reimage CARD_SHUTDOWN_IN_PROGRESS
Mar 18 2025 10:07:00 ev_xr_ready XR_RUN
    
```

This example shows how to verify the factory reset process that is performed with the **reload** option:

```

Router#show shelfmgr history events location 0/RP1/CPU0
Tue Mar 18 10:26:08.161 UTC
NODE NAME      : 0/RP1/CPU0
CURRENT STATE  : XR_RUN
TIME STAMP     : Mar 18 2025 10:07:00
-----
DATE           TIME (UTC)  EVENT                               STATE
-----
Mar 18 2025 10:07:00 ev_xr_ready XR_RUN
Mar 18 2025 10:06:14 ev_ack_ok    CARD_STATUS_CHECK_COMPLETE
Mar 18 2025 10:06:14 ev_card_status_check CARD_STATUS_CHECK
Mar 18 2025 10:06:14 ev_ack_ok    CARD_COMPATIBILITY_CHECK
Mar 18 2025 10:06:14 ev_xr_config_ready CARD_COMPATIBILITY_CHECK
Mar 18 2025 10:06:14 ev_card_info_rcvd  CARD_INFO_RCVD
Mar 18 2025 10:05:57 ev_xr_init    XR_INITIALIZING
Mar 18 2025 10:05:14 ev_kernel_booting STATE_NOT_CHANGED
Mar 18 2025 10:04:03 ev_kernel_booting KERNEL_BOOTING
Mar 18 2025 10:04:02 ev_install_success IMAGE_INSTALLED
Mar 18 2025 10:03:44 ev_install_image INSTALLING_IMAGE
Mar 18 2025 10:01:06 ev_install_image INSTALLING_IMAGE
Mar 18 2025 10:01:03 ev_install_image INSTALLING_IMAGE
Mar 18 2025 10:01:02 ev_install_image INSTALLING_IMAGE
Mar 18 2025 10:00:56 ev_install_image INSTALLING_IMAGE
Mar 18 2025 10:00:55 ev_install_image INSTALLING_IMAGE
Mar 18 2025 10:00:48 ev_install_image INSTALLING_IMAGE
Mar 18 2025 10:00:48 ev_install_image INSTALLING_IMAGE
Mar 18 2025 10:00:47 ev_install_image INSTALLING_IMAGE
Mar 18 2025 10:00:44 ev_install_image INSTALLING_IMAGE
Mar 18 2025 10:00:30 ev_install_image INSTALLING_IMAGE
Mar 18 2025 10:00:29 ev_install_image INSTALLING_IMAGE
Mar 18 2025 10:00:29 ev_install_image INSTALLING_IMAGE
Mar 18 2025 10:00:28 ev_install_image INSTALLING_IMAGE
Mar 18 2025 10:00:28 ev_install_image INSTALLING_IMAGE
Mar 18 2025 09:59:27 ev_install_image INSTALLING_IMAGE
Mar 18 2025 09:59:06 ev_factory_reset_done FACTORY_RESET_DONE
Mar 18 2025 09:46:00 ev_factory_reset_in_prog FACTORY_RESET_IN_PROGRESS
    
```

```

Mar 18 2025 09:45:59 ev_factory_reset_in_prog FACTORY_RESET_IN_PROGRESS
Mar 18 2025 09:45:58 ev_factory_reset_started FACTORY_RESET_IN_PROGRESS
Mar 18 2025 09:45:55 ev_install_image INSTALLING_IMAGE
Mar 18 2025 09:45:46 ev_kernel_booting KERNEL_BOOTING
Mar 18 2025 09:45:46 ev_unmapped_event STATE_NOT_CHANGED
Mar 18 2025 09:45:32 ev_unmapped_event STATE_NOT_CHANGED
Mar 18 2025 09:45:31 ev_unmapped_event STATE_NOT_CHANGED
Mar 18 2025 09:45:12 ev_ipxe_download DOWNLOADING_IMAGE
Mar 18 2025 09:45:10 ev_bios_ready BIOS_READY
Mar 18 2025 09:44:29 ev_powered_on STATE_NOT_CHANGED
Mar 18 2025 09:44:21 ev_powered_on CARD_POWERED_ON
Mar 18 2025 09:44:21 ev_unmapped_event STATE_NOT_CHANGED
Mar 18 2025 09:44:16 ev_powered_off CARD_POWERED_OFF
Mar 18 2025 09:44:12 ev_unmapped_event STATE_NOT_CHANGED
Mar 18 2025 09:44:12 transient_condition CARD_RESETTING
Mar 18 2025 09:44:12 ev_check_card_down_reaso CHECKING_DOWN_REASON
Mar 18 2025 09:44:12 ev_os_halted OS_HALTED
Mar 18 2025 09:44:04 ev_os_halting OS_HALT_IN_PROGRESS
Mar 18 2025 09:44:02 ev_xr_shut START_OS_HALT
Mar 18 2025 09:44:00 ev_ack_ok STATE_NOT_CHANGED
Mar 18 2025 09:44:00 ev_graceful_reimage CARD_SHUTDOWN_IN_PROGRESS
Mar 17 2025 16:28:24 ev_xr_ready XR_RUN

```

Secure erase of router SSD data

Secure erase is a data security feature that securely erases the solid state drive (SSD) data on a particular node such as a line card (LC) or a route processor (RP), or on the entire router and shuts it down. The feature ensures the removal of all customer-sensitive data, configurations, and keys from the storage device (SSD) in compliance with National Institute for Standards and Technology (NIST) 800-88 guidelines for media sanitization.

The secure erase feature is ideal for scenarios where the router is to be decommissioned. It is also useful when the data needs to be completely removed for security reasons.

Table 103: Feature History Table

Feature Name	Release Information	Feature Description
Secure erase of router SSD data	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: Q200, P100], 8010 [ASIC: A100]); Centralized Systems (8600 [ASIC:Q200]); Modular Systems (8800 [LC ASIC: Q100, Q200, P100])</p> <p>You can now efficiently and securely manage the data and configuration settings on your routers by ensuring complete removal of sensitive data from the routers that are to be decommissioned, or for security purposes. This feature securely erases the solid state drive (SSD) data on a particular card such as a line card or a route processor, or on the entire router and shuts it down.</p> <p>The feature introduces these changes:</p> <p>CLI:</p> <ul style="list-style-type: none"> • secure-erase <p>YANG Data Model:</p> <ul style="list-style-type: none"> • Cisco-IOS-XR-secure-erase-act

Benefits of secure erase functionality

The secure erase functionality on Cisco IOS XR routers provides these benefits:

- Complete data removal from the device for security purposes
- Suitable for device decommissioning or re-purposing
- Can be applied to individual nodes or to the entire router

Restriction for secure erase functionality

Secure erase functionality on routers is subjected to a restriction that you cannot initiate it if the entire system is down or if no active RP is booted to IOS XR OS.

Perform secure erase on a router

Before you begin

- Ensure the active RP is operational to initiate the secure erase process.
- Ensure that there is no immediate requirement for the router after the secure erase process, since it involves complete data removal and shutdown of the router.
- Keep a backup of the router data as a precautionary measure.

Procedure

Step 1 Initiate secure erase process on the router CLI.

Example:

```
Router#secure-erase location 0/RP1/CPU0
Tue Mar 11 11:17:51.294 UTC
Performing secure erase operation will erase the SSD and shut down the card. Proceed?
[confirm]
```

Step 2 Check system logs to confirm that the secure erase process is completed.

Example:

```
RP/0/RP0/CPU0:Mar 11 11:28:55.862 UTC: shelfmgr[159]:
%PLATFORM-CPA_INTF_SHELFMGR-4-CARD_REIMAGE_CFG_DONE : Successfully configured card 0/RP1/CPU0 for
reimage operation, boot mode: IPXE_INTERNAL
RP/0/RP0/CPU0:Mar 11 11:31:08.610 UTC: shelfmgr[159]: %PLATFORM-SHELFMGR-4-DISK_ERASE_START : Started
disk erase operation on 0/RP1/CPU0
RP/0/RP0/CPU0:Mar 11 11:31:10.455 UTC: shelfmgr[159]: %PLATFORM-SHELFMGR-4-DISK_ERASE_IN_PROGRESS :
[bash(1119)] Performing NIST recommended purge sanitization method on /dev/nvme0n1 on 0/RP1/CPU0
RP/0/RP0/CPU0:Mar 11 11:52:12.890 UTC: shelfmgr[159]: %PLATFORM-SHELFMGR-4-DISK_ERASE_DONE : Disk
erase operation finished successfully on 0/RP1/CPU0
RP/0/RP0/CPU0:Mar 11 11:52:18.553 UTC: shelfmgr[159]: %PLATFORM-SHELFMGR-6-INFO_LOG : 0/RP1/CPU0 is
shutdown
```

Excluding Sensitive Information in Show Running Configurations Output

Table 104: Feature History Table

Feature Name	Release Information	Feature Description
Excluding Sensitive Information in Show Running Configurations Command Output	Release 7.5.4	<p>You can now exclude sensitive information such as strings, usernames, passwords, comments, or IP addresses within the show running-configuration command output by enabling sanitization on the nonvolatile generation (NVGEN) process.</p> <p>With this feature, you can achieve better data protection to prevent cybersecurity risks compared to regular router algorithms.</p> <p>This feature introduces the nvgen default-sanitize command.</p>

The **show running configuration** command uses the nonvolatile generation (NVGEN) process in IOS-XR software to collect configuration information from every system component and construct a running configuration file to create its output. However, this file may contain sensitive information, including usernames, passwords, and IP addresses, which could pose a security threat when obfuscation algorithms in the router are weak compared to modern cryptographic standards.

In this feature, you can mask the following types of sensitive information in the show running configurations:

- Strings
- Usernames
- Passwords
- Comments
- IP Addresses

On enabling the sanitization in show running configurations, the NVGEN process replaces the corresponding information with **<removed>** string. For example, if you enable sanitization for IP Addresses, the show running configuration includes the **<removed>** string in place of all the IP Addresses in the output.

Sanitizing Strings

Configuration

```
Router# config
Router:(config)# nvgen default-sanitize strings
Router:(config)# commit
```

Running Configuration

```
Router# show run nvgen
nvgen
  default-sanitize strings
!
```

Verification

```
Router# show run int Hu0/2/0/4
interface HundredGigE0/2/0/4
  ! This is comment 1
  description <removed>
!
```

Sanitizing Usernames

Configuration

```
Router# config
Router:(config)# nvgen default-sanitize usernames
Router:(config)# commit
```

Running Configuration

```
Router# show run nvgen
nvgen
  default-sanitize usernames
!
```

Verification

```
Router# show run username test
username <removed>
  group root-lr
  password 7 172864HJWBHBCWH
!
```

Sanitizing Passwords

Configuration

```
Router# config
Router:(config)# nvgen default-sanitize passwords
Router:(config)# commit
```

Running Configuration

```
Router# show run nvgen
nvgen
  default-sanitize passwords
!
```

Verification

```
Router# show run username test
username test
  group root-lr
  password 7 <removed>
!
```

Sanitizing Comments

Configuration

```
Router# config
Router:(config)# nvgen default-sanitize comments
Router:(config)# commit
```

Running Configuration

```
Router# show run nvgen
nvgen
  default-sanitize comments
!
```

Verification

```
Router# show run int Hu0/2/0/4
interface HundredGigE0/2/0/4
  ! <comments removed>
  description This is bundle member
  !
```

Sanitizing IP Addresses

Configuration

```
Router# config
Router:(config)# nvgen default-sanitize ipaddrs
Router:(config)# commit
```

Verification

```
Router# show run int Hu0/2/0/4
interface HundredGigE0/2/0/4
  ! This is comment 1
  description This is bundle member
  ipv4 address <removed> <removed>
  !
```

Fabric Link Management for Uncorrectable Errors

Table 105: Feature History Table

Feature Name	Release Information	Feature Description
Fabric Link Management for Uncorrectable Errors	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8011-4G24Y4H-I • 8712-MOD-M

Feature Name	Release Information	Feature Description
Fabric Link Management for Uncorrectable Errors	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-36EH • 88-LC1-12TH24FH-E • 88-LC1-52Y8H-EM
Fabric Link Management for Uncorrectable Errors	Release 24.2.11	<p>You can now run your fabric links error-free using the forward error correction (FEC) technique.</p> <p>The feature allows you to determine the link quality by monitoring the noisy fabric links during and post bring-up.</p> <p>This feature introduces the hw-module fabric-fec-monitor disable command.</p>

Forward error correction (FEC) is a method for obtaining error control in data transmission in which the transmitter sends redundant data and the receiver recognizes only the portion of the data that contains no apparent errors. When FEC is used in data transmissions, the receiver can detect and correct a limited number of errors.

The Cisco IOS XR router will not bring the link to the data plane if the link is noisy at inception (during bring up). If the link becomes noisy post bring up, fabric link will be re-set and re-tuned. If this event continues for five times within an hour then fabric link will be shutdown permanently. Post link up, polling interval for link error is 10 minutes.

Fabric link management feature uses FEC as the criteria to determine if a link is good. The router receives a notification for every bad FEC on each fabric port. FEC can correct up to 15 bits beyond which the error is considered as uncorrectable error. This feature allows you to make fabric links run error-free.



Note In Cisco IOS XR Release 24.2.11, this feature is enabled only for Q200 based line cards and Fabric cards.

FEC bin index

FEC bin index indicates the number of bit errors.


```

LINK_UP_INTR          KEEPALIVE_START      Sat Jan 13 00:19:24 2018
LINK_UP_INTR          CHECK_REACH          Sat Jan 13 00:19:24 2018
LINK_UP_INTR          UP                  Sat Jan 13 00:19:24 2018
BAD_FEC               UP                  Sat Jan 13 00:20:16 2018
DIS_PERM_SHUT        MAC_UP              Sat Jan 13 00:20:16 2018
DIS_PERM_SHUT        STOPPED             Sat Jan 13 00:20:16 2018
+-----+

```

This table describes the significant fields shown in the above example.

Table 106: show controllers npu link-info Field Descriptions

Field	Description
<code>BAD_FEC_BELOW_THR</code>	There are FEC failures, but the number of failures has not exceeded the predefined threshold (in this case, 5 per hour). The router retunes and checks for FEC improvement.
<code>BAD_FEC</code>	This part of the log entry indicates that FEC detected failures, and the number of these failures surpassed a predefined threshold. As a result, the decision was made to permanently shut down the affected interface or port as a protective measure.
<code>DIS_PERM_SHUT</code>	The link or port has been intentionally disabled and is in a shutdown state after FEC fails for the threshold limit (After fifth failure).

System Log messages

The router displays the following syslog messages after retuning:

- If the link is noisy at inception (during bring up), the router displays the following syslog message after tuning for 100 times:

```

LC/0/2/CPU0:Jan 13 00:56:03.939 UTC: npu_drvr[128]:
%FABRIC-NPU_DRV-3-NPU_CPA_GEN_ERR_INFO : Link 0/254 has tuned 100 times and failed to
come up. FEC bin is filled to 11

```

- If the link is noisy post bring up, the router permanently shuts down the link and displays the following syslog message:

```

LC/0/2/CPU0:Jan 13 00:20:16.251 UTC: npu_drvr[128]:
%FABRIC-NPU_DRV-3-NPU_CPA_GEN_ERR_INFO : FEC check failures on link 0/254. FEC bin is
filled to 14

```

Disable Fabric Link Management for Uncorrectable Errors

Fabric link management for uncorrectable errors is enabled by default. To disable this feature, use the **hw-module fabric-fec-monitor disable** command in XR Config mode mode.

The following example shows how to disable the fabric FEC monitor:

```

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# hw-module fabric-fec-monitor disable
RP/0/RP0/CPU0:router(config)# commit

```

Fault recovery handling

Table 107: Feature History Table

Feature Name	Release Information	Feature Description
Fault recovery handling	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8011-4G24Y4H-I • 8712-MOD-M
Fault recovery handling	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Modular Systems (8800 [LC ASIC: P100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-36EH • 88-LC1-12TH24FH-E • 88-LC1-52Y8H-EM

Feature Name	Release Information	Feature Description
Fault recovery handling	Release 24.2.11	<p>You can now configure the number of fault recovery attempts by a line card, fabric card or a route processor before it permanently shuts down, thus preventing a faulty card from entering into a cycle of automatic recovery.</p> <p>This feature introduces the following change:</p> <p>CLI:</p> <ul style="list-style-type: none"> • hw-module fault-recovery <p>YANG DATA Model:</p> <ul style="list-style-type: none"> • New XPath for Cisco-IOS-XR-hw-module-cfg.yang (see Github, YANG Data Models Navigator)

In the previous releases, if a line card, fabric card or a route processor experienced a fault, they used to trigger fault recovery and reboot themselves to be operational. Fault recovery mechanism was time based as the fault recovery count used to reset to zero if the card remained operational for more than hour. After the fault recovery count exceeded five, then the faulty card was shut down. As power related faults triggered were not frequent, and fault recovery count used to reset to zero, the card never entered the shut down mode. As a result the card always attempted for fault recovery.

With the Cisco IOS XR Software Release 24.2.11, we have introduced the **hw-module fault-recovery** command with which you can set the number of times a fault recovery can take place before permanently shutting down a faulty card.



Note This configuration is not applicable for BMC instance

How to Configure the Fault Recovery Attempts

Configuration Examples

The configuration example shows how to configure the fault recovery attempts on the fabric card FC0.

```
Router#configure
Router (config)#hw-module fault-recovery location 0/FC0 count 1
Router (config)#commit
```

Verification

Use **show running-config formal | include hw-module** command to display the number of times a card can initiate recovery attempts before shutting down .

```
Router#show running-config formal | include hw-module
Building configuration...
hw-module fault-recovery location 0/FC0 count 1
```

The following system logs are generated when the number of fault recovery attempts on the card exceeds the configured count:

```
RP/0/RP0/CPU0:Dec 4 15:44:22.950 PST: shelfmgr[121]:
%PLATFORM-SHELFMGR-2-FAULT_ACTION_CARD_SHUTDOWN : Forced shutdown requested for card 0/FC0.
Reason Fault retry attempts exceeded configured count(1)

RP/0/RP0/CPU0:Dec 4 15:44:25.247 PST: shelfmgr[121]: %PLATFORM-SHELFMGR-4-CARD_SHUTDOWN :
Shutting down 0/FC0: Fault retry attempts exceeded configured count(1)
```

Use the **show reboot history** command to get the reason of card shutting down. In the following example, it shows that the card was shut down due to **Fault retry attempts exceeded configured count(1)**.

```
RP/0/RP0/CPU0:ios#show reboot history location 0/FC0 detail
Mon Dec 4 15:44:55.827 PST
```

```
-----
No   Attribute      Value
-----
1    Time (PST)      Dec 04 2023 15:44:22
     Cause Code     0x0800000d
     Cause String   REBOOT_CAUSE_FM
     Graceful Reload No
     Kdump Requested No
     Reason         Fault retry attempts exceeded configured count(1)
```

Use the **show platform** command to see the current state of the card that was shut down because of Fault recovery handling feature.

```
RP/0/RP0/CPU0:ios#show platform
Mon Oct 2 21:08:03.383 UTC
```

```
-----
Node           Type                               State                               Config state
-----
0/RP0/CPU0     8800-RP (Active)                   IOS XR RUN                          NSHUT
0/RP0/BMC0     8800-RP                             OPERATIONAL                          NSHUT
0/RP1/CPU0     8800-RP (Standby)                   IOS XR RUN                          NSHUT
0/RP1/BMC0     8800-RP                             OPERATIONAL                          NSHUT
0/3/CPU0       8800-LC-48H                         IOS XR RUN                          NSHUT
0/FC0         8812-FC                           SHUT DOWN                          NSHUT
0/FC3         8812-FC                             OPERATIONAL                          NSHUT
0/FT0         SF-D-12-FAN                         OPERATIONAL                          NSHUT
0/FT1         SF-D-12-FAN                         OPERATIONAL                          NSHUT
0/FT2         SF-D-12-FAN                         OPERATIONAL                          NSHUT
0/FT3         SF-D-12-FAN                         OPERATIONAL                          NSHUT
0/PT0         FAM7000-ACHV-TRAY                   OPERATIONAL                          NSHUT
0/PT1         FAM7000-ACHV-TRAY                   OPERATIONAL                          NSHUT
0/PT2         FAM7000-ACHV-TRAY                   OPERATIONAL                          NSHUT
Router#
```

Periodic syslog messages for shutdowns due to fault-recovery failures

Table 108: Feature History Table

Feature Name	Release Information	Feature Description
Periodic syslog messages for shutdowns due to fault-recovery failures	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8712-MOD-M • 8011-4G24Y4H-I
Periodic syslog messages for shutdowns due to fault-recovery failures	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200 [ASIC: P100], 8700 [ASIC: P100])(select variants only*); Centralized Systems (8600); Modular Systems (8800 [LC ASIC: Q100, Q200, P100])(select variants only*)</p> <p>Cisco IOS XR Software now generates a syslog message immediately to indicate its shutdown state after a Line Card (LC), Fabric Card (FC), or Route Processor (RP) shuts down due to fault-recovery failure. This syslog message is repeated every 60 minutes to keep you informed of the shutdown status.</p> <p>This enhancement helps in identifying and troubleshooting shutdown LC, FC, or RP components.</p> <p>*This feature is now supported on:</p> <ul style="list-style-type: none"> • 8212-48FH-M • 8711-32FH-M • 88-LC1-12TH24FH-E • 88-LC1-36EH • 88-LC1-52Y8H-EM

A periodic shutdown syslog message is a log message generated by the router when

- the LC, FC, or RP experiences a fault,
- the Cisco IOS XR software triggers the fault recovery cycle, attempting to reboot the LC, FC, or RP to restore operational status, and
- if the LC, FC, or RP fails to become operational after this recovery attempt, the Cisco IOS XR software proceeds to shut down the affected component and generates a shutdown syslog message immediately following the shutdown.

By default, the Cisco IOS XR software performs the fault recovery cycle five times before shutting down the LC, FC, or RP. If the fault recovery handling count is configured, the Cisco IOS XR software shuts down the LC, FC, or RP after the expiry of the fault recovery count. For more information, see [Fault recovery handling, on page 487](#).

Before Release 24.4.1, the Cisco IOS XR software generates a shutdown syslog message only once immediately after the LC, FC, or RP shut down to notify you of the shutdown.

From Release 24.4.1 onwards, the Cisco IOS XR software generates the following shutdown syslog message immediately after the LC, FC, or RP shuts down and repeats the shutdown syslog message every 60 minutes to notify you of the shutdown until you manually shut down the LC, FC, or RP using the **hw-module shutdown location** or **reload location** commands.

```
Router: Dec 4 15:44:22.950 PST: shelfmgr[121]: %PLATFORM-SHELFMGR-2-FAULT_ACTION_CARD_SHUTDOWN
: Forced shutdown requested for card 0/FC0. Reason Fault retry attempts exceeded configured
count(1)
Router:Dec 4 15:44:25.247 PST: shelfmgr[121]: %PLATFORM-SHELFMGR-4-CARD_SHUTDOWN : Shutting
down 0/FC0: Fault retry attempts exceeded configured count(1)
```

Limitations and restrictions for periodic shutdown syslog messages

When you manually shut down a specific node using the **shutdown location** command in XR EXEC mode or the **hw-module shutdown location** command in XR Config mode, the Cisco IOS XR software doesn't generate the shutdown syslog messages.

Machine check error notifications

Table 109: Feature History Table

Feature Name	Release Information	Feature Description
Machine check error notifications	Release 25.1.1	<p>Introduced in this release on: Fixed Systems (8700 [ASIC: K100], 8010 [ASIC: A100])(select variants only*)</p> <p>*This feature is supported on:</p> <ul style="list-style-type: none"> • 8712-MOD-M • 8011-4G24Y4H-I

Feature Name	Release Information	Feature Description
Machine check error notifications	Release 24.4.1	<p>Introduced in this release on: Fixed Systems (8200, 8700); Centralized Systems (8600); Modular Systems (8800 [LC ASIC: Q100, Q200, P100])</p> <p>You can now identify and resolve MCE-related issues quickly and easily because Cisco IOS XR Software displays a syslog notification for MCE errors, eliminating the need to manually check for them in the MCE log file.</p>

Machine Check Errors (MCE) in routers occur when the system's processors detect hardware errors.

Various hardware failures, such as issues with memory, CPUs, power, or other critical components, can cause these errors.

When a MCE occurs, the router logs a System Error Message (SEM) in `/var/log/mcelog.log` and may restart the affected Line Card (LC), Route Processor (RP), or the entire router as a corrective action.

Before Release 24.4.1, you must manually check the MCE error logs in the location `/var/log/mcelog.log` or on the syslog server to determine whether the router reboot was due to a MCE or another issue.

From Release 24.4.1 onwards, the Cisco IOS XR Software logs the error in the MCE log file and notifies you by displaying a syslog message.

This is an example of an MCE that the router displays:

```
RP/0/RP0/CPU0:Oct 28 22:37:44.293 UTC: shelfmgr[377]: %PLATFORM-CPA_INTF_SHELFMGR-3-CPU_MCERR
: CPU Machine Check Error condition reported for node0_RP0_CPU0: corrected DIMM memory
error count exceeded threshold: 10 in 24h . Reported at 2024-10-28 22:37:44.00000 UTC
```

Syslog message information

The syslog message displays the following information about the error:

- **Error title** - CPA_INTF_SHELFMGR-3-CPU_MCERR
- **Error description** - CPU Machine Check Error
- **Error location** - RP/0/RP0/CPU0
- **Error type** - DIMM memory error
- **Error time** - 2024-10-28 22:37:44.00000 UTC

Error detail and recommended action

- **Cisco feature navigator error messages tool** - Provides detailed error information and recommended actions. For more information, see [Viewing error details in the cisco feature navigator error messages tool, on page 493](#).

- **MCE log file** - Stores all past errors in the MCE log file located at `/var/log/mcelog.log`. You can determine if the current error has occurred in the past using the MCE log file and troubleshoot accordingly. For more information, see [Viewing error details in the MCE log file, on page 494](#)

MCE Major Errors in a Router

These are some of the MCE major errors that occurs in a router:

- **Card power zone error**: Displays under voltage or over voltage failure condition on the Line Card (LC) or Fabric Card (FC). During such an error, the system will attempt to recover by power-cycling the LC or FC.
- **Single Event Upset (SEU) error**: Displays corrected and uncorrected SEU events that can happen in FPGA devices.
- **Central Processing Unit (CPU) error**: Displays all CPU errors.

If these errors occur in a router, you can see the occurrence of these errors using the **show alarms** command. For more information, see [Monitoring Alarms and Implementing Alarm Log Correlation](#) section in the *System Monitoring Configuration Guide for Cisco 8000 Series Routers*.

Limitations and restrictions for MCE major errors

From Release 24.2.11, **show alarm** command output includes only the power zone errors.

Viewing error details in the cisco feature navigator error messages tool

Perform these steps to see error details in the cisco feature navigator error messages tool:

Procedure

- Step 1** Login to [Cisco Feature Navigator Error Messages Tool](#).
The cisco feature navigator error messages tool provides these search options:
- **Release** - Displays error details based on specific Cisco IOS XR Release.
 - **Error** - Displays the error details based on the provided error title.
 - **Compare** - Displays the error details by comparing different Cisco IOS XR Releases.
- Step 2** Click on **Error** option.
- Step 3** Enter the error title, for example, CPA_INTF_SHELFMGR-3-CPU_MCERR.
- Step 4** Click **Submit** to view the error details.
- The error details contain these sections:
- Error
 - Severity
 - Limit

- Format
- Explanation
- Recommended action

For more information about error details sections and Cisco Feature Navigator Error Messages Tool, see [Cisco IOS XR System Error Message Reference Guide](#).

Viewing error details in the MCE log file

Perform these steps to see error details in the MCE log file:

Procedure

- Step 1** Navigate to MCE log file located at `/var/log/mcelog.log`.
- Step 2** Open `mcelog.log` file to view the error details.

Guidelines for Online Insertion and Removal on Cisco 8700 Series routers

These guidelines apply for the Online Insertion and Removal (OIR) of the optical modules on these Cisco 8700 Series routers.

- Cisco 8711-48Z-M
- Cisco 8712-MOD-M router with 8K-MPA-18Z1D MPA

Guidelines for re-inserting optics

- After removing certain Cisco 1G Bidirectional optics, 1G Coarse Wavelength Division Multiplexing (CWDM) optics, or 10G Bidirectional SFP optics, wait for at least 15 minutes before re-inserting the same optics into any SFP port.
- The 15 minute wait time also applies to all third-party 1G and 10G optics, as their behavior is not verified by Cisco.
- This wait time does not apply when installing new or unused optics.

The wait time applies to these optics:

Optics Type	PID
Cisco 1G Bidirectional Optics	<ul style="list-style-type: none"> • GLC-BX40-DA-I • GLC-BX40-D-I

Optics Type	PID
	<ul style="list-style-type: none"> • GLC-BX40-U-I • GLC-BX80-D-I • GLC-BX80-U-I <p>For more details, see the Data sheet.</p>
Cisco 1G CWDM Optics	<p>CWDM-SFP-xxxx</p> <p>For more details, see the Data sheet.</p>
Cisco 10G Bidirectional Optics	<ul style="list-style-type: none"> • SFP-10G-BXD-I • SFP-10G-BXU-I • SFP-10G-BX40U-I • SFP-10G-BX40D-I <p>For more details, see the Data sheet.</p>
Third-party 1G and 10G optics	NA

Wait time guidelines applicable to Cisco 8711-48Z-M router

- The 48 SFP56 ports are divided into four groups:
 - Group 1: Ports 0–11
 - Group 2: Ports 12–23
 - Group 3: Ports 34–45
 - Group 4: Ports 46–57
- Group 5: Ports 24-33, includes four QSFP56 and six QSFP-DD ports.
- If the same optics are re-inserted on the router within 15 minutes, a brief disruption or link flap may occur on the remaining 11 SFP56 ports of the same group. Other groups remain unaffected.
- Inserting optics into Group-5 ports does not cause any disruptions.

Wait time guidelines applicable to Cisco 8712-MOD-M router with 8K-MPA-18Z1D MPA

- The 19 ports (18 SFP56 + 1 QSFP56-DD) are divided into two groups:
 - Group 1: Ports 0–8
 - Group 2: Ports 9–18
- If the same optics are re-inserted on the router within 15 minutes, a brief disruption or link flap may occur on the remaining ports of the same group. Other groups remain unaffected and ports on other MPAs remain unaffected.

- Inserting optics into QSFP28, QSFP56, or QSFP-DD ports does not cause any link disruptions.