cisco.



Interface and Hardware Component Configuration Guide for Cisco ASR 9000 Series Routers, IOS XR Release 7.10.x

First Published: 2023-06-30

Americas Headquarters

Cisco Systems, Inc. 170 West Tasman Drive San Jose, CA 95134-1706 USA http://www.cisco.com Tel: 408 526-4000 800 553-NETS (6387) Fax: 408 527-0883 THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

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

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

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT 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)

© 2023 Cisco Systems, Inc. All rights reserved.



CONTENTS

I

P R E F A C E	Preface xxi Changes to This Document xxi Obtaining Documentation and Submitting a Service Request xxi
CHAPTER 1	New and Changed Feature Information 1
	Interface and Hardware Component Features Added or Modified in IOS XR Release 7.10.x 1
CHAPTER 2	YANG Data Models for Interfaces and Hardware Component Features 3
	Using YANG Data Models 3
CHAPTER 3	Preconfiguring Physical Interfaces 5
	Preconfiguring Physical Interfaces 6
	Prerequisites for Preconfiguring Physical Interfaces 7
	Information About Preconfiguring Physical Interfaces 7
	Physical Interface Preconfiguration Overview 7
	Benefits of Interface Preconfiguration 8
	Use of the Interface Preconfigure Command 8
	Active and Standby RSPs and Virtual Interface Configuration 8
	How to Preconfigure Physical Interfaces 9
	Configuration Examples for Preconfiguring Physical Interfaces 11
	Preconfiguring an Interface: Example 11
CHAPTER 4	Advanced Configuration and Modification of the Management Ethernet Interface 13
	Advanced Configuration and Modification of the Management Ethernet Interface 14
	Prerequisites for Configuring Management Ethernet Interfaces 15
	Information About Configuring Management Ethernet Interfaces 15

Default Interface Settings 15

How to Perform Advanced Management Ethernet Interface Configuration 16
Configuring a Management Ethernet Interface 16
IPv6 Stateless Address Auto Configuration on Management Interface 18
Configuring the Duplex Mode for a Management Ethernet Interface 20
Configuring the Speed for a Management Ethernet Interface 21
Modifying the MAC Address for a Management Ethernet Interface 22
Verifying Management Ethernet Interface Configuration 24
Configuration Examples for Management Ethernet Interfaces 24
Configuring a Management Ethernet Interface: Example 24

CHAPTER 5 Configuring Ethernet Interfaces 27

Configuring Ethernet Interfaces 29 Prerequisites for Configuring Ethernet Interfaces 31 Information About Configuring Ethernet 32 16-Port 10-Gigabit Ethernet SFP+ Line Card 32 Features 32 Cisco ASR 9000 Modular Line Cards 33 Restrictions on Module Port Adaptors 33 Default Configuration Values for Gigabit Ethernet and 10-Gigabit Ethernet 34 Default Configuration Values for Fast Ethernet 34 Layer 2 VPN on Ethernet Interfaces 35 Gigabit Ethernet Protocol Standards Overview 36 IEEE 802.3 Physical Ethernet Infrastructure 36 IEEE 802.3ab 1000BASE-T Gigabit Ethernet 36 IEEE 802.3z 1000 Mbps Gigabit Ethernet 36 IEEE 802.3ae 10 Gbps Ethernet **36** IEEE 802.3ba 100 Gbps Ethernet 37 MAC Address 37 MAC Accounting 37 Ethernet MTU 37 Flow Control on Ethernet Interfaces 38 802.1Q VLAN 38 VRRP 38

HSRP 38 Link Autonegotiation on Ethernet Interfaces 39 Fast Polling for WAN-PHY 40 Early Indication of Link Loss Change 40 Subinterfaces on the Cisco ASR 9000 Series Router 41 Layer 2, Layer 3, and EFP's 44 Enhanced Performance Monitoring for Layer 2 Subinterfaces (EFPs) 46 Other Performance Management Enhancements 47 Frequency Synchronization and SyncE 47 LLDP 47 LLDP Frame Format 48 LLDP TLV Format 48 LLDP Operation 48 Supported LLDP Functions 49 Unsupported LLDP Functions 50 Enabling LLDP Per Interface 50 Unidirectional Link Routing 51 How to Configure Ethernet 51 Configuring Ethernet Interfaces 52 Configuring Gigabit Ethernet Interfaces 52 Configuring a Fast Ethernet Interface 55 Configuring MAC Accounting on an Ethernet Interface 55 Configuring a L2VPN Ethernet Port 57 Configuring LLDP 59 LLDP Default Configuration 59 Enabling LLDP Globally 60 Configuring Global LLDP Operational Characteristics 61 Disabling Transmission of Optional LLDP TLVs 62 Disabling LLDP Receive and Transmit Operation for an Interface 63 Verifying the LLDP Configuration **65** Verifying the LLDP Global Configuration 65 Verifying the LLDP Interface Configuration 65 Configuring UDLR 66 Configuring the Dual-Rate Line Cards 67

Convert Speed of a Dual-mode Optic 68
Creating Slices on a Router 70
Overview 70
Slice and Port Numbering 70
Configure Slices 72
Configuration Examples for Ethernet 74
Configuring an Ethernet Interface: Example 75
Configuring MAC-Accounting: Example 75
Configuring a Layer 2 VPN AC: Example 76
Configuring LLDP: Examples 76
How to Configure Interfaces in Breakout Mode 77
Information About Breakout 77
Breakout Configuration: Examples 78
Configure 4x25GbE Breakout in a Port 78
Configure 4x100GbE Breakout in a Port 78

CHAPTER 6 Configuring Ethernet OAM 81

Prerequisites for Configuring Ethernet OAM 83
Information About Configuring Ethernet OAM 83
Ethernet Link OAM 84
Neighbor Discovery 84
Link Monitoring 85
MIB Retrieval 85
Miswiring Detection (Cisco-Proprietary) 85
Remote Loopback 85
SNMP Traps 85
Unidirectional Link Fault Detection 85
Ethernet CFM 85
Maintenance Domains 86
Services 88
Maintenance Points 88
MIP Creation 89
MEP and CFM Processing Overview 89
CFM Protocol Messages 91

Continuity Check (IEEE 802.1ag and ITU-T Y.1731) 91 Loopback (IEEE 802.1ag and ITU-T Y.1731) 93 Linktrace (IEEE 802.1ag and ITU-T Y.1731) 94 Exploratory Linktrace (Cisco) 95 Delay and Jitter Measurement (ITU-T Y.1731) 96 Synthetic Loss Measurement (ITU-T Y.1731) 97 Loss Measurement (ITU-T Y.1731) 97 MEP Cross-Check 97 Configurable Logging 97 EFD 98 Flexible VLAN Tagging for CFM 99 CFM on MC-LAG 100 RG-Level CFM 100 POA-Level CFM 101 Supported Features for CFM on MC-LAG **101** Restrictions for CFM on MC-LAG **102** CFM Software Acceleration **102** Connectivity Fault Management Hardware Offload 103 Ethernet SLA 105 Y.1731 Performance Monitoring 106 Loss Measurement Terminology 107 Loss Measurement Performance Attributes 107 Limitations of Data Loss Measurement 108 Ethernet SLA Concepts 108 Loss Measurement Terminology 108 Ethernet SLA Measurement Packet 109 Ethernet SLA Sample 110 Ethernet SLA Probe 110 Ethernet SLA Burst 110 Ethernet SLA Schedule 110 Ethernet SLA Bucket 110 Ethernet SLA Aggregation Bin 111 Ethernet SLA Operation Profile 111 Ethernet SLA Operation 111

Ethernet SLA On-Demand Operation 111 Statistics Measurement and Ethernet SLA Operations Overview 112 Configuration Overview of Scheduled Ethernet SLA Operations 113 Ethernet LMI 113 E-LMI Messaging 114 Cisco-Proprietary Remote UNI Details Information Element 115 E-LMI Operation 115 Supported E-LMI PE Functions on the Cisco ASR 9000 Series Router 115 Unsupported E-LMI Functions 116 Unidirectional Link Detection Protocol **116** UDLD Operation 117 Types of Fault Detection 117 UDLD Modes of Operation 117 UDLD Aging Mechanism 118 State Machines 118 Main FSM 118 Detection FSM 118 Ethernet Data Plane Loopback 119 Features Supported for Ethernet Data Plane Loopback 119 Limitations of Ethernet Data Plane Loopback 120 Configuring Ethernet Data Plane Loopback 120 Ethernet Data Plane Loopback on Satellite nV System 122 How to Configure Ethernet OAM 125 Configuring Ethernet Link OAM 125 Configuring an Ethernet Link OAM Profile 125 Attaching an Ethernet OAM Profile to an Interface 131 Configuring Ethernet OAM at an Interface and Overriding the Profile Configuration 132 Verifying the Ethernet OAM Configuration 134 Configuring Ethernet CFM 135 Configuring a CFM Maintenance Domain 135 Configuring Services for a CFM Maintenance Domain 136 Enabling and Configuring Continuity Check for a CFM Service 138 Configuring Automatic MIP Creation for a CFM Service 140 Configuring Cross-Check on a MEP for a CFM Service 141

Configuring Other Options for a CFM Service 143 Configuring CFM MEPs 145 Configuring Y.1731 AIS 148 Configuring AIS in a CFM Domain Service 148 Configuring AIS on a CFM Interface **150** Configuring EFD for a CFM Service 151 Verifying the EFD Configuration 153 Configuring Flexible VLAN Tagging for CFM 153 Verifying the CFM Configuration 155 Troubleshooting Tips 155 Configuring Ethernet SLA 156 Ethernet SLA Configuration Guidelines 156 Configuring an SLA Operation Profile 157 Configuring a Schedule for an SLA Operation Probe in a Profile **158** Configuring SLA Probe Parameters in a Profile 159 Configuring SLA Statistics Measurement in a Profile 161 Configuring an SLA Operation 164 Configuring an On-Demand SLA Operation 166 Configuration Guidelines 166 Configuring an On-Demand Ethernet SLA Operation for CFM Delay Measurement **167** Configuring an On-Demand Ethernet SLA Operation for CFM Loopback 167 Configuring an On-Demand Ethernet SLA Operation for CFM Synthetic Loss Measurement 168 Verifying SLA Configuration 168 Bit Error Rate 169 Configuring Ethernet LMI 174 Prerequisites for Configuring E-LMI 174 Restrictions for Configuring E-LMI 174 Creating EVCs for E-LMI 174 Configuring EFPs 174 Configuring a Bridge Group and Assigning EFPs to a Bridge Domain 176 Configuring Ethernet CFM for E-LMI **177** Configuring Ethernet CFM 177 Configuring EFPs as CFM Up MEPs 177 Configuring UNI Names on the Physical Interface 179

Enabling E-LMI on the Physical Interface 180 Configuring the Polling Verification Timer 181 Configuring the Status Counter 183 Disabling Syslog Messages for E-LMI Errors or Events 184 Disabling Use of the Cisco-Proprietary Remote UNI Details Information Element **186** Verifying the Ethernet LMI Configuration 187 Troubleshooting Tips for E-LMI Configuration 188 Ethernet LMI Link Status Troubleshooting 188 Ethernet LMI Line Protocol State Troubleshooting 189 Ethernet LMI Error Counter Troubleshooting 189 Ethernet LMI Remote UNI Troubleshooting 190 Configuring UDLD 190 191 Configuration Examples for Ethernet OAM Configuration Examples for EOAM Interfaces 191 Configuring an Ethernet OAM Profile Globally: Example 192 Configuring Ethernet OAM Features on an Individual Interface: Example 192 Configuring Ethernet OAM Features to Override the Profile on an Individual Interface: Example 193 Recovering from error-disable: Example 193 Configuring a Remote Loopback on an Ethernet OAM Peer: Example 194 Clearing Ethernet OAM Statistics on an Interface: Example 194 Enabling SNMP Server Traps on a Router: Example 194 Configuration Examples for Ethernet CFM 194 Ethernet CFM Domain Configuration: Example 194 Ethernet CFM Service Configuration: Example 194 Flexible Tagging for an Ethernet CFM Service Configuration: Example 194 Continuity Check for an Ethernet CFM Service Configuration: Example 195 MIP Creation for an Ethernet CFM Service Configuration: Example 195 Cross-check for an Ethernet CFM Service Configuration: Example 195 Other Ethernet CFM Service Parameter Configuration: Example 195 MEP Configuration: Example 195 Ethernet CFM Show Command: Examples 195 AIS for CFM Configuration: Examples 199 AIS for CFM Show Commands: Examples 200

show ethernet cfm interfaces ais Command: Example 200
show ethernet cfm local meps Command: Examples 200
EFD Configuration: Examples 202
Displaying EFD Information: Examples 203
show efd interfaces Command: Example 203
show ethernet cfm local meps detail Command: Example 203
Configuration Examples for Ethernet SLA 204
Ethernet SLA Profile Type Configuration: Examples 204
Ethernet SLA Probe Configuration: Examples 204
Profile Statistics Measurement Configuration: Examples 205
Scheduled SLA Operation Probe Configuration: Examples 206
Ethernet SLA Operation Probe Scheduling and Aggregation Configuration: Example 206
Ongoing Ethernet SLA Operation Configuration: Example 207
On-Demand Ethernet SLA Operation Basic Configuration: Examples 208
Ethernet SLA Y.1731 SLM Configuration: Examples 208
Ethernet SLA Show Commands: Examples 209
Configuration Example for Ethernet LMI 212
Configuration Examples for Ethernet Data Plane Loopback 214
Verification 214
CFM Adaptive Bandwidth Notifications 216
Bandwidth Notification Messages 216
Restrictions for CFM Bandwidth Notifications 217
Bandwidth Reporting 218
Damping Algorithm 218
Conformance Testing Algorithm 220
Embedded Event Manager 220
Event Publishing 221
Configuring CFM Bandwidth Notifications 222
Configuring Integrated Routing and Bridging 225
Prerequisites for Configuring IRB 226
Guidelines and Restrictions for Configuring IRB 227

Information About Configuring IRB **228**

IRB Introduction 228

CHAPTER 7

Bridge-Group Virtual Interface 229 Bridge-Group Virtual Interface 229 Supported Features on a BVI 230 BVI MAC Address 230 BVI Interface and Line Protocol States 230 Packet Flows Using IRB 231 Packet Flows When Host A Sends to Host B on the Bridge Domain 232 Packet Flows When Host A Sends to Host C From the Bridge Domain to a Routed Interface 232 Packet Flows When Host C Sends to Host B From a Routed Interface to the Bridge Domain 232 Supported Environments for IRB 233 Additional IPv4-Specific Environments Supported for IRB 233 Additional IPv6-Specific Environments Supported for IRB 234 How to Configure IRB 234 Configuring the Bridge Group Virtual Interface 234 Configuration Guidelines 235 Configuring the Layer 2 AC Interfaces 237 Configuring a Bridge Group and Assigning Interfaces to a Bridge Domain 238 Associating the BVI as the Routed Interface on a Bridge Domain 240 Displaying Information About a BVI 241 Configuration Examples for IRB 242 Basic IRB Configuration: Example 242 IRB Using ACs With VLANs: Example 242 IPv4 Addressing on a BVI Supporting Multiple IP Networks: Example 243 Comprehensive IRB Configuration with BVI Bundle Interfaces and Multicast Configuration: Example 243 IRB With BVI and VRRP Configuration: Example 244 6PE/6VPE With BVI Configuration: Example 245 Configuring Link Bundling 249 Prerequisites for Configuring Link Bundling 250

Prerequisites for Configuring Link Bundling on a Cisco ASR 9000 Series Router 250 Information About Configuring Link Bundling 251 Link Bundling Overview 251 Features and Compatible Characteristics of Ethernet Link Bundles 252

CHAPTER 8

Characteristics of POS Link Bundles in Cisco ASR 9000 Series Router 253 Restrictions of POS Link Bundles in Cisco ASR 9000 Series Router 254 Link Aggregation Through LACP 254 IEEE 802.3ad Standard 254 Non Revertive Behavior for LACP Bundle Interface 255 Multichassis Link Aggregation 255 Failure Cases 256 Interchassis Communication Protocol 256 Access Network Redundancy Model 257 ICCP Based Service Multihoming 258 ICCP-SM Redundancy Group 258 Advantages of Pseudo mLACP: 259 Failure Modes 259 Core Network Redundancy Model 260 **One-way Pseudowire Redundancy** 260 Two-way Pseudowire Redundancy 260 Switchovers 261 Dynamic Priority Management 261 Brute Force Behavior 262 MC-LAG Topologies 262 LACP Short Period Time Intervals 263 Load Balancing 263 Layer 2 Ingress Load Balancing on Link Bundles 263 Layer 3 Egress Load Balancing on Link Bundles 264 Layer 3 Load Balancing Before Cisco IOS XR Release 4.0.1 264 Layer 3 Load Balancing From Cisco IOS XR Release 4.0.1 264 Dynamic Load Balancing for LAG 265 QoS and Link Bundling 265 VLANs on an Ethernet Link Bundle 266 Link Bundle Configuration Overview 266 Nonstop Forwarding During Card Failover 267 Link Failover 267 Multi-Gigabit Service Control Point 267 How to Configure Link Bundling 268

Configuring Ethernet Link Bundles 268 Configuring EFP Load Balancing on an Ethernet Link Bundle 272 Configuring VLAN Bundles 274 274 Configuring POS Link Bundles 278 Configuring Multichassis Link Aggregation 281 Configuring Interchassis Communication Protocol 281 Configuring Multichassis Link Aggregation Control Protocol Session 283 Configuring Multichassis Link Aggregation Control Protocol Bundle 285 Configuring Dual-Homed Device 287 Configuring One-way Pseudowire Redundancy in MC-LAG 289 Configuring VPWS Cross-Connects in MC-LAG 290 Configuring ICCP based Service Homing 293 Configuring VPLS in MC-LAG 295 How to Configure MGSCP 297 Prerequisites for Configuring MGSCP 297 **Restrictions for Configuring MGSCP** 298 Configuring the Access Bundle for the Subscriber-Facing Side 298 Configuring the Network Bundle for the Core-Facing Side 300 Configuring the Bundle Member Interfaces 302 Configuring VRFs to Route Traffic to the Bundles 304 Configuring VRFs with Static Routing 304 Configuring VRFs with Dynamic Routing 304 Configuration Examples for Link Bundling **305** Example: Configuring an Ethernet Link Bundle 305 Example: Configuring a VLAN Link Bundle 307 Example: Configuring a POS Link Bundle 307 Example: Configuring EFP Load Balancing on an Ethernet Link Bundle 307 Example: Configuring Multichassis Link Aggregation 308 Configuration Examples for MGSCP 312 Example: Configuring Bundle Interfaces and Member Links 313 Examples: Configuring VRFs to Route Traffic to the Bundles 314 Example: Configuring VRFs with Static Routing 314 Example: Configuring VRFs with OSPF Routing 315

	Example: Configuring MGSCP with ABF to Route Traffic to the Bundles 316
9	Configuring Traffic Mirroring 319
	Introduction to Traffic Mirroring 319
	Sampled Traffic Mirroring 321
	Implementing Traffic Mirroring on the Cisco ASR 9000 Series Router 321
	Traffic Mirroring Terminology 321
	Characteristics of the Source Port 322
	Characteristics of the Monitor Session 322
	Characteristics of the Destination Port 323
	Supported Traffic Mirroring Types 324
	Pseudowire Traffic Mirroring 324
	ACL-Based Traffic Mirroring 325
	Restrictions for Traffic Mirroring 325
	Restrictions of Sampled Traffic Mirroring 326
	Performance Impact with Traffic Mirroring 327
	Configuring Traffic Mirroring 327
	How to Configure Local Traffic Mirroring 327
	How to Configure Remote Traffic Mirroring 329
	How to Configure Traffic Mirroring over Pseudowire 331
	How to Configure ACL-Based Traffic Mirroring 335
	Troubleshooting ACL-Based Traffic Mirroring 338
	How to Configure Partial Packet Mirroring 338
	Traffic Mirroring Configuration Examples 340
	Traffic Mirroring with Physical Interfaces (Local): Example 340
	Traffic Mirroring with EFPs (Remote): Example 341
	Traffic Mirroring with Bundle Interfaces: Example 341
	Viewing Monitor Session Status: Example 342
	Monitor Session Statistics: Example 343
	Traffic Mirroring over Pseudowire: Example 343
	Layer 3 ACL-Based Traffic Mirroring: Example 343
	Layer 2 ACL-Based Traffic Mirroring: Example 344
	Partial Packet Mirroring: Example 344
	Sampled Traffic Mirroring: Example 344

I

CHAPTER

I

	Troubleshooting Traffic Mirroring 345
	SPAN to File 348
	Guidelines and Restrictions for SPAN to File 349
	Action Commands for SPAN to File 349
	Configuring SPAN to File 350
	Introduction to File Mirroring 352
	Limitations 352
	Configure File Mirroring 353
CHAPTER 10	Configuring Virtual Loopback and Null Interfaces 355
	Prerequisites for Configuring Virtual Interfaces 355
	Information About Configuring Virtual Interfaces 355
	Virtual Loopback Interface Overview 356
	Null Interface Overview 356
	Virtual Management Interface Overview 356
	Active and Standby RPs and Virtual Interface Configuration 357
	How to Configure Virtual Interfaces 357
	Configuring Virtual Loopback Interfaces 357
	Configuring Null Interfaces 359
	Configuring Virtual IPv4 Interfaces 360
	Configuration Examples for Virtual Interfaces 361
	Configuring a Loopback Interface: Example 361
	Configuring a Null Interface: Example 361
	Configuring a Virtual IPv4 Interface: Example 362
CHAPTER 11	Configuring Dense Wavelength Division Multiplexing Controllers 363
	Configuring Dense Wavelength Division Multiplexing Controllers 364
	Prerequisites for Configuring DWDM Controller Interfaces 365
	Information About the DWDM Controllers 365
	CFP2 DCO Optics Version Compatibility 366
	Information about IPoDWDM 367
	How to Configure DWDM Controllers 368
	Configuring the Optical Parameters 368

Configuring G.709 Parameters **371** How to Perform Performance Monitoring on DWDM Controllers 373 Configuring DWDM Controller Performance Monitoring 374 Configuring IPoDWDM 378 Configuring the Optical Layer DWDM Ports 378 Configuring the Administrative State of DWDM Optical Ports 379 Configuring Proactive FEC-FRR Triggering 381 Configuration Examples 383 Turning On the Laser: Example 383 Turning Off the Laser: Example 383 **DWDM** Controller Configuration: Examples 383 **DWDM Performance Monitoring: Examples** 384 IPoDWDM Configuration: Examples 385 Optical Layer DWDM Port Configuration: Examples 385 Administrative State of DWDM Optical Ports Configuration: Examples 385 Proactive FEC-FRR Triggering Configuration: Examples 385

387

Configuring DAC Rate **394** Configuring Breakout **396** Configuring FEC **398** Configuring Frequency **400** Configuring Chromatic Dispersion **401** Configuring Optical Transmit Power **403** Configuring Performance Monitoring **406**

CHAPTER 12

Configuring PM Parameters 406

Configuring 400G Digital Coherent Optics

Configuring Loopback 409

Laser Squelching 411

Small Frame Padding 412

Configuring Alarms Threshold 413

CHAPTER 13 Configuring 802.1Q VLAN Interfaces 417

Prerequisites for Configuring 802.1Q VLAN Interfaces 417 Information About Configuring 802.1Q VLAN Interfaces 418

802.1Q VLAN Overview 418 CFM on 802.1Q VLAN Interfaces 418 Subinterfaces 418 Subinterface MTU 419 Native VLAN 419 **EFPs** 419 Layer 2 VPN on VLANs 419 Other Layer 2 VPN Features 420 How to Configure 802.1Q VLAN Interfaces 420 Configuring 802.1Q VLAN Subinterfaces 420 Configuring an Attachment Circuit on a VLAN 423 Removing an 802.1Q VLAN Subinterface 425 Configuration Examples for VLAN Interfaces 427 VLAN Subinterfaces: Example 427

CHAPTER 14 Managing Router Hardware 431

Prerequisites for Managing Router Hardware 432 Displaying Hardware Status 432 Displaying SDR Hardware Version Information 432 Displaying System Hardware Version Information **435** Displaying Software and Hardware Information 439 Displaying SDR Node IDs and Status 440 Displaying Router Node IDs and Status 441 **Displaying Router Environment Information** 442 Displaying RP Redundancy Status 446 Displaying Field-Programmable Device Compatibility 446 RSP Redundancy and Switchover 448 Establishing RSP Redundancy 448 Determining the Active RP in a Redundant Pair 449 Role of the Standby RSP 450 Summary of Redundancy Commands 450 Automatic Switchover 450 RSP Redundancy During RSP Reload 451 Manual Switchover 451

System Logs during RSP Switchover 452 Communicating with a Standby RP 453 Configuring Breakout on a 400GE Port 453 Reloading, Shutting Down, or Power Cycling a Node 457 Reloading the Active RSP 458 Flash Disk Recovery 459 460 Using Controller Commands to Manage Hardware Components Formatting Hard Drives, Flash Drives, and Other Storage Devices 460 Removing and Replacing Cards 461 Removing Line Cards 461 Replacing a Line Card with the Same Media Type and Port Count 462 Replacing a Line Card with the Same Media Type and a Different Port Count 462 Replacing a Line Card or PLIM with a Different Media Type 463 Proactive Line Card Shut Down 463 Proactive Line Card Shut Down Implementation Consideration 464 Configure Proactive Line Card Shut Down 464 Advanced Power Management 465 Configuring Advance Power Management 466 Overview of Erase and Wipeout Disk Memory 467 Erase Disk Memory 467 How to Erase Disk Memory 467 Wipe Out Disk Memory 468 How to Wipe out Disk Memory 468 Upgrading the CPU Controller Bits 470 Examples 470 Configuring Port Modes 470 Configuring Port Mode in Cisco A9903-8HG-PEC 471 Configuring Port Mode in Cisco ASR 9902 Router 476 Configuring Port Modes in Cisco ASR 9903 Router and Cisco ASR 9000 Series Line Cards 478 Configure Single Feed Power Mode 479 Additional References 480

Contents



Preface

The Cisco ASR 9000 Series Aggregation Services Router Interface and Hardware Component Configuration Guide provides information and procedures related to router interface and hardware configuration.

From Release 6.1.2 onwards, Cisco introduces support for the 64-bit Linux-based IOS XR operating system. Extensive feature parity is maintained between the 32-bit and 64-bit environments. Unless explicitly marked otherwise, the contents of this document are applicable for both the environments. For more details on Cisco IOS XR 64 bit, refer to the Release Notes for Cisco ASR 9000 Series Routers, Release 6.1.2 document.

The preface contains the following sections:

- Changes to This Document, on page xxi
- Obtaining Documentation and Submitting a Service Request, on page xxi

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
August 2023	Initial release of this document

Obtaining Documentation and Submitting a Service Request

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

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



New and Changed Feature Information

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

• Interface and Hardware Component Features Added or Modified in IOS XR Release 7.10.x, on page 1

Interface and Hardware Component Features Added or Modified in IOS XR Release 7.10.x

Feature	Description	Changed in Release	Where Documented
Disable Laser Squelching	This feature was introduced.	Release 7.10.1	Laser Squelching
Support for DP04QSDD-HE0 optical module	This feature was introduced.	Release 7.10.1	Configuring 400G Digital Coherent Optics
Small Frame Padding	This feature was introduced.	Release 7.10.1	Small Frame Padding
Cyclic Redundancy Check based Bit Error Ratio on A9K-20HG-FLEX, A9K-8HG-FLEX, and A9K-4HG-FLEX Line cards	This feature was introduced.	Release 7.10.1	Bit Error Rate

Table 2: New and Changed Features



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

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*.



Preconfiguring Physical Interfaces

This module describes the preconfiguration of physical interfaces on the Cisco ASR 9000 Series Aggregation Services Routers.

Preconfiguration is supported for the following types of interfaces and controllers:

- Gigabit Ethernet
- 10-Gigabit Ethernet
- Management Ethernet
- Packet-over-SONET/SDH (POS)
- Serial
- SONET controllers and channelized SONET controllers

Preconfiguration allows you to configure modular services cards before they are inserted into the router. When the cards are inserted, they are instantly configured.

The preconfiguration information is created in a different system database tree (known as the *preconfiguration directory* on the route switch processor [RSP]), rather than with the regularly configured interfaces.

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



Note Ten GigE interfaces will not show egress statistics when loopback line is configured because the loopback is closed at the interface controller level, before the Network Processor (NP). But on One GigE interfaces the line loopback is closed in the NP.



Note Only physical interfaces can be preconfigured.

Release	Modification
Release 3.7.2	Ethernet interface preconfiguration was introduced.
Release 4.0.0	POS interface preconfiguration was introduced.

Feature History for Preconfiguring Physical Interfaces

- Preconfiguring Physical Interfaces, on page 6
- Prerequisites for Preconfiguring Physical Interfaces, on page 7
- Information About Preconfiguring Physical Interfaces, on page 7
- How to Preconfigure Physical Interfaces, on page 9
- Configuration Examples for Preconfiguring Physical Interfaces, on page 11

Preconfiguring Physical Interfaces

This module describes the preconfiguration of physical interfaces on the Cisco ASR 9000 Series Aggregation Services Routers.

Preconfiguration is supported for the following types of interfaces and controllers:

- Gigabit Ethernet
- 10-Gigabit Ethernet
- Management Ethernet
- Packet-over-SONET/SDH (POS)
- Serial
- SONET controllers and channelized SONET controllers

Preconfiguration allows you to configure modular services cards before they are inserted into the router. When the cards are inserted, they are instantly configured.

The preconfiguration information is created in a different system database tree (known as the *preconfiguration directory* on the route switch processor [RSP]), rather than with the regularly configured interfaces.

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



Note

Ten GigE interfaces will not show egress statistics when loopback line is configured because the loopback is closed at the interface controller level, before the Network Processor (NP). But on One GigE interfaces the line loopback is closed in the NP.



Note Only physical interfaces can be preconfigured.

Release	Modification
Release 3.7.2	Ethernet interface preconfiguration was introduced.
Release 4.0.0	POS interface preconfiguration was introduced.

Feature History for Preconfiguring Physical Interfaces

Prerequisites for Preconfiguring Physical 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 preconfiguring physical interfaces, be sure that the following condition is met:

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

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. Preconfigured interfaces are not verified or applied until the actual interface with the matching location (rack/slot/module) is inserted into the router. When the anticipated modular services card is inserted and the interfaces are created, the precreated configuration information is verified and, if successful, immediately applied to the router's running configuration.



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

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



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

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

Benefits of Interface Preconfiguration

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

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

Use of the Interface Preconfigure Command

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

The **interface preconfigure** command places the router in interface configuration mode. Users should be able to add any possible interface commands. The verifiers registered for the preconfigured interfaces verify the configuration. The preconfiguration is complete when the user enters the **end** command, or any matching exit or global configuration mode command.



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

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

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

Note Specifying an interface name that already exists and is configured (or an abbreviated name like $e^{0/3}/0/0$) is not permitted.

Active and Standby RSPs and Virtual Interface Configuration

The standby RSP is available and in a state in which it can take over the work from the active RSP should that prove necessary. Conditions that necessitate the standby RSP to become the active RSP and assume the active RSP's duties include:

· Failure detection by a watchdog

- · Standby RSP is administratively commanded to take over
- Removal of the active RSP from the chassis

If a second RSP is not present in the chassis while the first is in operation, a second RSP may be inserted and will automatically become the standby RSP. The standby RSP may also be removed from the chassis with no effect on the system other than loss of RSP redundancy.

After failover, the virtual interfaces will all be present on the standby (now active) RSP. Their state and configuration will be unchanged, and there will have been no loss of forwarding (in the case of tunnels) over the interfaces during the failover. The Cisco ASR 9000 Series Router uses nonstop forwarding (NSF) over tunnels through the failover of the host RSP.



Note

The user does 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.

SUMMARY STEPS

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

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/ configure	
Step 2	interface preconfigure type interface-path-id	Enters interface preconfiguration mode for an interface,
	Example:	where <i>type</i> specifies the supported interface type that you want to configure and <i>interface-path-id</i> specifies the

	Command or Action	Purpose
	<pre>RP/0//CPU0:router(config)# interface preconfigure GigabitEthernet 0/1/0/0</pre>	location where the interface will be located in <i>rack/slot/module/port</i> notation.
Step 3	Use one of the following commands: • ipv4 address <i>ip-address subnet-mask</i> • ipv4 address <i>ip-address/prefix</i> Example: RP/0//CPU0:router(config-if-pre)# ipv4 address 192.168.1.2/32	Assigns an IP address and mask to the interface.
Step 4	Configure additional interface parameters, as described in this manual in the configuration chapter that applies to the type of interface that you are configuring.	
Step 5	end or commit best-effort	Saves configuration changes.
	<pre>Example: RP/0//CPU0:router(config-if-pre)# end</pre>	• When you issue the end command, the system prompts you to commit changes: Uncommitted changes found, commit them before exiting (yes/no/cancel)?
	Or RP/0//CPU0:router(config-if-pre)# commit	• Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the commit best-effort command to save the configuration changes to the running configuration file and remain within the configuration session. The commit best-effort command merges the target configuration with the running configuration and commits only valid changes (best effort). Some configuration changes might fail due to semantic errors.
Step 6	show running-config	(Optional) Displays the configuration information currently
	Example:	running on the router.
	RP/0//CPU0:router# show running-config	

Configuration Examples for Preconfiguring Physical Interfaces

This section contains the following example:

Preconfiguring an Interface: Example

The following example shows how to preconfigure a basic Ethernet interface:

```
RP/0//CPU0:router# configure
RP/0//CPU0:router(config)# interface preconfigure GigabitEthernet 0/1/0/0
RP/0//CPU0:router(config-if)# ipv4 address 192.168.1.2/32
RP/0//CPU0:router(config-if)# commit
```



CHAPTER

Advanced Configuration and Modification of the Management Ethernet Interface

This module describes the configuration of Management Ethernet interfaces.

Before you can use Telnet to access the router through the LAN IP address, you must set up a Management Ethernet interface and enable Telnet servers, as described in the Configuring General Router Features module of the Cisco ASR 9000 Series Router Getting Started Guide. This module describes how to modify the default configuration of the Management Ethernet interface after it has been configured, as described in the Cisco ASR 9000 Series Router



Note

In 32-bit IOS XR OS, the management interfaces are available from XR VM. In 64-bit IOS XR OS, the Management ports on the RP/RSP are available as follows:

- MGT LAN 0 is available in XR VM.
- MGT LAN 1 is available in Admin VM.



Note

Forwarding between physical layer interface modules (PLIM) ports and Management Ethernet interface ports is disabled by default. To enable forwarding between PLIM ports and Management Ethernet interface ports, use the **rp mgmtethernet forwarding** command.



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

Feature History for Configuring Management Ethernet Interfaces

Release	Modification
Release 2.0	This feature was introduced on theCisco CRS-1 Router.

Release 3.2	This feature was first supported on the Cisco XR 12000 Series Router
Release 3.7.2	This feature was introduced.

Advanced Configuration and Modification of the Management Ethernet Interface, on page 14

• Prerequisites for Configuring Management Ethernet Interfaces, on page 15

• Information About Configuring Management Ethernet Interfaces, on page 15

• How to Perform Advanced Management Ethernet Interface Configuration, on page 16

• Configuration Examples for Management Ethernet Interfaces, on page 24

Advanced Configuration and Modification of the Management Ethernet Interface

This module describes the configuration of Management Ethernet interfaces.

Before you can use Telnet to access the router through the LAN IP address, you must set up a Management Ethernet interface and enable Telnet servers, as described in the *Configuring General Router Features* module of the *Cisco ASR 9000 Series Router Getting Started Guide*. This module describes how to modify the default configuration of the Management Ethernet interface after it has been configured, as described in the *Cisco ASR 9000 Series Router*



Note

In 32-bit IOS XR OS, the management interfaces are available from XR VM. In 64-bit IOS XR OS, the Management ports on the RP/RSP are available as follows:

- MGT LAN 0 is available in XR VM.
- MGT LAN 1 is available in Admin VM.



Note

Forwarding between physical layer interface modules (PLIM) ports and Management Ethernet interface ports is disabled by default. To enable forwarding between PLIM ports and Management Ethernet interface ports, use the **rp mgmtethernet forwarding** command.



Note

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

Feature History for Configuring Management Ethernet Interfaces

Release	Modification
Release 2.0	This feature was introduced on theCisco CRS-1 Router.
Release 3.2	This feature was first supported on the Cisco XR 12000 Series Router
---------------	--
Release 3.7.2	This feature was introduced.

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

- You have performed the initial configuration of the Management Ethernet interface, as described in the *Configuring General Router Features* module of the *Cisco ASR 9000 Series Router Getting Started Guide*.
- 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.
- You know how to apply the generalized interface name specification rack/slot/module/port.

For further information on interface naming conventions, refer to the *Cisco ASR 9000 Series Router Getting Started Guide*.



Note For transparent switchover, both active and standby Management Ethernet interfaces are expected to be 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:

Default Interface Settings

This table describes the default Management Ethernet interface settings that can be changed by manual configuration. Default settings are not displayed in the **show running-config** command output.

Table 3: Management Ethernet Interface Default Settings

Parameter	Default Value	Configuration File Entry
Speed in Mbps	Speed is autonegotiated.	speed [10 100 1000]
		To return the system to autonegotiate speed, use the no speed [10 100 1000] command.

Parameter	Default Value	Configuration File Entry
Duplex mode	Duplex mode is autonegotiated.	duplex {full half}
		To return the system to autonegotiated duplex operation, use the no duplex {full half} command, as appropriate.
MAC address	MAC address is read from the hardware burned-in address (BIA).	mac-address address To return the device to its default MAC address, use the no mac-address address command.

How to Perform Advanced Management Ethernet Interface Configuration

This section contains the following procedures:

Configuring a Management Ethernet Interface

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

The MTU is not configurable for the Management Ethernet Interface. The default value is 1514 bytes.

Ś

Note

You do not need to perform this task if you have already set up the Management Ethernet interface to enable telnet servers, as described in the "Configuring General Router Features" Features module of the Cisco ASR 9000 Series Router Getting Started Guide.

SUMMARY STEPS

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

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	interface MgmtEth interface-path-id Example:	Enters interface configuration mode and specifies the Ethernet interface name and notation <i>rack/slot/module/port</i> .
	<pre>RP/0/RSP0/CPU0:router(config)# interface MgmtEth 0/RSP0/CPU0/0</pre>	The example indicates port 0 on the RSP card that is installed in slot 0.
Step 3	ipv4 address ip-address mask	Assigns an IP address and subnet mask to the interface.
	Example:	• Replace <i>ip-address</i> with the primary IPv4 address for the interface.
	RP/0/RSP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224	• Replace <i>mask</i> 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 4	mtu bytes	(Optional) Sets the maximum transmission unit (MTU) byte
	Example:	value for the interface. The default is 1514.
	RP/0//CPU0:router(config-if# mtu 1448	• The default is 1514 bytes.
		• The range for the Management Ethernet interface Interface mtu values is 64 to 1514 bytes.
Step 5	no shutdown	Removes the shutdown configuration, which removes the
	Example:	forced administrative down on the interface, enabling it to move to an up or down state.
	<pre>RP/0/RSP0/CPU0:router(config-if)# no shutdown</pre>	
Step 6	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:

	Command or Action	Purpose
	<pre>RP/0/RSP0/CPU0:router(config-if)# end Of</pre>	Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:
	<pre>RP/0/RSP0/CPU0:router(config-if) # commit</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.
Step 7	show interfaces MgmtEth interface-path-id	(Optional) Displays statistics for interfaces on the router.
	Example:	
	<pre>RP/0/RSP0/CPU0:router# show interfaces MgmtEth 0/RSP0/CPU0/0</pre>	

IPv6 Stateless Address Auto Configuration on Management Interface

The IPv6 Stateless Address Auto Configuration (SLAAC) is used to automatically assign IPv6 addresses to the host interfaces. This functionality can be used when the exact addresses used by the host need not be specific, as long as they are unique and properly routable. SLAAC helps in automating provisioning of the router.

IPv6 auto configuration is disabled by default. Perform this task to enable IPv6 SLAAC. The SLAAC functionality has to be enabled from the Management interface.

SUMMARY STEPS

- 1. configure
- 2. interface MgmtEth interface-path-id
- 3. ipv6 address autoconfig
- 4. show ipv6 interfaces interface-path-id

DETAILED STEPS

Procedure

Step 1 configure

Example:

RP/0/RSP0/CPU0:router# configure

Enters global configuration mode.

Step 2 interface MgmtEth interface-path-id

Example:

RP/0/RSP0/CPU0:router(config) # interface MgmtEth 0/RP0/CPU0/0

Enters interface configuration mode on the specified Management Ethernet interface. Use the notation *rack/slot/module/port*. This example indicates port 0 on the RP card that is installed in slot 0.

Step 3ipv6 address autoconfig

Example:

RP/0/RSP0/CPU0:router(config-if)# ipv6 address autoconfig

Enable IPv6 stateless address auto configuration on the management port.

Step 4 show ipv6 interfaces *interface-path-id*

Example:

RP/0/RSP0/CPU0:router# show ipv6 interfaces gigabitEthernet 0/2/0/0

Displays statistics for interfaces on the router. You can see in the output shown below that IPv6 is enabled and the IPv6 addresses has been auto configured.

Example

This example displays how to enable IPv6 SLAAC auto configuration on management interface:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface MgmtEth 0/RP0/CPU0/1
RP/0/RSP0/CPU0:router(config) # ipv6 address autoconfig
RP/0/RSP0/CPU0:router# show ipv6 interfaces MgmtEth 0/RP0/CPU0/1
MgmtEth0/RSP0/CPU0/1 is Up, ipv6 protocol is Up, Vrfid is default (0x60000000)
  IPv6 is enabled, link-local address is fe80::cecc:ccff:fecc:cccc
  Global unicast address(es):
    25::cecc:ccff:fecc:cccc, subnet is 25::/64
    2020:abcd:1234:1001:cecc:ccff:fecc:cccc, subnet is 2020:abcd:1234:1001::/64
    20::cecc:ccff:fecc:cccc, subnet is 20::/64
    447::cecc:ccff:fecc:cccc, subnet is 447::/64
    448::cecc:ccff:fecc:cccc, subnet is 448::/64
    13::cecc:ccff:fecc:cccc, subnet is 13::/64
    3457::cecc:ccff:fecc:cccc, subnet is 3457::/64
    19::cecc:ccff:fecc:cccc, subnet is 19::/64
  Joined group address(es): ff02::1:ffcc:cccc ff02::2 ff02::1
  MTU is 1514 (1500 is available to IPv6)
  ICMP redirects are disabled
  ICMP unreachables are enabled
  ND DAD is enabled, number of DAD attempts 1
```

```
ND reachable time is 0 milliseconds

ND cache entry limit is 100000000

ND advertised retransmit interval is 0 milliseconds

Hosts use stateless autoconfig for addresses.

Outgoing access list is not set

Inbound common access list is not set, access list is not set

Table Id is 0xe0800000

Complete protocol adjacency: 2

Complete glean adjacency: 0

Incomplete glean adjacency: 0

Dropped protocol request: 0

Dropped glean request: 0
```

Configuring the Duplex Mode for a Management Ethernet Interface

Perform this task to configure the duplex mode of the Management Ethernet interfaces for the RPs.

SUMMARY STEPS

- 1. configure
- 2. interface MgmtEth interface-path-id
- 3. duplex [full | half]
- 4. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	interface MgmtEth interface-path-id	Enters interface configuration mode and specifies the
	Example:	Management Ethernet interface name and instance.
	<pre>RP/0/RSP0/CPU0:router(config)# interface MgmtEth 0/RSP0/CPU0/0</pre>	
Step 3	duplex [full half]	Configures the interface duplex mode. Valid options are
	Example:	full or half.
	RP/0/RSP0/CPU0:router(config-if)# duplex full	 Note To return the system to autonegotiated duplex operation, use the no duplex command.
Step 4	end or commit	Saves configuration changes.
	Example:	

Command or Action	Purpose
 RP/0/RSP0/CPU0:router(config-if)# end	• When you issue the end command, the system prompts you to commit changes:
or	
RP/0/RSP0/CPU0:router(config-if)# commit	Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:
	• Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
	• Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
	• Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
	• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring the Speed for a Management Ethernet Interface

Perform this task to configure the speed of the Management Ethernet interfaces for the RPs.

SUMMARY STEPS

- 1. configure
- 2. interface MgmtEth interface-path-id
- **3.** speed {10 | 100 | 1000}
- 4. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	interface MgmtEth interface-path-id	Enters interface configuration mode and specifies the
	Example:	Management Ethernet interface name and instance.

	Command or Action	Purpose
	<pre>RP/0/RSP0/CPU0:router(config)# interface MgmtEth 0/RSP0/CPU0/0</pre>	
Step 3	speed {10 100 1000}	Configures the interface speed parameter.
	Example:	On a Cisco ASR 9000 Series Router, valid options are 10 or 100 Mbps.
	RP/0/RSP0/CPU0:router(config-if)# speed 100	Note • The default Management Ethernet interface speed is autonegotiated.
		• To return the system to the default autonegotiated speed, use the no speed command.
Step 4	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	<pre>RP/0/RSP0/CPU0:router(config-if) # end</pre>	
	or	Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:
	RP/0/RSP0/CPU0:router(config-if)# commit	• 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.

Modifying the MAC Address for a Management Ethernet Interface

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

SUMMARY STEPS

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

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	interface MgmtEth interface-path-id	Enters interface configuration mode and specifies the
	Example:	Management Ethernet interface name and instance.
	<pre>RP/0/RSP0/CPU0:router(config)# interface MgmtEth 0/RSP0/CPU0/0</pre>	
Step 3	mac-address address	Configures the MAC layer address of the Management
	Example:	Ethernet interface.
	RP/0/RSP0/CPU0:router(config-if)# mac-address 0001.2468.ABCD	• To return the device to its default MAC address, use the no mac-address address command.
Step 4	end or commit	Saves configuration changes.
	Example:	 When you issue the end command, the system promp you to commit changes:
	<pre>RP/0/RSP0/CPU0:router(config-if)# end</pre>	
	or	Uncommitted changes found, commit them bef exiting(yes/no/cancel)? [cancel]:
	RP/0/RSP0/CPU0:router(config-if)# commit	• Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Verifying Management Ethernet Interface Configuration

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

SUMMARY STEPS

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

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	show interfaces MgmtEth interface-path-id	Displays the Management Ethernet interface configuration.
	Example:	
	RP/0/RSP0/CPU0:router# show interfaces MgmtEth 0/RSP0/CPU0/0	
Step 2	show running-config interface MgmtEth interface-path-id	Displays the running configuration.
	Example:	
	RP/0/RSP0/CPU0:router# show running-config interface MgmtEth 0/RSP0/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/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface MgmtEth 0/RSP0RP0
RP/0/RSP0/CPU0:router(config)# ipv4 address 172.29.52.70 255.255.255.0
RP/0/RSP0/CPU0:router(config-if)# speed 100
RP/0/RSP0/CPU0:router(config-if)# duplex full
RP/0/RSP0/CPU0:router(config-if)# no shutdown
RP/0/RSP0/CPU0:router(config-if)# commit
RP/0/RSP0/CPU0:Mar 26 01:09:28.685 :ifmgr[190]:%LINK-3-UPDOWN :Interface MgmtEth0/RSP0/CPU0/0,
changed state to Up
RP/0/RSP0/CPU0:router(config-if)# end
```

```
RP/0/RSP0/CPU0:router# show interfaces MgmtEth 0/RSP0RP0
```

L

MMgmtEth0/RSP0/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 57 input errors, 40 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/RSP0/CPU0:router# show running-config interface MgmtEth 0/RSP0RP0

interface MgmtEth0/RSP0/CPU0/0
description Connected to Lab LAN
ipv4 address 172.29.52.70 255.255.0
!



Configuring Ethernet Interfaces

This module describes the configuration of Ethernet interfaces.

The distributed Gigabit Ethernet and 10-Gigabit, 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 and Layer 2 and Layer 3 switches.



Tip You can programmatically configure and manage the Ethernet interfaces using <code>openconfig-ethernet-if.yang</code> and <code>openconfig-interfaces.yang</code> OpenConfig data models. To get started with using data models, see the *Programmability Configuration Guide for Cisco ASR 9000 Series Routers*.

Release	Modification
Release 3.7.2	Support was added on the Cisco ASR 9000 Series Router for the following line cards:
	• 40-Port Gigabit Ethernet Medium Queue and High Queue Line Cards (A9K-40GE-B and A9K-40GE-E)
	• 4-Port 10-Gigabit Ethernet Medium Queue and High Queue Line Cards (A9K-4T-B and A9K-4T-E)
	• 8-Port 10-Gigabit Ethernet Medium Queue and High Queue DX Line Cards (A9K-8T/4-B and A9K-8T/4-E) (2:1 oversubscribed)

Feature History for Configuring Ethernet Interfaces

Release 3.9.0	Support was added on the Cisco ASR 9000 Series Router for the following line cards:	
	• 40-Port Gigabit Ethernet Low Queue Line Card (A9K-40GE-L)	
	• 4-Port 10-Gigabit Ethernet Low Queue Line Card (A9K-4T-L)	
	• 8-Port 10-Gigabit Ethernet Low Queue DX Line Card (A9K-8T/4-L) (2:1 oversubscribed)	
	• 8-Port 10-Gigabit Ethernet Low and High Queue Line Card (A9K-8T-L and A9K-8T-E)	
	• 2-Port 10-Gigabit Ethernet, 20-Port Gigabit Ethernet Medium Queue and High Queue Combination Line Cards (A9K-2T20GE-B and A9K-2T20GE-L)	
	Support for these features was added:	
	Frequency Synchronization	
	• SyncE	
Release 3.9.1	Support was added on the Cisco ASR 9000 Series Router for the following line cards:	
	• 8-Port 10-Gigabit Ethernet Medium Queue Line Card (A9K-8T-B)	
	• 16-Port 10-Gigabit Ethernet SFP+ Line Card (A9K-16T/8-B and A9K-16T/8-B+AIP)	
Release 4.0.1	Support for Layer 2 statistics collection for performance monitoring on Layer 2 subinterfaces (EFPs) is added.	
Release 4.1.0	Support for Link Layer Discovery Protocol (LLDP) was added.	
	Note LLDP is not supported under management interface for this platform.	
Release 4.1.1	Support was added for MAC address accounting feature.	
Release 4.2.2	Support for Unidirectional Link Routing (UDLR) was introduced.	

Release 4.3.1	Support was added on the Cisco ASR 9000 Series Router for these line cards:
	• A9K-MOD80-SE
	• A9K-MOD80-TR
	• A9K-MOD160-SE
	• A9K-MOD160-TR
	Support was added on the Cisco ASR 9000 Series Router for these Modular Port Adaptors (MPAs):
	• A9K-MPA-20X1GE
	• A9K-MPA-4X10GE
	• A9K-MPA-2X10GE
	• A9K-MPA-8X10GE (supported only with MOD160 Line cards)
	• A9K-MPA-2X40GE (supported only with MOD160 Line cards)
	• A9K-MPA-1X40GE
Release 5.3.1	Support for IRB with Provider Backbone Bridge (PBB).
Release 6.2.1	Link Degrade Alarm in Case of Link Loss Changing by 2 dB from the Preset Value

- Configuring Ethernet Interfaces, on page 29
- Prerequisites for Configuring Ethernet Interfaces, on page 31
- Information About Configuring Ethernet, on page 32
- How to Configure Ethernet, on page 51
- Configuration Examples for Ethernet, on page 74
- How to Configure Interfaces in Breakout Mode, on page 77

Configuring Ethernet Interfaces

This module describes the configuration of Ethernet interfaces.

The distributed Gigabit Ethernet and 10-Gigabit, 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 and Layer 2 and Layer 3 switches.



Tip You can programmatically configure and manage the Ethernet interfaces using <code>openconfig-ethernet-if.yang</code> and <code>openconfig-interfaces.yang</code> OpenConfig data models. To get started with using data models, see the *Programmability Configuration Guide for Cisco ASR 9000 Series Routers*.

Release	Modification
Release 3.7.2	Support was added on the Cisco ASR 9000 Series Router for the following line cards:
	• 40-Port Gigabit Ethernet Medium Queue and High Queue Line Cards (A9K-40GE-B and A9K-40GE-E)
	• 4-Port 10-Gigabit Ethernet Medium Queue and High Queue Line Cards (A9K-4T-B and A9K-4T-E)
	• 8-Port 10-Gigabit Ethernet Medium Queue and High Queue DX Line Cards (A9K-8T/4-B and A9K-8T/4-E) (2:1 oversubscribed)
Release 3.9.0	Support was added on the Cisco ASR 9000 Series Router for the following line cards:
	• 40-Port Gigabit Ethernet Low Queue Line Card (A9K-40GE-L)
	• 4-Port 10-Gigabit Ethernet Low Queue Line Card (A9K-4T-L)
	• 8-Port 10-Gigabit Ethernet Low Queue DX Line Card (A9K-8T/4-L) (2:1 oversubscribed)
	• 8-Port 10-Gigabit Ethernet Low and High Queue Line Card (A9K-8T-L and A9K-8T-E)
	• 2-Port 10-Gigabit Ethernet, 20-Port Gigabit Ethernet Medium Queue and High Queue Combination Line Cards (A9K-2T20GE-B and A9K-2T20GE-L)
	Support for these features was added:
	Frequency Synchronization
	• SyncE
Release 3.9.1	Support was added on the Cisco ASR 9000 Series Router for the following line cards:
	• 8-Port 10-Gigabit Ethernet Medium Queue Line Card (A9K-8T-B)
	• 16-Port 10-Gigabit Ethernet SFP+ Line Card (A9K-16T/8-B and A9K-16T/8-B+AIP)
Release 4.0.1	Support for Layer 2 statistics collection for performance monitoring on Layer 2 subinterfaces (EFPs) is added.
Release 4.1.0	Support for Link Layer Discovery Protocol (LLDP) was added.
	Note LLDP is not supported under management interface for this platform.
Release 4.1.1	Support was added for MAC address accounting feature.

Feature History for Configuring Ethernet Interfaces

L

Release 4.2.2	Support for Unidirectional Link Routing (UDLR) was introduced.	
Release 4.3.1	Support was added on the Cisco ASR 9000 Series Router for these line cards: • A9K-MOD80-SE • A9K-MOD80-TR • A9K-MOD160-SE • A9K-MOD160-TR Support was added on the Cisco ASR 9000 Series Router for these Modular Port Adaptors (MPAs):	
	 A9K-MPA-20X1GE A9K-MPA-4X10GE A9K-MPA-2X10GE A9K-MPA-8X10GE (supported only with MOD160 Line cards) A9K-MPA-2X40GE (supported only with MOD160 Line cards) A9K-MPA-1X40GE 	
Release 5.3.1	Support for IRB with Provider Backbone Bridge (PBB).	
Release 6.2.1	Link Degrade Alarm in Case of Link Loss Changing by 2 dB from the Preset Value	

Prerequisites for Configuring Ethernet 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 Ethernet interfaces, be sure that these tasks and conditions are met:

- Confirm that at least one of these line cards supported on the router is installed:
 - 2-Port 10-Gigabit Ethernet, 20-Port Gigabit Ethernet Combination line card (A9K-2T20GE-B and A9K-2T20GE-L)
 - 4-Port 10-Gigabit Ethernet line card (A9K-4T-L, -B, or -E)
 - 8-Port 10-Gigabit Ethernet DX line card (A9K-8T/4-L, -B, or -E)
 - 8-Port 10-Gigabit Ethernet line card (A9K-8T-L, -B, or -E)
 - 16-Port 10-Gigabit Ethernet SFP+ line card (A9K-16T/8-B and A9K-16T/8-B+AIP)
 - 40-Port Gigabit Ethernet line card (A9K-40GE-L, -B, or -E)
 - 24-Port 10-Gigabit Ethernet Line Card

- 36-Port 10-Gigabit Ethernet Line Card
- 2-Port 100-Gigabit Ethernet Line Card
- ASR 9000 Mod80 Modular Line Card, Service Edge Optimized with modular port adapters
- ASR 9000 Mod80 Modular Line Card, Packet Transport Optimized with modular port adapters
- ASR 9000 Mod160 Modular Line Card, Service Edge Optimized with modular port adapters
- ASR 9000 Mod160 Modular Line Card, Packet Transport Optimized with modular port adapters
- Know the interface IP address.
- You know how to apply the specify the generalized interface name with the generalized notation *rack/slot/module/port*.

Information About Configuring Ethernet

Ethernet is defined by the IEEE 802.3 international standard. It enables the connection of up to 1024 nodes over coaxial, twisted-pair, or fiber-optic cable.

The Cisco ASR 9000 Series Router supports Gigabit Ethernet (1000 Mbps),10-Gigabit Ethernet (10 Gbps), 40-Gigabit Ethernet (40 Gbps), and 100-Gigabit Ethernet (100 Gbps) interfaces.

This section provides the following information sections:

16-Port 10-Gigabit Ethernet SFP+ Line Card

The 16-Port10-Gigabit Ethernet SFP+ line card is a Small Form Factor (SFP transceiver) optical line card introduced in Cisco IOS XR Release 3.9.1 on the Cisco ASR 9000 Series Router. The 16-Port10-Gigabit Ethernet SFP+ line card supports all of the Gigabit Ethernet commands and configurations currently supported on the router.

The 16-Port10-Gigabit Ethernet SFP+ line card is compatible with all existing Cisco ASR 9000 Series Router line cards, route/switch processors (RSPs), and chassis.

Features

The 16-Port10-Gigabit Ethernet SFP+ line card supports these features:

- 16 10-Gigabit Ethernet ports
- 128 10-Gigabit Ethernet ports per system
- 1.28 Tbps per system
- 160 Gbps forwarding
- 120 Gbps bidirectional performance
- SR/LR/ER SFP+ optics
- Feature parity with existing line cards
- Unicast and multicast forwarding at 160 Gbps, with zero packet loss during RSP switchover

Restrictions

The following features are not supported on the 16-Port10-Gigabit Ethernet SFP+ line card:

• DWDM (G.709)

Cisco ASR 9000 Modular Line Cards

The Cisco ASR 9000 Series modular line cards provides a flexible solution to support multiple combinations of Ethernet ports, all in a single slot of the Cisco ASR 9000 Series Aggregation Services Routers. Modular line cards support a wide range of interfaces and densities offering the benefits of network scalability with lower initial costs and ease of upgrades.

The Cisco ASR 9000 Series modular line cards are designed for the Cisco ASR9000 Series Router which accepts pluggable modules. It allows you to cost effectively address lower density Gigabit Ethernet, 10-Gigabit Ethernet, and 40-Gigabit Ethernet traffic. This line card is developed based on the ASR 9000 Enhanced Ethernet Network Processor (NP) and allows you to configure different interface types and also conserve chassis slots.

The Cisco ASR 9000 Series modular line cards accept two Ethernet Plugs (EP). Each Ethernet Plug provides optics, and support circuitry in order to provide GE, 10GE or 40GE ports.

The two versions of Modular Line Cards are:

- Cisco ASR 9000 Mod80 Modular Line Card 2 ASR 9000 Enhanced Ethernet Network Processors (NP) which supports 2 pluggable Ethernet Plugs(EP), and 1 NP for each EP.
- Cisco ASR 9000 Mod160 Modular Line Card 4 ASR 9000 Enhanced Ethernet Network Processors which supports 2 pluggable Ethernet Plugs, and 2 NPs for each EP.



Note A9K-MPA-20X1GE supports a speed of 10Mbps or 100Mbps when using only GLC-TE optics, regardless of MOD models.

Restrictions on Module Port Adaptors

The two MPAs, A9K-MPA-8X10GE and A9K-MPA-2X40GE are supported only in A9K-MOD160-SE and A9K-MOD160-TR Line cards.

These are the specifications of the MPAs:

- A9K-MPA-8X10GE is supported only on the 160 Gigabyte Modular Line Card.
- A9K-MPA-8X10GE is not supported on 80 Gigabyte Modular Line Card.
- A9K-MPA-8X10GE is not supported on the Cisco ASR 9001 Chassis.
- A9K-MPA-8X10GE uses SFP+ Optics. The supported optics are SFP+ LR, SFP+ SR and SFP+ DWDM optics.

All other MPAs are supported in both the flavors of A9K-MOD80-SE/TR and A9K-MOD160-SE/TR Line Cards. For more information on these line cards, see *Cisco ASR 9000 Series Aggregation Services Router Ethernet Line Card Installation Guide* and *Cisco ASR 9000 Series Aggregation Services Router Overview and Reference Guide*.

Default Configuration Values for Gigabit Ethernet and 10-Gigabit Ethernet

This table describes the default interface configuration parameters that are present when an interface is enabled on a Gigabit Ethernet or 10-Gigabit Ethernet modular services card and its associated PLIM.

Note You must use the **shutdown** command to bring an interface administratively down. The interface default is **no shutdown**. When a modular services 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 be entering the **no shutdown** command.

Table 4: Gigabit Ethernet and 10-Gi	nahit Ethernet Modular Services	Card Default Configuration Values
Table 4. Gigable Edicinet and To Gi	gubit Ethernet mouului ocivioco	oura benaun configuration falaes

Parameter	Configuration File Entry	Default Value
MAC accounting	mac-accounting	off
Flow control	flow-control	egress on ingress off
MTU	mtu	 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)

Default Configuration Values for Fast Ethernet

Table 5: Fast Ethernet Default Configuration Values

Parameter	Configuration File Entry	Default Value
MAC accounting	mac-accounting	off
Duplex operation	duplex full duplex half	Auto-negotiates duplex operation
MTU	mtu	1500 bytes
Interface speed	speed	100 Mbps
Auto-negotiation	negotiation auto	disable

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 Cisco ASR 9000 Series 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 Cisco ASR 9000 Series Router, an AC is an interface that is attached to an L2VPN component, such as a bridge domain, pseudowire, or local connect.

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

Cisco IOS XR Software supports a point-to-point end-to-end service, where two Ethernet circuits are connected together. An L2VPN Ethernet port can operate in one of two modes:

- Port Mode—In this mode, all packets reaching the port are sent over the PW (pseudowire), regardless of any VLAN tags that are present on the packets. In VLAN mode, the configuration is performed under the l2transport configuration mode.
- VLAN Mode—Each VLAN on a CE (customer edge) or access network to PE (provider edge) link can be configured as a separate L2VPN connection (using either VC type 4 or VC type 5). In VLAN mode, the configuration is performed under the individual subinterface.



Note The system sets a limit of 24K single vlan tags per NP and a 64K LC limit on the following line cards:

- A9K-MOD400-SE
- A9K-MOD400-CM
- A9K-MOD200-SE/CM
- Cisco ASR 9000 Series 24-port and 48-port dual-rate 10GE/1GE SE/CM
- A9K-8x100 SE/CM
- A99-8x100 SE/CM

Switching can take place in three ways:

- AC-to-PW—Traffic reaching the PE is tunneled over a PW (and conversely, traffic arriving over the PW is sent out over the AC). This is the most common scenario.
- Local switching—Traffic arriving on one AC is immediately sent out of another AC without passing through a pseudowire.
- PW stitching—Traffic arriving on a PW is not sent to an AC, but is sent back into the core over another PW.

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 and PW information.

To configure a point-to-point pseudowire xconnect on an AC, refer to these documents:

- Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Configuration Guide.
- Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Command Reference

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 a variety of speeds over a variety of physical media. The IEEE 802.3 standard covers 10 Mbps Ethernet. Extensions to the IEEE 802.3 standard specify implementations for Gigabit Ethernet, 10-Gigabit Ethernet, and Fast Ethernet.

IEEE 802.3ab 1000BASE-T Gigabit Ethernet

The IEEE 802.3ab protocol standards, or Gigabit Ethernet over copper (also known as 1000BaseT) is an extension of the existing Fast Ethernet standard. It specifies Gigabit Ethernet operation over the Category 5e/6 cabling systems already installed, making it a highly cost-effective solution. As a result, most copper-based environments that run Fast Ethernet can also run Gigabit Ethernet over the existing network infrastructure to dramatically boost network performance for demanding applications.

IEEE 802.3z 1000 Mbps Gigabit Ethernet

Gigabit Ethernet builds on top of the Ethernet protocol, but increases speed tenfold over Fast Ethernet to 1000 Mbps, or 1 Gbps. Gigabit Ethernet allows Ethernet to scale from 10 or 100 Mbps at the desktop to 100 Mbps up to 1000 Mbps in the data center. Gigabit Ethernet conforms to the IEEE 802.3z protocol standard.

By leveraging the current Ethernet standard and the installed base of Ethernet and Fast Ethernet switches and routers, network managers do not need to retrain and relearn a new technology in order to provide support for 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 4.0.1.

MAC Address

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

MAC Accounting

The MAC address accounting feature provides accounting information for IP traffic based on the source and destination MAC addresses on LAN interfaces. This feature calculates the total packet and byte counts for a LAN interface that receives or sends IP packets to or from a unique MAC address. It also records a time stamp for the last packet received or sent.

These statistics are used for traffic monitoring, debugging and billing. For example, with this feature you can determine the volume of traffic that is being sent to and/or received from various peers at NAPS/peering points. This feature is currently supported on Ethernet, FastEthernet, and bundle interfaces and supports Cisco Express Forwarding (CEF), distributed CEF (dCEF), flow, and optimum switching.



Note

A maximum of 512 MAC addresses per trunk interface are supported for MAC address accounting.

Ethernet MTU

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

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

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



Note

IPv6 does not support fragmentation.

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

Jumbo frame support is automatically enabled 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).



ASIC on 9000v considers all the packets greater than 1514 byte as oversized frame.

Flow Control on Ethernet Interfaces

The flow control used 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 used on standard management interfaces. Flow control can be activated or deactivated for ingress traffic only. It is automatically implemented for egress traffic.

802.10 VLAN

A VLAN is a group of devices on one or more LANs that are configured so that they can communicate as if they were attached to the same wire, when in fact they are located on a number of different LAN segments. Because VLANs are based on logical instead of physical connections, it is very 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 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.

VRRP

The Virtual Router Redundancy Protocol (VRRP) eliminates the single point of failure inherent in the static default routed environment. VRRP specifies an election protocol that dynamically assigns responsibility for a virtual router to one of the VPN concentrators on a LAN. The VRRP VPN concentrator controlling the IP addresses associated with a virtual router is termed as the primary concentrator, and forwards packets sent to those IP addresses. When the primary concentrator becomes unavailable, a backup VPN concentrator takes over.

For more information on VRRP, see the *Implementing VRRP* module of *Cisco ASR 9000 Series Router IP* Addresses and Services Configuration Guide.

HSRP

Hot Standby Routing Protocol (HSRP) is a proprietary protocol from Cisco. HSRP is a routing protocol that provides backup to a router in the event of failure. Several routers are connected to the same segment of an Ethernet, FDDI, or token-ring network and work together to present the appearance of a single virtual router

on the LAN. The routers share the same IP and MAC addresses and therefore, in the event of failure of one router, the hosts on the LAN are able to continue forwarding packets to a consistent IP and MAC address. The transfer of routing responsibilities from one device to another is transparent to the user.

HSRP is designed to support non disruptive switchover of IP traffic in certain circumstances and to allow hosts to appear to use a single router and to maintain connectivity even if the actual first hop router they are using fails. In other words, HSRP protects against the failure of the first hop router when the source host cannot learn the IP address of the first hop router dynamically. Multiple routers participate in HSRP and in concert create the illusion of a single virtual router. HSRP ensures that one and only one of the routers is forwarding packets on behalf of the virtual router. End hosts forward their packets to the virtual router.

The router forwarding packets is known as the *active router*. A standby router is selected to replace the active router should it fail. HSRP provides a mechanism for determining active and standby routers, using the IP addresses on the participating routers. If an active router fails a standby router can take over without a major interruption in the host's connectivity.

HSRP runs on top of User Datagram Protocol (UDP), and uses port number 1985. Routers use their actual IP address as the source address for protocol packets, not the virtual IP address, so that the HSRP routers can identify each other.

For more information on HSRP, see the *Implementing HSRP* module of *Cisco ASR 9000 Series Router Cisco IOS XR*

Link Autonegotiation on Ethernet Interfaces

Feature Name	Release Information	Description
Autonegotiation Support on Cisco ASR 9901 Router	Release 7.4.1	This release introduces support for Autonegotiation feature on the 1GE fiber optic plugged in Cisco ASR 9901 Router's dual-rate (1/10 GE) interface.
		This feature enables devices to automatically exchange, over a link, information about port speed and duplex modes.
		To enable the auto negotiation on 1GE interface, use the negotiation auto command.

Table 6: Feature History Table

Link autonegotiation ensures that devices that share a link segment are automatically configured with the highest performance mode of interoperation. Use the **negotiation auto** command in interface configuration mode to enable link autonegotiation on an Ethernet interface. On line card Ethernet interfaces, link autonegotiation is disabled by default.

Note

The **negotiation auto** command is available on Gigabit Ethernet interfaces only.

This table describes the performance of the system for different combinations of the speed modes. The specified command produces the resulting system action, provided that you have configured autonegotiation on the interface.

duplex Command	speed Command	
full-duplex	no speed	Forces full duplex and auto-negotiates for speed.
full-duplex	speed 1000	Forces full duplex and 1000 Mbps.
full-duplex	speed 100	Forces full duplex and 100 Mbps.
full-duplex	speed 10	Forces full duplex and 10 Mbps.
half-duplex	no speed	Forces half duplex and auto-negotiates for speed.
half-duplex	speed 1000	Forces half duplex and 1000 Mbps.
half-duplex	speed 100	Forces half duplex and 100 Mbps.
half-duplex	speed 10	Forces half duplex and 10 Mbps.

Fast Polling for WAN-PHY

Fast Ethernet interfaces must be continuously monitored in order to detect any link that is not working due to BER errors (bit error rate) and to bring down the interface connected to that link. The Fast Polling feature polls interfaces at a fast rate and brings down the interface in case of BER errors, thereby minimizing service impact. The Fast Polling feature for WAN-PHY reduces convergence time to 150 ms on the ASR 9000 Enhanced Ethernet line card and the ASR 9000 High Density 100GE Ethernet line cards with 10G and higher rate ports.

To configure the fast polling for WAN-PHY on ASR 9000 Enhanced Ethernet line card, use the **wanphy poll-timer** *value-in-milliseconds* command in configuration mode.

The fast polling for WAN-PHY on ASR 9000 High Density 100GE Ethernet line card is enabled by default.

Early Indication of Link Loss Change

This feature helps in early detection of a link loss between two devices and prevents any service impact. To enable this feature user must configure a receiving optical power threshold value. Whenever the receiving power crosses the threshold, the power degrade alarm is raised. The alram resets after the power exceeds threshold value by 2db.

For example, consider you have configured receiving optical power threshold value to -10db. When the input power on the interface reduces below -10db due to fiber degradation, then the optical power alarm is raised. After fiber degradation is attended power value will improve. Once the power value crosses above -8db, that is 2db more than threshold configured, then the alarm resets.

To configure the receiving power optical threshold value, use **optical-power alarm rx** < value in db> command in interface configuration mode.

Use this show command to view the alarm status:

show controllers <ethernet-interface> internal

Use this command to view threshold value configured, and the minimum and maximum threshold value:

show controllers <ethernet-interface> control

Subinterfaces on the Cisco ASR 9000 Series Router

In Cisco IOS XR, interfaces are, by default, main interfaces. A main interface is also called a trunk interface, which is not to be confused with the usage of the word trunk in the context of VLAN trunking.

There are three types of trunk interfaces:

- · Physical
- Bundle

On the Cisco ASR 9000 Series Router, physical interfaces are automatically created when the router recognizes a card and its physical interfaces. However, bundle interfaces are not automatically created. They are created when they are configured by the user.

The following configuration samples are examples of trunk interfaces being created:

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

A subinterface is a logical interface that is created under a trunk interface.

To create a subinterface, the user must first identify a trunk interface under which to place it. In the case of bundle interfaces, if one does not already exist, a bundle interface must be created before any subinterfaces can be created under it.

The user then assigns a subinterface number to the subinterface to be created. 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 would be 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, GigabitEthernet 0/5/0/0. A subinterface, GigabitEthernet 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 GigabitEthernet0/5/0/0.0
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 GigabitEthernet0/5/0/0
Mon Sep 21 11:15:42.654 EDT
Building configuration...
interface GigabitEthernet0/5/0/0
shutdown
!
interface GigabitEthernet0/5/0/0.0
encapsulation dotlq 100
!
interface GigabitEthernet0/5/0/1
shutdown
!
```

When a subinterface is first created, the Cisco ASR 9000 Series 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, GigabitEthernet 0/5/0/0, followed by the display output for the subinterface GigabitEthernet 0/5/0/0.0.

```
RP/0/RSP0/CPU0:router# show interface gigabitEthernet 0/5/0/0
Mon Sep 21 11:12:51.068 EDT
GigabitEthernet0/5/0/0 is administratively down, line protocol is administratively down.
  Interface state transitions: 0
  Hardware is GigabitEthernet, 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 802.10 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 gigabitEthernet0/5/0/0.0
Mon Sep 21 11:12:55.657 EDT
GigabitEthernet0/5/0/0.0 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 dot1g 100
```

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

```
RP/0/RSP0/CPU0:router#
RP/0/RSP0/CPU0:router# show run | begin GigabitEthernet0/5/0/0
Mon Sep 21 11:42:27.100 EDT
Building configuration ...
interface GigabitEthernet0/5/0/0
negotiation auto
interface GigabitEthernet0/5/0/0.0
encapsulation dot1q 100
!
interface GigabitEthernet0/5/0/1
shutdown
!
RP/0/RSP0/CPU0:router# conf
Mon Sep 21 11:42:32.374 EDT
RP/0/RSP0/CPU0:router(config) # no interface GigabitEthernet0/5/0/0.0
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 GigabitEthernet0/5/0/0
Mon Sep 21 11:42:57.262 EDT
Building configuration...
interface GigabitEthernet0/5/0/0
negotiation auto
interface GigabitEthernet0/5/0/1
```

shutdown

Layer 2, Layer 3, and EFP's

On the Cisco ASR 9000 Series 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/RSP0/CPU0:router# config
Mon Sep 21 12:05:33.142 EDT
RP/0/RSP0/CPU0:router(config) # interface GigabitEthernet0/5/0/0
RP/0/RSP0/CPU0:router(config-if)# ipv4 address 10.0.0.1/24
RP/0/RSP0/CPU0:router(config-if) # commit
RP/0/RSP0/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/RSP0/CPU0:router(config-if) # end
RP/0/RSP0/CPU0:Sep 21 12:06:01.890 : config[65794]: %MGBL-SYS-5-CONFIG I : Configured from
console by root
RP/0/RSP0/CPU0:router# show run | begin GigabitEthernet0/5/0/0
Mon Sep 21 12:06:19.535 EDT
Building configuration...
interface GigabitEthernet0/5/0/0
 ipv4 address 10.0.0.1 255.255.255.0
negotiation auto
interface GigabitEthernet0/5/0/1
shutdown
T.
RP/0/RSP0/CPU0:router#
RP/0/RSP0/CPU0:router#
RP/0/RSP0/CPU0:router# conf
Mon Sep 21 12:08:07.426 EDT
RP/0/RSP0/CPU0:router(config) # interface GigabitEthernet0/5/0/0 l2transport
RP/0/RSP0/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/RSP0/CPU0:router(config-if-12) # no ipv4 address
```

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

RP/0/RSP0/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/RSP0/CPU0:router(config-if) # end
RP/0/RSP0/CPU0:Sep 21 12:08:38.726 : config[65794]: %MGBL-SYS-5-CONFIG I : Configured from
console by root
RP/0/RSP0/CPU0:router#
RP/0/RSP0/CPU0:router# show run interface GigabitEthernet0/5/0/0
Mon Sep 21 12:09:02.471 EDT
interface GigabitEthernet0/5/0/0
negotiation auto
12transport
 1
1
RP/0/RSP0/CPU0:router#
RP/0/RSP0/CPU0:router# conf
Mon Sep 21 12:09:08.658 EDT
RP/0/RSP0/CPU0:router(config) # interface GigabitEthernet0/5/0/0.0
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet0/5/0/0.0
RP/0/RSP0/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/RSP0/CPU0:router(config-subif)#
RP/0/RSP0/CPU0:router(config-subif)# interface GigabitEthernet0/5/0/0
RP/0/RSP0/CPU0:router(config-if) # no l2transport
RP/0/RSP0/CPU0:router(config-if)# interface GigabitEthernet0/5/0/0.0
RP/0/RSP0/CPU0:router(config-subif) # encapsulation dot1q 99
RP/0/RSP0/CPU0:router(config-subif) # ipv4 address 11.0.0.1/24
RP/0/RSP0/CPU0:router(config-subif) # interface GigabitEthernet0/5/0/0.1 l2transport
RP/0/RSP0/CPU0:router(config-subif) # encapsulation dot1q 700
RP/0/RSP0/CPU0:router(config-subif) # commit
RP/0/RSP0/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/RSP0/CPU0:router(config-subif) # end
RP/0/RSP0/CPU0:Sep 21 12:11:50.133 : config[65794]: %MGBL-SYS-5-CONFIG I : Configured from
console by root
RP/0/RSP0/CPU0:router#
RP/0/RSP0/CPU0:router# show run | b GigabitEthernet0/5/0/0
Mon Sep 21 12:12:00.248 EDT
Building configuration ...
interface GigabitEthernet0/5/0/0
negotiation auto
1
interface GigabitEthernet0/5/0/0.0
ipv4 address 11.0.0.1 255.255.255.0
 encapsulation dot1q 99
1
interface GigabitEthernet0/5/0/0.1 l2transport
 encapsulation dot1q 700
1
interface GigabitEthernet0/5/0/1
shutdown
1
```

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. For more information about EFPs, see the Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Configuration Guide.

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.

Enhanced Performance Monitoring for Layer 2 Subinterfaces (EFPs)

Beginning in Cisco IOS XR Release 4.0.1, the Cisco ASR 9000 Series 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. For information about how to configure Performance Monitoring, see the "Implementing Performance Management" chapter of the Cisco ASR 9000 Series Aggregation Services Router System Monitoring Configuration Guide.

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 4.0.1:

- 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 provides support for SyncE-capable Ethernet on the Cisco ASR 9000 Series Router. Frequency Synchronization provides the ability to distribute precision clock signals around the network. Highly accurate timing signals are initially injected into the Cisco ASR 9000 Series Router in the network from an external timing technology (such as Cesium atomic clocks, or GPS), and used to clock 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/SDH networks, but is now provided over Ethernet for Cisco ASR 9000 Series Aggregation Services Routers with Synchronous Ethernet capability. For more information, see *Cisco ASR 9000 Series Aggregation Services Router System Management Configuration Guide*.

LLDP



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

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 Cisco ASR 9000 Series Router also supports the IEEE 802.1AB LLDP. LLDP is also a neighbor discovery protocol that is used for network devices to advertise information about themselves to other devices on the network. This protocol runs over the Data Link Layer, which allows two systems running different network layer protocols to learn about each other.

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, 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 disable** command.

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)

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 for a certain amount time specified in the TTL TLV, before aging and removing the information.

LLDP supports the following additional operational characteristics:

- LLDP can operate independently in transmit or receive modes.
- LLDP operates as a slow protocol using only untagged frames, with transmission speeds of less than 5 frames per second.

- 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 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 Cisco ASR 9000 Series 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 TLV will be populated 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 Cisco ASR 9000 Series Router:

- Locally configured address
- MgmtEth0/RSP0/CPU0/0
- MgmtEth0/RSP0/CPU0/1
- MgmtEth0/RSP1/CPU0/0
- MgmtEth0/RSP1/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 is can be used as an LLDP management address.
- LLDP is supported for the nearest physically attached, non-tunneled neighbors.

• Port ID TLVs are supported for Ethernet interfaces, subinterfaces, bundle interfaces, and bundle subinterfaces.

Unsupported LLDP Functions

The following LLDP functions are not supported on the Cisco ASR 9000 Series 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.

Enabling LLDP Per Interface

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

- 1. RP/0/RSP0/CPU0:ios(config) # int gigabitEthernet 0/2/0/0
- 2. RP/0/RSP0/CPU0:ios(config-if) # no sh
- RP/0/RSP0/CPU0:ios(config-if)#commit
- 4. RP/0/RSP0/CPU0:ios(config-if)#lldp ?
- 5. RP/0/RSP0/CPU0:ios(config-if)#lldp enable
- 6. 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 GigabitEthernet0/1/0/0
shutdown
I.
interface GigabitEthernet0/1/0/1
shutdown
interface GigabitEthernet0/1/0/2
shutdown
T.
interface GigabitEthernet0/2/0/0
Shutdown
interface GigabitEthernet0/2/0/1
shutdown
interface GigabitEthernet0/2/0/2
```
shutdown ! end

Verification

```
Verifying the config
_____
RP/0/RSP0/CPU0:ios#sh lldp interface <==== LLDP enabled only on GigEth0/2/0/0
Wed Jun 27 12:43:26.252 IST
GigabitEthernet0/2/0/0:
       Tx: enabled
       Rx: enabled
       Tx state: IDLE
       Rx state: WAIT FOR FRAME
RP/0/RSP0/CPU0: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
              Gi0/2/0/0
                                  120
                                                            Gi0/2/0/0
                                                                            <===== T.T.DP
ios
                                           R
enabled only on GigEth0/2/0/0 and neighborship seen for the same.
Total entries displayed: 1
RP/0/RSP0/CPU0:ios#
```

Unidirectional Link Routing

Unidirectional Link Routing(UDLR) feature allows a port to unidirectionally transmit or receive traffic. Therefore, instead of using two strands of fiber for a full-duplex Gigabit Ethernet or 10Gigabit Ethernet port, UDLR uses only one strand of fiber that either transmits or receives the one-way traffic depending on the configuration. This improves the effectiveness and also enables you to double the bandwidth with existing fiber infrastructure.

Cisco IOS XR Software supports Unidirectional Link Routing feature on these line cards:

- A9K- 24T-TR 24-port 10 Gigabit Ethernet line cards
- A9K- 24T-SE 24-port 10 Gigabit Ethernet line cards
- A9K- 36T-TR 36-port 10 Gigabit Ethernet line cards
- A9K- 36T-SE 36-port 10 Gigabit Ethernet line cards

UDLR is used for applications such as video streaming, where most of the traffic is sent as unacknowledged unidirectional video broadcast streams.

How to Configure Ethernet

This section provides the following configuration procedures:

Configuring Ethernet Interfaces

This section provides the following configuration procedures:

Configuring Gigabit Ethernet Interfaces

Use the following procedure to create a basic Gigabit Ethernet or 10-Gigabit Ethernet interface configuration.

SUMMARY STEPS

- 1. show version
- 2. show interfaces [GigabitEthernet | TenGigE
- **3**. configure
- 4. interface [GigabitEthernet | TenGigE
- 5. ipv4 address ip-address mask
- 6. flow-control {bidirectional| egress | ingress}
- 7. mtu bytes
- 8. mac-address value1.value2.value3
- 9. negotiation auto
- 10. no shutdown
- 11. end or commit
- **12**. **show interfaces** [GigabitEthernet | TenGigE] *interface-path-id*

DETAILED STEPS

	Command or Action	Purpose
Step 1	show version Example:	(Optional) Displays the current software version, and can also be used to confirm that the router recognizes the modular services card.
	RP/0/RSP0/CPU0:router# show version	
Step 2	<pre>show interfaces [GigabitEthernet TenGigE Example: RP/0/RSP0/CPU0:router# show interface TenGigE 0/1/0/0</pre>	 (Optional) Displays the configured interface and checks the status of each interface port. Possible interface types for this procedure are: GigabitEthernet TenGigE
Step 3	<pre>configure Example: RP/0/RSP0/CPU0:router# configure terminal</pre>	Enters global configuration mode.

I

	Command or Action	Purpose
Step 4	<pre>interface [GigabitEthernet TenGigE Example: RP/0/RSP0/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>. Possible interface types for this procedure are: GigabitEthernet TenGigE Note The example indicates an 8-port 10-Gigabit
Step 5	<pre>ipv4 address ip-address mask Example: RP/0/RSP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.224</pre>	 Ethernet interface in modular services card slot 1. Assigns an IP address and subnet mask to the interface. Replace <i>ip-address</i> with the primary IPv4 address for the interface. Replace <i>mask</i> 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	<pre>flow-control {bidirectional egress ingress} Example: RP/0/RSP0/CPU0:router(config-if)# flow control ingress</pre>	 (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	<pre>mtu bytes Example: RP/0/RSP0/CPU0:router(config-if)# mtu 1448</pre>	 (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 Gigabit Ethernet and 10-Gigabit Ethernet mtu values is 64 bytes to 65535 bytes.

tress of the Management alle, and low 2 bytes, lress in hexadecimal.
lle, and low 2 bytes, lress in hexadecimal.
le is 0 to ffff.
n on a Gigabit Ethernet licitly enabled on both eed and duplex settings on both ends of the any speed or duplex nually take precedence. ommand is available on ces only.
ion, which forces an
mand, the system ges:
, commit them before
tion changes to the its the configuration r to EXEC mode.
ation session and returns shout committing the uter in the current t exiting or committing save the configuration guration file and remain ion.

	Command or Action	Purpose
Step 12	<pre>show interfaces [GigabitEthernet TenGigE] interface-path-id</pre>	(Optional) Displays statistics for interfaces on the router.
	Example:	
	RP/0/RSP0/CPU0:router# show interfaces TenGigE 0/3/0/0	

What to do next

To configure MAC Accounting on the Ethernet interface, see the "Configuring MAC Accounting on an Ethernet Interface" section later in this module.

To configure an AC on the Ethernet port for Layer 2 VPN implementation, see the "Configuring a L2VPN Ethernet Port" section later in this module.

To attach Layer 3 service policies, such as Multiprotocol Label Switching (MPLS) or Quality of Service (QoS), to the Ethernet interface, refer to the appropriate Cisco IOS XR software configuration guide.

Configuring a Fast Ethernet Interface

What to do next

- To configure an AC on the Fast Ethernet port for Layer 2 VPN implementation, see the "Configuring a L2VPN Ethernet Port" section later in this module.
- To attach Layer 3 service policies, such as Multiprotocol Label Switching (MPLS) or Quality of Service (QoS), to the Fast Ethernet interface, refer to the appropriate Cisco ASR 9000 Series Router or Cisco IOS XR software configuration guide.

Configuring MAC Accounting on an Ethernet Interface

This task explains how to configure MAC accounting on an Ethernet interface. MAC accounting has special show commands, which are illustrated in this procedure. Otherwise, the configuration is the same as configuring a basic Ethernet interface, and the steps can be combined in one configuration session. See "Configuring Gigabit Ethernet Interfaces" in this module for information about configuring the other common parameters for Ethernet interfaces.

SUMMARY STEPS

- 1. configure
- 2. interface [GigabitEthernet | TenGigE | fastethernet] interface-path-id
- 3. ipv4 address ip-address mask
- 4. mac-accounting {egress | ingress}
- 5. end or commit
- 6. show mac-accounting type location instance

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RP0/CPU0:router# configure	
Step 2	interface [GigabitEthernet TenGigE fastethernet] interface-path-id	Physical interface or virtual interface.
	Example:	• Use the show interfaces command to see a list of all interfaces currently configured on the router.
	<pre>RP/0/RP0/CPU0:router(config) # interface TenGigE 0/1/0/0</pre>	For more information about the syntax for the router, use the question mark (?) online help function.
Step 3	ipv4 address ip-address mask	Assigns an IP address and subnet mask to the interface.
	Example:	• Replace <i>ip-address</i> with the primary IPv4 address for the interface.
	RP/0/RP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224	• Replace <i>mask</i> 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 4	mac-accounting {egress ingress}	Generates accounting information for IP traffic based on
-	Example:	the source and destination MAC addresses on LAN interfaces.
	<pre>RP/0/RP0/CPU0:router(config-if)# mac-accounting egress</pre>	• To disable MAC accounting, use the no form of this command.
Step 5	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	RP/0/RP0/CPU0:router(config-if)# end	
	or	Uncommitted changes found, commit them before

	Command or Action	Purpose
	RP/0/RP0/CPU0:router(config-if)# commit	<pre>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.
Step 6	show mac-accounting type location instance	Displays MAC accounting statistics for an interface.
	Example:	
	RP/0/RP0/CPU0:router# show mac-accounting TenGigE location 0/2/0/4	

Configuring a L2VPN Ethernet Port

Use the following procedure to configure an L2VPN Ethernet port.

Note

The steps in this procedure configure the L2VPN Ethernet port to operate in port mode.

To configure a point-to-point pseudowire xconnect on an AC, see the Implementing MPLS Layer 2 VPNs module of the Cisco IOS XR L2VPN and Ethernet Services Configuration Guide for the Cisco CRS Router.

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

SUMMARY STEPS

- 1. configure
- 2. interface [GigabitEthernet | TenGigE] interface-path-id
- 3. l2transport
- 4. l2protocol cpsv {tunnel | reverse-tunnel}
- 5. end or commit
- 6. show interfaces [GigabitEthernet | TenGigE] interface-path-id

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	interface [GigabitEthernet TenGigE] interface-path-id Example:	Enters interface configuration mode and specifies the Ethernet interface name and notation <i>rack/slot/module/port</i> . Possible interface types for this procedure are:
	RP/0/RSP0/CPU0:router(config)# interface	• GigabitEthernet
		• TenGigE
Step 3	l2transport	Enables Layer 2 transport mode on a port and enter Layer
	Example:	2 transport configuration mode.
	RP/0/RSP0/CPU0:router(config-if)# 12transport	
Step 4	l2protocol cpsv {tunnel reverse-tunnel}	Configures Layer 2 protocol tunneling and protocol data
	Example:	unit (PDU) filtering on an Ethernet interface.
	RP/0/RSP0/CPU0:router(config-if-l2)# l2protocol cpsv tunnel	• tunnel —Specifies L2P1 encapsulation on frames as they enter the interface, and de-encapsulation on frames as they exit they interface.
		• reverse-tunnel —Specifies L2PT encapsulation on frames as they exit the interface, and de-encapsulation on frames as they enter the interface.
Step 5	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	RP/0/RSP0/CPU0:router(config-if-l2)# end	The committee of the second committee before
	or	exiting(yes/no/cancel)? [cancel]:
	RP/0/RSP0/CPU0:router(config-if-l2)# commit	 Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.

	Command or Action	Purpose
		• 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 6	<pre>show interfaces [GigabitEthernet TenGigE] interface-path-id</pre>	(Optional) Displays statistics for interfaces on the router.
	Example:	
	RP/0/RSP0/CPU0:router# show interfaces TenGigE 0/3/0/0	

To configure a point-to-point pseudowire xconnect on an AC, refer to these documents:

- Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Configuration Guide
- Cisco ASR 9000 Series Aggregation Services Router VPN and Ethernet Services Command Reference

Configuring LLDP

N

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 Cisco ASR 9000 Series 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 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 on page 57.

To enable LLDP globally, complete the following steps:

SUMMARY STEPS

- 1. configure
- 2. lldp
- 3. end or commit

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	lldp	Enables LLDP globally for both transmit and receive
	Example:	operation on the system.
	RP/0/RSP0/CPU0:router(config)#11dp	
Step 3	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	RP/0/RSP0/CPU0:router(config)# end	
	or	Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:
	RP/0/RSP0/CPU0:router(config)# commit	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
	• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring Global LLDP Operational Characteristics

The "LLDP Default Configuration" section on page 53 describes the default operational characteristics for LLDP. 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:

SUMMARY STEPS

- **1**. configure
- 2. lldp holdtime seconds
- 3. Ildp reinit seconds
- 4. lldp timer seconds
- 5. end or commit

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	lldp holdtime seconds	(Optional) Specifies the length of time that information
	Example:	from an LLDP packet should be held by the receiving dev before aging and removing it.
	<pre>RP/0/RSP0/CPU0:router(config)#lldp holdtime 60</pre>	
Step 3	lldp reinit seconds	(Optional) Specifies the length of time to delay initialization of LLDP on an interface.
	Example:	
	<pre>RP/0/RSP0/CPU0:router(config)# lldp reinit 4</pre>	
Step 4	lldp timer seconds	(Optional) Specifies the LLDP packet rate.
	Example:	
	<pre>RP/0/RSP0/CPU0:router(config)#lldp reinit 60</pre>	

	Command or Action	Purpose		
Step 5	end or commit	Saves configuration changes.		
	Example:	• When you issue the end command, the system prompts you to commit changes:		
	RP/0/RSP0/CPU0:router(config)# end			
	or	<pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre>		
	RP/0/RSP0/CPU0:router(config)# commit	• 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:

SUMMARY STEPS

- 1. configure
- 2. lldp tlv-select *tlv-name* disable
- 3. end or commit

DETAILED STEPS

	Command or Action	Purpose	
Step 1 configure Er		Enters global configuration mode.	
	Example:		
	RP/0/RSP0/CPU0:router# configure		

	Command or Action	Purpose
Step 2	lldp tlv-select <i>tlv-name</i> disable Example:	(Optional) Specifies that transmission of the selected TLV in LLDP packets is disabled. The <i>tlv-name</i> can be one of the following LLDP TLV types:
	<pre>RP/0/RSP0/CPU0:router(config)# lldp tlv-select system-capabilities disable</pre>	 management-address port-description
		 system-capabilities system-description system-name
Step 3	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	RP/0/RSP0/CPU0:router(config)# end	
	or	Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:
	RP/0/RSP0/CPU0:router(config)# commit	• 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:

SUMMARY STEPS

- 1. configure
- 2. interface GigabitEthernet 0/2/0/0
- 3. lldp
- 4. receive disable

- 5. transmit disable
- 6. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	<pre>interface GigabitEthernet 0/2/0/0 Example: RP/0/RSP0/CPU0:router(config)#interface GigabitEthernet 0/2/0/0</pre>	Enters interface configuration mode and specifies the Ethernet interface name and notation <i>rack/slot/module/port</i> . Possible interface types for this procedure are: • GigabitEthernet • TenGigE
Step 3	Ildp Example: RP/0/RSP0/CPU0:router(config-if)#11dp	(Optional) Enters LLDP configuration mode for the specified interface.
Step 4	<pre>receive disable Example: RP/0/RSP0/CPU0:router(config-lldp)#receive disable</pre>	(Optional) Disables LLDP receive operations on the interface.
Step 5	<pre>transmit disable Example: RP/0/RSP0/CPU0:router(config-lldp)#transmit disable</pre>	(Optional) Disables LLDP transmit operations on the interface.
Step 6	end or commit	Saves configuration changes.
	Example: RP/0/RSP0/CPU0:router(config)# end or RP/0/RSP0/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.

 Command or Action	Purpose
	• 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 GigabitEthernet 0/1/0/7
Wed Apr 13 13:22:30.501 DST
GigabitEthernet0/1/0/7:
    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:

	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.

Configuring UDLR

Use the following procedure to configure UDLR:

SUMMARY STEPS

- 1. configure
- 2. interface TenGigE interface-path-id
- **3.** transport-mode $\{ rx-only | tx-only \}$
- 4. end or commit

DETAILED STEPS

	Command or Action	Purpose		
Step 1	configure	Enters global configuration mode.		
	Example:			
	RP/0/RSP0/CPU0:router# configure			
Step 2	interface TenGigE interface-path-id	Enters interface configuration mode and specifies the		
	Example:	Ethernet interface name and notation <i>rack/slot/module/port</i>		
	<pre>RP/0/RSP0/CPU0:router(config)# interface TenGigE 0/2/0/0</pre>			
Step 3	transport-mode { rx-only tx-only }	Configures the 10GE UDLR mode as receive-only or		
	Example:	transmit-only.		
	<pre>RP/0/RSP0/CPU0:router(config-if)# transport-mode tx-only</pre>			
Step 4	end or commit	Saves configuration changes.		
	Example:	• When you issue the end command, the system prompts you to commit changes:		

Command or Action	Purpose
RP/0/RSP0/CPU0:router(config-if)# end O r	Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:
RP/0/RSP0/CPU0:router(config-if)# commit	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.
E	Command or Action RP/0/RSP0/CPU0:router(config-if)# end or RP/0/RSP0/CPU0:router(config-if)# commit

Configuring the Dual-Rate Line Cards



• speed – Valid values are 1 (for GE) or 10 (for 10GE)



Note Observe the following restrictions:

- The total for *run-length* must equal the total number of ports (either 24 or 48).
- If you configure the speed of the first port in a set of 12 ports to 1 (GE), then all 12 ports in that set must be 1G (for example: 12x1). If you configure the speed of the first port in a set of 12 ports to 10 (10G), then ports can be mixed in groups of 4 (for example: 4x10,4x1,4x10; or 8x10,4x1; or 12x10).
 - The following example is a valid port-mode configuration on the 48-port line card:

port-mode 4x10,8x1,12x10,12x1,12x10

• The following example is not a valid port-mode configuration on the 48-port line card:

```
port-mode 4x1,8x10,12x10,12x1,12x10
```

The following procedure is for configuring the port speed on the 48-port 10-Gigabit Ethernet/Gigabit Ethernet Line Cards:

1. Enter global configuration mode and specify that the console terminal will be the source of the configuration commands:

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

2. Specify the port mode:

```
RP/0/RSP0/CPU0:router(config)# hw-module location 0/5/CPU0 port-mode
4x10,8x1,12x10,12x1,12x10
```

3. Enter the **commit** command to commit all changes you made to the running configuration:

RP/0/RSP0/CPU0:router(config) # commit

Convert Speed of a Dual-mode Optic

A dual-mode optic operates in two modes. You can convert the speed of a dual-mode optic from one to another using the hw-module location node-id port port number breakout interface command.

Cisco 100G and 40G SR-BiDi QSFP Transceiver optic (QSFP-40/100G-SRBD) is a dual-mode optic. By default, the optic works in 100G mode.

In 40-Gbps mode, the Cisco QSFP 40/100-Gbps BiDi transceiver supports link lengths of 100 and 150 meters on laser-optimized OM3 and OM4 multimode fibers, respectively. In 100-Gbps mode, it supports 70 and 100 meters on OM3 and OM4, respectively.

FEC is not supported on QSFP-40/100G-SRBD optic.

Support for QSFP-40/100G-SRBD optic is available on following hardware:

Cisco ASR 9901 Router



Note Configuring breakout on port 20 will shut down ports 10 and 11. Configuring breakout on port 21 will shut down ports 24 and 25. Use the **no shutdown** command to reenable these ports.

- Cisco ASR 9000 12-Port 100GE line card (A99-12x100GE)
- Cisco ASR 9000 4-Port 100GE line card (A9K-4x100GE)

Configuration

This task enables the user to change the dual-mode optic speed option:

RP/0/RSP0/CPU0:router (config) # hw-module location 0/0/CPU0 port 21 breakout 1xFortyGigE RP/0/RSP0/CPU0:router (config) # commit

Verification

Use the show controller command to verify the configuration:

RP/0/RSP0/CPU0:router# show controllers FortyGigE 0/0/0/21/0 internal Wed Nov 11 06:34:26.861 UTC

Internal data for interface: FortyGigE0/0/0/21/0 Subport Number : 0 Port Number : 21 Bay Number : 0 Ifinst : 6 Ifinst Subport : 21 Board Type : 0x003d1013 Bandwidth(Kbps) : 4000000 Transport mode : LAN BIA MAC addr : badb.ad03.a84d Oper. MAC addr : badb.ad03.a84d Egress MAC addr : badb.ad03.a84d Port Available : true Status polling is : enabled Status events are : enabled I/F Handle : 0x04001300 Cfg Link Enabled : tx/rx enabled H/W Tx Enable : yes MTU : 1514 H/W Speed : 40 Gbps H/W Loopback Type : None FEC : Disable H/W FlowCtrl Type : None H/W AutoNeg Enable : Off Rx OPD : Not Supported H/W Link Defects : (0x00000000000000) none H/W Raw Link Defects : (0x00000000000000) none Link Up : yes Link Led Status : Link up -- Green/Amber Serdes fw version : 100.0 Pluggable Present : yes Pluggable Type : 100/40G SRBD Pluggable PID : QSFP-40/100-SRBD Pluggable Compl. : Compliant Pluggable Type Supp.: Supported

Pluggable PID Supp. : Supported

Creating Slices on a Router

This section describes the procedures to configure slices and port-groups on the Cisco ASR 9000 Series 5th Generation Line Cards, and includes the following topics:

Overview

Slicing is a way to transform a shared network into a set of logical networks. A network slice is a logical group of components, such as Optics (QSFP/SFP), PHY, Network Processor, and Fabric Cards. The system already has the default slicing configured when you enable it. However, you can configure the slices as per your network or service requirement.

You can enable slice configuration on the following:

Routers:

- ASR 9902
- ASR 9903
 - A9903-8HG-PEC

Line cards:

- A9K-4HG-FLEX-SE/-TR
- A99-4HG-FLEX-SE/-TR

Advantages and Benefits

- Each slice enables you to deliver traffic of up to 400 Gbps, which is the maximum capacity that an NPU can handle.
- Provides the capability to the users to choose a speed from 400GE to 10GE.

Restrictions on Slice Configurations

- You must configure port speed on the slices in the same or decreasing order. Beginning from the highest port speed to the lowest port speed.
- For 10x1GE or 5x1GE-5x10GE breakout configuration modes with SFP-1G-SX, SFP-1G-LH, SFP-1G-T-X optics, the flow-control configuration must be added under the GigabitEthernet Interface settings to prevent packet drops.

Slice and Port Numbering

The following tables lists the possible group combinations on the line cards and routers:

Router/Line card	Release	Supported Configurations
ASR 9902	Release 7.4.1	Default: 1x100GE,1x100GE,10x10GE,10x10GE
		1x100GE,1x100GE,4x25GE,10x10GE
		1x100GE,4x25GE,4x25GE,1x100GE
		1x100GE,1x100GE,1x100GE,1x100GE
	Release 7.5.1	Default: 1x100GE,1x100GE,10x10GE,10x10GE
		1x100GE,1x100GE,4x25GE,10x10GE
		1x100GE,4x25GE,4x25GE,1x100GE
		1x100GE,1x100GE,1x100GE,1x100GE
		Note The following configuration modes are not supported:
		• 1x40GE/4x10GE
		• 5x1GE_5x10GE
	Release 7.5.2	Added support for the following configuration mode(s):
		• 5x1GE_5x10GE
ASR 9903	Release 7.4.1	Default: 10x10GE,10x10GE,4x10GE,UNUSED
		4x25GE,4x25GE,4x25GE,4x25GE
	Release 7.5.1	Default: 10x10GE,10x10GE,4x10GE,UNUSED
		4x25GE,4x25GE,4x25GE,4x25GE
	Release 7.9.1	Added support for the following configuration mode(s):
		• 5x1GE_5x10GE

Table 8: Possible Configurations for Cisco ASR 9000 Series Routers and 5th Generation Line Cards

Router/Line card	Release	Supported Configurations	
Cisco ASR 9903 800G Multirate Port Expansion Card (A9903-8HG-PEC)	Release 7.4.1	Default: 4x25GE,4x25GE,4x25GE,4x25GE 1x100GE,1x100GE,4x25GE,10x100	
	Release 7.9.1	Added support for the following configuration mode(s):	
		• 5x1GE_5x10GE	
A9K-4HG-FLEX-SE		Default:	
A9K-4HG-FLEX-TR		IOXIOGE, IOXIOGE, AXIOGE, ONOSED	
A99-4HG-FLEX-SE		4x25GE,4x25GE,4x25GE,4x25GE	
A99-4HG-FLEX-TR	Release 7.9.1	Added support for the following configuration mode(s):	
		• 5x1GE_5x10GE	

Configure Slices

All configurations can be accomplished by using appropriate values for the **hw-module** command.

Note Slice configuration is available on A9903-8HG-PEC. For the other ports on PECs, fixed ports, or slices, use the breakout commands.

To configure the slice, use the following command:

```
configurehw-module location location slice [slice_number ] config-mode
[1x100GE,1x100GE,4x25GE,10x10GE]
```

Consider a scenario, where, you as a user or a customer wants to effectively utilize the bandwidth of the router. In this scenario, we are going to configure a slice on a ASR 9000 Series Router that has a A9903-8HG-PEC line card installed. With A9903-8HG-PEC line card, you can configure slice 4 and 5. The rest of the slices are reserved for other line cards and fixed boards.

To configure the slices on the router or the line card, perform the following steps:

- 1. Identify the router capability, the line cards installed on the router, and the maximum bandwidth supported. In this scenario, the maximum supported bandwidth is 400GE per Network Processor.
- Configure slices on the line card or router by using the configure hw-module location location slice [slice_number] config-mode command.
 - **a.** Specify the slices, such as slice 4.
 - **b.** Specify the speed on the ports, such as 4x25GE,4x25GE,4x25GE,4x25GE or 1x100GE,1x100GE,4x25GE,10x10GE.
 - **c.** Save the configuration.

Now, you have successfully configured the slices on the router, which will allow you to use the network bandwidth as per your requirement, without letting the bandwidth getting wasted, which wasn't possible earlier with the port breakout functionality.

For example, you can use the 4x25GE bandwidth for a lower bandwidth consumption service, such as voice calling. And use the 1x100GE bandwidth for video calling service.

Configuration Steps

```
router# configure terminal
router(config)# hw-module location 0/0/CPU0 slice 4 config-mode 4x25GE,4x25GE,4x25GE,4x25GE
  /* Specify the slice where you configure the groups and specify the port speeds for each
group*/
router(config) # commit
```

Verify Slice Configuration

Use this procedure to verify whether the slice is correctly configured.

Verification on ASR 9903

```
Router: #show controllers np valid-port-groups all location 0/0/CPU0
Tue Dec 21 10:47:37.522 UTC
```

Node: 0/0/CPU0:

NPO	valid-	port-grou	ips is	not	supported

NP1 valid-port-groups is not supported

NP2 valid-port-groups is not supported

NP3 valid-port-groups is not supported

```
NP4: Valid Port Groups
```

SEL	Group 0	Group 1	Group 2	Group 3
DFLT	10x10GE	10x10GE	4x10GE	UNUSED
1	4x25GE	4x2.5GE	4x25GE	4x25GE

Notes:

DFLT: SFP10g port 0-23 (default) DFLT is set by default or by clearing any non-default configuration using "no hw-module loc <> slice config-mode" 1: SFP25g port 0-15

NP5: Valid Port Groups

SEL	Group 0		Group 2	Group 3
DFLT	10x10GE	10x10GE	4x10GE	UNUSED
1	4x25GE	4x25GE	4x25GE	4x25GE

.

Notes:

DFLT: SFP10g port 24-47 (default) DFLT is set by default or by clearing any non-default configuration using "no hw-module loc <> slice config-mode" 1: SFP25g port 32-47

```
NP6 valid-port-groups is not supported
NP7 valid-port-groups is not supported
NP8 valid-port-groups is not supported
Router: #
RP/0/RP0/CPU0:router#sh controllers np ports all location 0/0/CPU0
Tue Jan 11 10:43:35.581 UTC
             Node: 0/0/CPU0:
_____
NP Bridge Fia
                               Ports
       _____
0 -- 0 TenGigE0/0/0/0/0 - TenGigE0/0/0/0/3, TenGigE0/0/0/1/0 - TenGigE0/0/0/1/3,
HundredGigE0/0/0/2, FortyGigE0/0/0/3/0
1 --
        1
           FortyGigE0/0/0/4/0, HundredGigE0/0/0/5, TenGigE0/0/0/6/0 - TenGigE0/0/0/6/3,
FortyGigE0/0/0/7/0
2 --
       2 FortyGigE0/0/0/8/0, FortyGigE0/0/0/9/0, TenGigE0/0/0/26 - TenGigE0/0/0/35,
FortyGigE0/0/0/11/0
      3 FortyGigE0/0/0/12/0, HundredGigE0/0/0/13 - HundredGigE0/0/0/15
.3 --
            TwentyFiveGigE0/0/1/0 - TwentyFiveGigE0/0/1/15
4
  --
        4
5 --
        5
           TenGigE0/0/1/24 - TenGigE0/0/1/47
```

Verification on A9K-4HG-FLEX-SE Line Cards

RP/0/RSP0/CPU0:asr-sat-host1(config)#do show controllers np valid-port-groups all location 0/0/CPU0 Tue Dec 21 05:07:32.819 UTC

```
00.07.02.019 010
```

Node: 0/0/CPU0:

```
NPO: Valid Port Groups
```

SEL	Group 0	Group 1	Group 2	Group 3
DFLT	1x100GE	1x100GE	1x100GE	1x100GE
1	1x100GE/1x40GE/4x25GE	1x100GE/1x40GE/4x25GE	1x100GE/1x40GE/4x25GE	1x100GE/1x40GE/4x25GE
2	1x100GE/1x40GE/4x25GE	1x100GE/1x40GE/4x25GE	1x100GE/1x40GE/4x25G	E 10x10GE
3	1x100GE/1x40GE/4x25GE	1x100GE/1x40GE/4x25GE	2 10x10GE	10x10GE
4	1x100GE/1x40GE/4x25GE	10x10GE	10x10GE	10x10GE
5	10x10GE	10x10GE	10x10GE	10x10GE
Notes: DFLT : QSFP port 40, QSFP port 41, QSFP port 42, QSFP port 43 (default) Group 0: QSFP port 40 or SFP10g port 0-9 or SFP25g port 6-9 Group 1: QSFP port 41 or SFP10g port 10-19 or SFP25g port 16-19 Group 2: QSFP port 42 or SFP10g port 20-29 or SFP25g port 26-29 Group 3: QSFP port 43 or SFP10g port 30-39 or SFP25g port 36-39 PP(0/RSP0/CPU0:asr=sat=host1(config)#				
Gro RP/0,	oup 3: QSFP port 43 or /RSP0/CPU0:asr-sat-hos	SFP10g port 30-39 or t1(config)#	SFP25g port 36-39	

For information on breakout, see the hw-module location command.

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/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface TenGigE 0/0/0/1
RP/0/RSP0/CPU0:router(config-if)# ipv4 address 172.18.189.38 255.255.255.224
RP/0/RSP0/CPU0:router(config-if)# flow-control ingress
RP/0/RSP0/CPU0:router(config-if)# mtu 1448
RP/0/RSP0/CPU0:router(config-if)# mac-address 0001.2468.ABCD
RP/0/RSP0/CPU0:router(config-if) # no shutdown
RP/0/RSP0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes
RP/0/RSP0/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 1000000 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 MAC-Accounting: Example

This example indicates how to configure MAC-accounting 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)# ipv4 address 172.18.189.38 255.255.255.224
RP/0/RSP0/CPU0:router(config-if)# mac-accounting egress
RP/0/RSP0/CPU0:router(config-if)# commit
RP/0/RSP0/CPU0:router(config-if)# exit
RP/0/RSP0/CPU0:router(config)# exit
```

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 LLDP: Examples

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

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# lldp
RP/0/RSP0/CPU0:router(config)# lldp holdtime 60
RP/0/RSP0/CPU0:router(config)# lldp reinit 4
RP/0/RSP0/CPU0:router(config)# lldp timer 60
RP/0/RSP0/CPU0:router(config)# commit
```

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

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/2/0/0
RP/0/RSP0/CPU0:router(config-if)# lldp
RP/0/RSP0/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 on the Cisco ASR 9000 Series RouterAdvanced Configuration and Modification of the Management Ethernet Interface on the Cisco ASR 9000 Series Router module 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.

How to Configure Interfaces in Breakout Mode

Table 9: Feature History Table

Feature name	Release Information	Feature Description
4x25GbE Breakout Ports on Cisco ASR 9903 Router	Release 7.5.2	You can now configure 4x25GbE breakout ports on all QSFP28 ports of the Cisco ASR 9903 fixed port router with QSFP-100G-SR4-S hot-pluggable transceiver module.

Information About Breakout

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

Breakout mode options:

- 1x400GbE
- 1x100GbE
- 2x100GbE
- 4x100GbE
- 2x40GbE
- 1x40GbE
- 4x25GbE
- 4x10GbE



Note

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

Restrictions on breakout mode configuration

These restrictions outline the hardware-level limitations, port dependencies, and configuration requirements that must be considered when configuring breakout modes:

- **Primary port configuration:** When the primary port in the same bay is configured in a restricted breakout mode, for example, 400G or 8x50G, breakout configuration on other ports is not allowed.
- Hardware-Level restriction: The system enforces hardware-level restrictions that prevent configuring breakout modes on other ports within the same logical group.

See the hw-module location command for more information.

Breakout Configuration: Examples

These examples show how to configure breakout in a port.

Configure 4x25GbE Breakout in a Port

This example shows how to configure 4x25GbE breakout in a port:

```
RP/0/RP0/CPU0:Router# configure
RP/0/RP0/CPU0:Router(config)# hw-module location 0/0/CPU0 bay 0 port 11 breakout
4xTwentyFiveGigE
RP/0/RP0/CPU0:Router(config)# commit
RP/0/RP0/CPU0:Router(config)# end
RP/0/RP0/CPU0:Router#
```

Verify a Breakout Configuration

Verifying the 4x25GbE breakout configuration:

RP/0/RP0/CPU0:Router# show ipv4	interfaces brief inc	: 0/11/		
TwentyFiveGigE0/0/0/11/0	198.127.6.1	Up	Up	vpn24
TwentyFiveGigE0/0/0/11/1	198.127.7.1	Up	Up	vpn7
TwentyFiveGigE0/0/0/11/2	198.127.4.1	Up	Up	vpn4
TwentyFiveGigE0/0/0/11/3	198.127.9.1	Up	Up	vpn21

Remove the Breakout Configuration

Removing the 4x25GbE breakout configuration:

```
RP/0/RP0/CPU0:uut# configure
RP/0/RP0/CPU0:uut(config)# no hw-module location 0/0/CPU0 bay 0 port 11 breakout
4xTwentyFiveGigE
RP/0/RP0/CPU0:uut(config)# commit
RP/0/RP0/CPU0:uut(config)# end
```

Configure 4x100GbE Breakout in a Port

This example shows how to configure 4x100GbE breakout in a port:

```
RP/0/RP0/CPU0:Router# configure
RP/0/RP0/CPU0:Router(config)# hw-module location 0/0/CPU0 port 0 breakout 4xHundredGigE
RP/0/RP0/CPU0:Router(config)# commit
RP/0/RP0/CPU0:Router(config)# end
RP/0/RP0/CPU0:Router#
```

Verify a Breakout Configuration

Verifying the 4x100GbE breakout configuration:

RP/0/RP0/CPU0:Router# show ipv4	interfaces brief	inc 0/11/		
HundredGigE0/0/0/11/0	198.127.6.1	Up	Up	vpn24
HundredGigE0/0/0/11/1	198.127.7.1	Up	Up	vpn7
HundredGigE0/0/0/11/2	198.127.4.1	Up	Up	vpn4
HundredGigE0/0/0/11/3	198.127.9.1	Up	Up	vpn21

Remove the Breakout Configuration

Removing the 4x100GbE breakout configuration:

```
RP/0/RP0/CPU0:uut# configure
RP/0/RP0/CPU0:uut(config)# no hw-module location 0/0/CPU0 port 0 breakout 4xHundredGigE
```

RP/0/RP0/CPU0:uut(config) # commit RP/0/RP0/CPU0:uut(config) # end



Configuring Ethernet OAM

This module describes the configuration of Ethernet Operations, Administration, and Maintenance (OAM) on the Cisco ASR 9000 Series Aggregation Services Routers.

Release	Modification	
Release 3.7.2	Support for the following features was introduced: • Ethernet Link OAM • Ethernet CFM	
Release 3.7.3	Support for the CFM Exploratory Linktrace feature was introduced.	
Release 3.9.0	Support for the Ethernet SLA feature was introduced.	
Release 3.9.1	 Support for the following features was introduced: Ethernet CFM on Link Aggregation Group (LAG) interfaces (Ethernet bundle interfaces), Ethernet and bundle sub interfaces, and LAG member (bundle member) interfaces. EFD AIS Flexible tagging The ethernet cfm mep domain command is replaced by the ethernet cfm and mep domain commands. 	

Feature History for Configuring Ethernet OAM

I

Release 4.0.0	Support for the following features was introduced:
	• The action link-fault command is replaced by the action uni-directional link fault command.
	• The efd keyword is added to put an interface into the line protocol down state, as an option for the following commands:
	 action capabilities-conflict
	• action discovery-timeout
	• action session-down
	 action uni-directional link-fault
	• Uni-directional link-fault detection to identify local link-faults and send notification to a remote Ethernet OAM peer using the uni-directional link-fault detection command.
	• Support for the following enhancements to Ethernet SLA was added:
	• Support for on-demand Ethernet SLA operations using the ethernet sla on-demand operation commands.
	 One-way delay and jitter measurements using the following new keyword options for the statistics measure command: one-way-delay-ds. one-way-delay-sd. one-way-jitter-ds. one-way-jitter-sd
	• Specification of a test pattern to pad loopback packets when measuring delay.
	• Displaying the time when the minimum (Min) and maximum (Max) values of a statistic occurred in the measurement time period in the show ethernet sla statistics detail command.
Release 4.0.1	Support for Ethernet CFM on Multi-Chassis Link Aggregation Groups (MC-LAG) was added.
Release 4.1.0	Support for the following feature was introduced:
	• E-LMI
	• Timestamps for delay packets were changed from being derived by the sytem time-of-day (NTP) clock to the DTI timing input on the clock-interfaces on the RSP.
	• CFM Y.1731 ITU Carrier Code (ICC)-based MEG ID (MAID) format.
Release 4.2.0	Support for Unidirectional Link Detection Protocol (UDLD) was introduced.
Release 4.3.0	Support for ITU-T Y.1731 Synthetic Loss Measurement was introduced.
Release 4.3.1	Support for ITU-T Y.1731 Loss Measurement was introduced.

Support for Ethernet Data Plane Loopback was introduced. Support for Ethernet CFM down MEP was included. CFM support on the Bundle over Bundle is limited as follows: CFM software offload is not supported on the satellite access bundles (sub)
Support for Ethernet CFM down MEP was included. CFM support on the Bundle over Bundle is limited as follows: CFM software offload is not supported on the satellite access bundles (sub)
CFM support on the Bundle over Bundle is limited as follows: CFM software offload is not supported on the satellite access bundles (sub)
interface over bundle ICL. If CFM is configured on any satellite access bundle interface over bundle ICL, bundle-offload configuration can not be applied. If CFM is configured only on interface in ASR 9000 series other than satellite access bundle interface over bundle ICL, then bundle-offload configuration can be applied.
Support for CFM adaptive bandwidth notifications was introduced.
Support for CFM Hardware Offload on ASR 9000 5th Generation High Density line cards was introduced.

• Prerequisites for Configuring Ethernet OAM, on page 83

- Information About Configuring Ethernet OAM, on page 83
- How to Configure Ethernet OAM, on page 125
- Configuration Examples for Ethernet OAM, on page 191
- CFM Adaptive Bandwidth Notifications, on page 216

Prerequisites for Configuring Ethernet OAM

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 Ethernet OAM, confirm that at least one of the Gigabit Ethernet line cards or Cisco ASR 9000 Enhanced Ethernet line cards are installed on the router.

Information About Configuring Ethernet OAM

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

Ethernet Link OAM

Table 10: Feature Hi	istory Table
----------------------	--------------

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

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 (ELO) features allow you to monitor the quality of the connections on a MAN or a WAN. Service providers can monitor specific events, and if necessary, put specific interfaces into loopback mode for troubleshooting ELO operates on a single physical link, and it can be configured to monitor either side or both sides of that link.

ELO can be configured in the following ways:

- Using an ELO profile: An ELO profile can be configured to set the parameters for multiple interfaces. This simplifies the process of configuring Ethernet Link OAM features on multiple interfaces. An ELO profile and its features can be referenced by other interfaces, allowing them to inherit those features. This is the preferred method of configuring custom ELO settings.
- **Configuring directly on an interface:** Individual ELO features can be configured directly on an interface without being part of a profile. When an interface uses an ELO profile, specific parameters can still be overridden by configuring different values directly on the interface. In such cases, the individually configured features take precedence over the profile settings.

When an ELO packet is received on any one of the Attachment Circuit (AC) interfaces where ELO is not configured, the AC interface multicasts the received packets to other AC interfaces that are part of the Ethernet Virtual Private Network Broadcast Domain (EVPN-BD) to reach the peer. An ELO can be configured on any physical Ethernet interface, including bundle members.

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.

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.

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.

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.

SNMP Traps

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

Unidirectional Link Fault Detection

Unidirectional link fault detection describes an Ethernet link OAM function that runs directly on physical Ethernet interfaces (not VLAN subinterfaces or bundles) that uses a defined link fault message to signal link faults to a remote host. Unidirectional link fault detection offers similar functionality to Gigabit Ethernet and Ten Gigabit Ethernet hardware-level signaling of a link fault, but it is done at a higher protocol layer as part of Ethernet link OAM. The hardware function uses the Remote Fault Indication bit set in a frame that is signaled out-of-band, where unidirectional link fault detection signals the error using an OAMPDU.

Unidirectional link fault detection only applies to a single, physical link. When the remote host receives the link fault message, the interface can be shut down for all higher-layer protocols, and specifically, Layer 2 switching and Layer 3 routing protocols. While the fault is detected, a link fault message is sent periodically to the remote host. Once a fault is no longer detected, the link fault message is no longer sent, and the remote host can bring back the interface.

Unidirectional link fault detection is configured using the **uni-directional link-fault detection** command, and does not affect how the receipt of link-fault messages are handled by the router. Actions to be taken for the receipt of link-fault messages are configured using the **action uni-directional link-fault** command.

Ethernet CFM

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 on the Cisco ASR 9000 Series Router 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.
 - ETH-DM, ETH-SLM—This is supported with the Ethernet SLA feature. For more information about Ethernet SLA, see the "Ethernet SLA" section on page 87.

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

Maintenance Domains

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

Figure 1: 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. For more information about MEPs and MIPs, see the "Maintenance Points" section on page 71.





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 described in the "Maintenance Domains" section on page 69, and ensures that CFM frames for a particular domain cannot leak out beyond the boundary of the domain.

There are two types of MP:

Maintenance End Points (MEPs)—Created at the edge of the domain. Maintenance end points (MEPs) are members of a particular service within a domain and are responsible for sourcing and sinking CFM frames. They periodically transmit continuity check messages and receive similar messages from other

MEPs within their domain. They also transmit traceroute and loopback messages at the request of the administrator. MEPs are responsible for confining CFM messages within the domain.

Maintenance Intermediate Points (MIPs)—Created in the middle of the domain. Unlike MEPS, MIPs do
allow CFM frames at their own level to be forwarded.

MIP Creation

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

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



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

MEP and CFM Processing Overview

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

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



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

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



Figure 3: Monitored Areas for Down and Up MEPs

This figure shows maintenance points at different levels. Because domains are allowed to nest but not intersect (see Figure 3), a MEP at a low level always corresponds with a MEP or MIP at a higher level. In addition, only a single MIP is allowed on any interface—this is generally created in the lowest domain that exists at the interface and that does not have a MEP.



MIPs and Up MEPs can only exist on switched (Layer 2) interfaces, because they send and receive frames from the bridge relay function. Down MEPs can be created on switched (Layer 2) or routed (Layer 3) interfaces.

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

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



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

CFM Protocol Messages

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

This section describes the following CFM messages:

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

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

MIPs also receive CCMs. MIPs use the information to build a MAC learning database that is used when responding to Linktrace. For more information about Linktrace, see the "Linktrace (IEEE 802.1ag and ITU-T Y.1731)" section on page 77.



Figure 4: Continuity Check Message Flow

All the MEPs in a service must transmit CCMs at the same interval. IEEE 802.1ag defines 7 possible intervals that can be used:

- 10ms (applicable on the Cisco ASR 9000 Enhanced Ethernet Line Card)
- 100ms
- 1s
- 10s
- 1 minute
- 10 minutes

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

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.
- A configured numeric identifier for the MEP (the MEP ID). Each MEP in the service must be configured with a different MEP ID.
- A sequence number.
- 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.
- 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 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 an 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. On receiving each LBM, the target maintenance point sends an LBR back to the originating MEP. Loopback indicates whether the destination is reachable or not—it does not allow hop-by-hop discovery of the path. It is similar in concept to an ICMP Echo (ping). Since loopback messages are destined for unicast addresses, they are forwarded like normal data traffic, while observing the maintenance levels. At each device that the loopback reaches, if the outgoing interface is known (in the bridge's forwarding database), then the frame is sent out on that interface. If the outgoing interface is not known, then the message is flooded on all interfaces.

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



Figure 5: 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.

Except for one-way delay and jitter measurements, loopback messages can also be used for Ethernet SLA, if the peer does not support delay measurement.



Note

The Ethernet CFM loopback function should not be confused with the remote loopback functionality in Ethernet Link OAM (see the "Remote Loopback" section on page 68). CFM loopback is used to test connectivity with a remote MP, and only the CFM LBM packets are reflected back, but Ethernet Link OAM remote loopback is used to test a link by taking it out of normal service and putting it into a mode where it reflects back all packets.

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

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

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



Figure 6: 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.

Exploratory Linktrace (Cisco)

Exploratory Linktrace is a Cisco extension to the standard linktrace mechanism described above. It has two primary purposes:

- Provide a mechanism to locate faults in cases where standard linktrace does not work, such as when a MAC address has never been seen previously in the network. For example, if a new MEP has been provisioned but is not working, standard linktrace does not help isolate a problem because no frames will ever have been received from the new MEP. Exploratory Linktrace overcomes this problem.
- Provide a mechanism to map the complete active network topology from a single node. This can only be done currently by examining the topology (for example, the STP blocking state) on each node in the network individually, and manually combining this information to create the overall active topology map. Exploratory linktrace allows this to be done automatically from a single node.

Exploratory Linktrace is implemented using the Vendor Specific Message (VSM) and Vendor Specific Reply (VSR) frames defined in ITU-T Y.1731. These allow vendor-specific extensions to be implemented without degrading interoperability. Exploratory Linktrace can safely be deployed in a network that includes other CFM implementations because those implementations will simply ignore the Exploratory Linktrace messages.

Exploratory Linktrace is initiated at the request of the administrator, and results in the local MEP sending a multicast Exploratory Linktrace message. Each MP in the network that receives the message sends an Exploratory Linktrace reply. MIPs that receive the message also forward it on. The initiating MEP uses all the replies to create a tree of the overall network topology.



This figure show an example of the Exploratory Linktrace message flow between MEPs.

Figure 7: Exploratory Linktrace Messages and Replies

To avoid overloading the originating MEP with replies in a large network, responding MPs delay sending their replies for a random amount of time, and that time increases as the size of the network increases.

In a large network, there will be a corresponding large number of replies and the resulting topology map will be equally large. If only a part of the network is of interest, for example, because a problem has already been narrowed down to a small area, then the Exploratory Linktrace can be "directed" to start at a particular MP. Replies will thus only be received from MPs beyond that point in the network. The replies are still sent back to the originating MEP.

Delay and Jitter Measurement (ITU-T Y.1731)

The router supports one-way and two-way delay measurement using two packet types:

- Delay Measurement Message (DMM)
- Delay Measurement Response (DMR)

These packets are unicast similar to loopback messages. The packets carry timestamps generated by the system time-of-day clock to support more accurate delay measurement, and also support an SLA manageability front-end. Beginning in Cisco IOS XR Release 4.1, the DDM & DDR packets carry timestamps derived from the DTI timing input on the clock-interface port on the RSP.

However, unlike loopback messages, these message types can also measure one-way delay and jitter either from destination to source, or from source to destination.

For more information about SLA, see the "Ethernet SLA" section on page 87.

Synthetic Loss Measurement (ITU-T Y.1731)

Synthetic Loss Measurement (SLM) is a mechanism that injects synthetic measurement probes, and measures the loss of these probes in order to measure the loss of real data traffic. Each probe packet carries a sequence number, and the sender increments the sequence number by one for each packet that is sent and the receiver can thereby detect the lost packets by looking for missing sequence numbers.

SLM packets contain two sequence numbers; one written by the initiator into the SLM and copied by the responder into the SLR, and the other allocated by the responder and written into the SLR. These are refered to as the source-to-destination (sd) sequence number and the destination-to-source (ds) sequence number respectively.

This figure shows an example of how the sequence numbers are used to calculate the Frame Loss Ratio (FLR) in each direction.

Figure 8: Synthetic Loss Measurement

Loss Measurement (ITU-T Y.1731)

Y.1731 Loss Measurement is a mechanism that measures the actual data traffic loss between a pair of MEPs in a point-to-point Ethernet service. This is in contrast to the Synthetic Loss Measurement, which measures the frame loss of synthetic frames. By using Y.1731 Loss Measurement, you can measure the one-way loss in each direction, for each priority class and also measure the loss aggregated across all priority classes.

To enable loss measurements to be made, each MEP maintains, for each priority class, both source-to-destination and destination-to-source frame counts for its peer MEPs.

There are two Loss Measurement Mechanisms (LMM); namely, single-ended and dual-ended. Cisco IOS XR Software supports only single-ended LMM.

MEP Cross-Check

MEP cross-check supports configuration of a set of expected peer MEPs so that errors can be detected when any of the known MEPs are missing, or if any additional peer MEPs are detected that are not in the expected group.

The set of expected MEP IDs in the service is user-defined. Optionally, the corresponding MAC addresses can also be specified. CFM monitors the set of peer MEPs from which CCMs are being received. If no CCMs are ever received from one of the specified expected peer MEPs, or if a loss of continuity is detected, then a cross-check "missing" defect is detected. Similarly, if CCMs are received from a matching MEP ID but with the wrong source MAC address, a cross-check "missing" defect is detected. If CCMs are subsequently received that match the expected MEP ID, and if specified, the expected MAC address, then the defect is cleared.



Note While loss of continuity can be detected for any peer MEP, it is only treated as a defect condition if cross-check is configured.

If cross-check is configured and CCMs are received from a peer MEP with a MEP ID that is not expected, this is detected as a cross-check "unexpected" condition. However, this is not treated as a defect condition.

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.

EFD

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

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

EFD changes this to allow CFM to act as the line protocol for Ethernet interfaces. This allows CFM to control the interface state so that if a CFM defect (such as AIS or loss of continuity) is detected with an expected peer MEP, the interface can be shut down. This not only stops any traffic flowing, but also triggers actions in any higher-level protocols to route around the problem. For example, in the case of Layer 2 interfaces, the MAC table would be cleared and MSTP would reconverge. For Layer 3 interfaces, the ARP cache would be cleared and potentially the IGP would reconverge.



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

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



Figure 9: CFM Error Detection and EFD Trigger

Flexible VLAN Tagging for CFM

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

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

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

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



Figure 10: Service Provider Network With Multiple VLANs and CFM

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

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

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

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

CFM on MC-LAG

CFM on Multi-Chassis Link Aggregation Groups is supported on the Cisco ASR 9000 Series Router in the following typical network environment:

- The customer edge (CE) device is a dual-homed device that is connected to two provider edge (PE) point-of-attachment (POA) devices. However, the dual-homed device operates without awareness of connectivity to multiple PEs.
- The two points of attachment at the PE form a redundancy group (RG), with one POA functioning as the active POA, and the other as the standby POA for the dual-homed device link.
- As with typical failover scenarios, if a failure occurs with the active POA, the standby POA takes over to retain the dual-homed device's connectivity to the network.

CFM on MC-LAG support can be qualified at two levels:

- CFM for the RG level—CFM context is per redundancy group and verifies connectivity for the entire RG.
- CFM for the POA level—CFM context is per point of attachment and verifies connectivity to a single POA.

Both levels of CFM support have certain restrictions and configuration guidelines that you must consider for successful implementation.

This section includes the following topics:

For more information about LAG and MC-LAG on the Cisco ASR 9000 Series Router, see the *Configuring Link Bundling* chapter in this guide.

RG-Level CFM

RG-level CFM is comprised of three areas of monitoring:

100

RG Downlink Monitoring

RG downlink monitoring uses CFM to verify connectivity between the dual-homed device and the RG.

To configure RG downlink monitoring, be sure that the following requirements are met:

- Down MEPs are configured on the bundle.
- Down MEPs on each POA are configured identically, using the same MEP ID and source MAC address.

This configuration has the following restrictions:

• The CCM loss time is greater than the failover time (typically 50 ms), due to the shortest CCM interval of 100 ms that is currently supported, which results in the shortest CCM loss time of 350 ms.

RG Uplink Monitoring

RG uplink monitoring uses CFM to verify connectivity from the active POA to the core.

To configure RG uplink monitoring, be sure that the following requirements are met:

- Up MEPs are configured on the bundle interface or bundle subinterface on each POA.
- Up MEPs on each POA are configured identically, using the same MEP ID and source MAC address.

End-to-End Service Monitoring

End-to-end service monitoring uses CFM to verify the end-to-end service between the dual-homed devices. To configure end-to-end service monitoring, be sure that the following requirements are met:

- A down MEP is configured on the dual-homed device bundle interface or bundle subinterface.
- If optional MIPs are configured, then each POA is configured with a MIP on the bundle.
- Each POA can have a MIP on the uplink interface (if native Ethernet is used).
- The active and standby POA is configured identically.

This configuration has the following restrictions:

• The MIP on the standby POA will not respond to loopback or linktrace requests.

POA-Level CFM

POA-level monitoing uses CFM to verify connectivity between the dual-homed device and a single POA. To configure POA-level CFM, be sure that the following requirements are met:

Down MEPs are configured on bundle members only.

This configuration has the following restrictions:

• POA-level monitoring is not supported on uplinks between a single POA and the core.

Supported Features for CFM on MC-LAG

CFM on MC-LAG supports these CFM features:

- All existing IEEE 802.1ag and Y.1731 functionality on the Cisco ASR 9000 Series Router is supported on an MC-LAG RG.
- CFM maintenance points are supported on an MC-LAG interface. Maintenance points on a standby link are put into standby state.
- Maintenance points in standby state receive CFM messages, but do not send or reply to any CFM messages.
- When a MEP transitions from active to standby, all CCM defects and alarms are cleared.
- Standby MEPs record remote MEP errors and timeouts, but do not report faults. This means that remote MEPs and their errors will appear in **show** commands, but no logs, alarms, MIB traps, or EFD are triggered and AIS messages are not sent.
- When a MEP transitions from standby to active, any CCM defects previously detected while the MEP
 was in standby are reapplied and immediate actions are taken (logs, alarms, MIB traps, EFD, and so on).
- CFM on MC-LAG supports the same scale for bundle interfaces that is supported on the Cisco ASR 9000 Series Router.

Restrictions for CFM on MC-LAG

To support CFM on MC-LAG, you must consider these restrictions and requirements:

- The CFM configuration must be the same on both the active and standby POAs.
- The CFM state is not synchronized between the two POAs. This can lead to flapping of the interface line protocol state on POA failover if EFD is configured. Fault alarms might also be delayed if a failover occurs just after a fault has been detected.
- POA-level CFM monitoring is not supported on a native Ethernet uplink interface.
- MEPs on bundle interfaces at level 0 are not supported.
- Loopback, linktrace, and Y.1731 SLA operations cannot be started from a MEP in standby state.
- Checks for configuration consistency of MEP IDs to ensure identical configuration of POAs is not supported.
- Y.1731 SLA statistics can be split between the two POAs if a failover occurs. An external network management system would need to collect and collate these statistics from the two POAs.

CFM Software Acceleration

Cisco ASR 9000 Series Router provides bundle-offload configuration for CFM under global configuration mode. This configuration enables CFM software acceleration to support aggressive CCM intervals of 10ms and higher CFM scale on bundle interfaces. This feature is applicable only for cases when the bundle members are configured under the Cisco ASR 9000 Enhanced Ethernet Line Card or higher generation line cards.

CFM would not work if the bundle members are also present on the Cisco ASR 9000 Ethernet line Cards. The CFM software acceleration feature is turned off by default. The bundle-offload feature acts as a knob to switch the feature either ON or OFF.

Connectivity Fault Management Hardware Offload

Connectivity Fault Management (CFM) hardware offload feature allows service providers to implement connectivity monitoring at 3.3 ms and 10 ms for physical interfaces. This feature helps detecting network failure within short time and achieve high network availability for Layer 2.

This feature is supported on following ASR 9000 line cards:

- Cisco ASR 9000 Series 5th Generation High-Density Line Cards
- Cisco ASR 9000 Series 4th Generation QSFP28 based dense 100GE Line Cards
- Cisco ASR 9000 Series 3rd Generation Ethernet Line Cards

See Configuring Ethernet OAM.

Limitations

Consider these points before implementing CFM hardware offload:

- This feature supports only physical interfaces and subinterfaces.
- This feature can only be used for down MEPs.
- This feature supports a single offloaded local MEP per interface.

Configuring CFM Hardware Offload

Configuring CFM hardware offload feature involves these steps:

1. Configure a down MEP with continuity-check enabled at a supported offload interval of 3.3ms or 10ms





A Continuity Check Message (CCM) interval should have crosscheck configuration.

Example:

```
ethernet cfm
domain dom1 level 1
service ser1 down-meps
continuity-check interval 3.3ms
mep crosscheck
mep-id 1
!
```

2. Apply the down MEP on an interface

Example:

```
interface GigabitEthernet0/0/0/0.1 l2 transport
encapsulation dotlq 22
   ethernet cfm
   mep domain doml service ser1 mep-id 100
```

Verification

To verify the CFM hardware offload configuration, use the **show ethernet cfm peer meps interface** show command:

```
Router# show ethernet cfm peer meps interface tenGigE 0/0/0/0.1 detail
Thu Apr 16 22:59:01.886 IST
Domain lsp down mep hw (level 3), Service lsp hw dn 1
Down MEP on TenGigE0/0/0/0.1 MEP-ID 100
_____
Peer MEP-ID 1, MAC 00c1.6440.ed17
  CFM state: Ok, for 00:00:44
  Received CCM handling offloaded to hardware
  Port state: Up
  CCMs received: 0
    Out-of-sequence:
                             0
                             0
    Remote Defect received:
    Wrong level:
                             0
    Cross-connect (wrong MAID): 0
    Wrong interval:
                             0
    Loop (our MAC received):
                             0
    Config (our ID received):
                             0
  Last CCM received:
    Level: 3, Version: 0, Interval: 3.3ms
    Sequence number: 4464, MEP-ID: 1
    MAID: String: lsp down mep hw, String: lsp hw dn 1
    Chassis ID: Local: Z13-9006; Management address: 'Not specified'
    Port status: Up, Interface status: Up
Router# show ethernet cfm local meps interface tenGigE 0/0/0/0.1 verbose
Thu Apr 16 23:01:14.454 IST
Domain lsp down mep_hw (level 3), Service lsp_hw_dn_1
Down MEP on TenGigE0/0/0/0.1 MEP-ID 100
_____
 Interface state: Up MAC address: 10b3.d672.86a8
 Peer MEPs: 1 up, 0 with errors, 0 timed out (archived)
 Cross-check errors: 0 missing, 0 unexpected
 CCM generation enabled: Yes, 3.3ms (Remote Defect detected: No)
                       CCM processing offloaded to hardware
 AIS generation enabled: No
 Sending AIS:
                       No
 Receiving AIS:
                      No
 Packet
           Sent Received
 -----
                  _____
 CCM
            17157
                           0 (out of seq: 0)
```

The show ethernet cfm local meps interface tenGigE 0/0/0/0.1 verbose command displays the number of cross-check errors, and CCM packets sent to the peer. The command also confirms the CFM hardware offload feature is enabled with 3.3ms offload interval.

```
Router# show ethernet cfm summary
CFM Summary for 0/0/CPU0
```

Domains	24
Services	12236
Total CCM rate (pps)	4111
Local Meps	4004
Operational	4004
Down MEPs	3

Up MEPs	4001
Offloaded	4004
3.3ms	1
10ms	0
Disabled (misconfiguration)	0
Disabled (resource limit)	0
Disabled (operational error)	0
Peer MEPs	4003
Operational	4003
Defect detected	0
No defect detected	4003
Timed out	0
MIPs	4000
Interfaces	4004
Bridge domains/Xconnects	4401
Traceroute cache entries	0
Traceroute cache replies	0
CCM Learning database entries	4215
BNM Enabled Links	0
ISSU Role	Primary

The show ethernet cfm summary command shows the number of hardware offloaded sessions with 3.3ms and 10ms offload intervals.

Ethernet SLA

Customers require their service providers to conform to a Service Level Agreement (SLA). Consequently, service providers must be able to monitor the performance characteristics of their networks. Similarly, customers also want to monitor the performance characteristics of their networks. Cisco provides Y.1731 performance monitoring using the Cisco Ethernet SLA feature.

An SLA defines a set of criteria that guarantees a minimum level of service for customers using a service provider network. The criteria can cover many different areas, including latency, jitter, frame loss, and availability.

The Cisco Ethernet SLA feature conforms to these standards:

- IEEE 802.1ag
- ITU-T Y.1731

The Cisco Ethernet SLA feature provides the architecture to monitor a network at Layer 2. This architecture provides functions such as collecting, storing, displaying, and analyzing SLA statistics. These SLA statistics can be stored and displayed in various ways, so that statistical analysis can be performed.

Ethernet SLA provides the framework for performing the following major functions of performance monitoring:

· Sending probes consisting of one or more packets to measure performance

Ethernet SLA provides a flexible mechanism for sending SLA probes to measure performance. Probes can consist of either CFM loopback or CFM delay measurement packets. Options are available to modify how often the packets are sent, and to specify the attributes of the probe packets such as the size and priority.

Scheduling of operations consisting of periodic probes.

A flexible mechanism is provided by Ethernet SLA to specify how often each probe should be executed, how long it should last, and when the first probe should start. Probes can be scheduled to run back-to-back to provide continuous measurements, or at a defined interval ranging from once a minute to once a week.

• Collecting and storing results.

Ethernet SLA provides flexibility to specify which performance parameters should be collected and stored for each measurement probe. Performance parameters include frame delay and jitter (inter-frame delay variation). For each performance parameter, either each individual result can be stored, or the results can be aggregated by storing a counter of the number of results that fall within a particular range. A configurable amount of historical data can also be stored as well as the latest results.

Analyzing and displaying results.

Ethernet SLA performs some basic statistical analysis on the collected results, such as calculating the minimum, maximum, mean and standard deviation. It also records whether any of the probe packets were lost or misordered, or if there is any reason why the results may not be a true reflection of the performance (for example if a big jump in the local time-of-day clock was detected during the time when the measurements were being made).

Y.1731 Performance Monitoring

The ITU-T Y.1731 standard defines several mechanisms that can be used for performance monitoring in Carrier Ethernet networks. These are the measurement mechanisms that were defined in the standard:

Delay Measurement: This can be used to accurately measure frame delay by exchanging CFM frames containing timestamps, and to measure inter-frame delay variation (jitter) by comparing consecutive delay measurements. Delay Measurement messages can be used to perform these measurements:

- Round-trip time
- · Round-trip Jitter
- One-way delay (both SD and DS)
- One-way jitter (both SD and DS)
- SLA Probe Packet corruption count
- · Out of order SLA probe packet count
- SLA probe packet loss

Loss Measurement: Loss Measurement is an extension to the existing Ethernet SLA feature; it adds the functionality for loss measurement defined in the Y.1731 and G.8021 ITU-T standards. This is used to accurately measure the loss of data traffic, by exchanging CFM frames containing sent and received frame counters. It is also used to measure the availability of the network by tracking periods of high loss over time. Loss Measurement messages can be used to perform these measurements:

- Data packet loss
- SLA probe packet loss
- Out of order SLA Probe packet count
- SLA Probe Packet corruption count

Synthetic Loss Measurement: The loss measurement mechanism defined in Y.1731 can only be used in point-to-point networks, and only works when there is sufficient data traffic flowing. The difficulties with the Y.1731 Loss Measurement mechanism was recognized across the industry and hence an alternative mechanism has been defined and standardized for measuring loss.

This alternative mechanism does not measure the loss of the actual data traffic, but instead injects synthetic CFM frames and measures the loss of these synthetic frames. Statistical analysis can then be used to give an approximation to the loss of data traffic. This technique is called Synthetic Loss Measurement. This has been included in the latest version of the Y.1731 standard. Synthetic Loss Measurement messages can be used to perform these measurements:

- One-way loss (Source to Destination)
- One-way loss (Destination to Source)

Loopback: This is not primarily targetted at performance monitoring, but can be used to approximate round-trip delay and jitter, such as when the peer device does not support delay measurement. Loopback messages can be used to perform these measurements:

- Round-trip time
- Round-trip jitter
- SLA probe packet corruption count
- Out of order SLA probe packet count
- SLA probe packet loss

Loss Measurement Terminology

These are the commonly used terminology in Loss Measurement Mechanism:

- **Single-ended**: A mechanism where device A sends a measurement packet to device B, which in turn sends a response back to device A. All calculations and results are done on device A.
- **Dual-ended**: A mechanism where device A sends a measurement packet to device B, which does not send a response. All calculations and results are done on device B.
- **One-way**: A measurement of the performance of packets flowing in one direction, from device A to device B, or from device B to device A.
- **Two-way**: A measurement of the performance of packets flowing from device A to device B, and back to device A.
- Forwards: A one-way measurement from the initiator (device A) to the receiver, or responder (device B).
- Backwards: A one-way measurement from the responder (device B) to the initiator (device A).

Note Cisco IOS XR Software supports only single-ended LMM.

Loss Measurement Performance Attributes

These are two primary attributes that can be calculated based on loss measurements:

- Frame Loss Ratio (FLR)
- Availability

Frame Loss Ratio is the ratio of lost packets to sent packets:

(<num_sent > - <num_rcvd>)/(<num_sent>)

It is normally expressed as a percentage. The accuracy of the measurement depends majorly on the number of packets sent.

Availability is a complex attribute, typically measured over a long period of time, such as weeks or months. The intent of this performance attribute is to measure the proportion of time when there was prolonged high loss. Cisco IOS XR Software does not track the availability.

Limitations of Data Loss Measurement

- 1. Data loss measurement cannot be used in a multipoint service; it can only be used in a peer-to-peer service.
- 2. As a Loss Measurement Reply (LMR) contains no sequence IDs, the only field, which can be used to distinguish to which probe a given LMR corresponds, is the priority level. Also, the priority level is the only field that can determine whether the LMR is in response to an on-demand or proactive operation. This limits the number of Loss Measurement probes that can be active at a time for each local MEP to 16.
- **3.** As loss measurements are made on a per-priority class basis, QoS policies, which alter the priority of packets processed by the network element, or re-order packets can affect the accuracy of the calculations. For the highest accuracy, packets must be counted after any QoS policies have been applied.
- **4.** The accuracy of data loss measurement is highly dependent on the number of data packets that are sent. If the volume of data traffic is low, errors with the packet counts might be magnified. If there is no data traffic flowing, no loss measurement performance attributes can be calculated. If aggregate measurements are taken, then only 2 probes can be active at the same time: one proactive and one on-demand.
- **5.** The accuracy of data loss measurement is highly dependent on the accuracy of platform-specific packet counters. Due to hardware limitations, it may not be possible to achieve completely accurate packet counters, especially if QoS policies are applied to the packets being counted.
- 6. Performing data loss measurement can have an impact on the forwarding performance of network elements; this is because of the need to count, as well as forward the packets.
- 7. Before starting any LMM probes, it is necessary to allocate packet counters for use with LMM on both ends (assuming both ends are running Cisco IOS XR Software).

Ethernet SLA Concepts

To successfully configure the Cisco Ethernet SLA feature, you should understand the following concepts:

Loss Measurement Terminology

A *statistic* in Ethernet SLA is a single performance parameter. These statistics can be measured by Ethernet SLA:

- Round-trip delay
- Round-trip jitter
- · One-way delay from source to destination
- · One-way jitter from source to destination

- One-way frame loss from source to destination
- One-way delay from destination to source
- One-way jitter from destination to source
- One-way frame loss from destination to source

Note Not all statistics can be measured by all types of packet. For example, one-way statistics cannot be measured when using CFM loopback packets.

Ethernet SLA Measurement Packet

An Ethernet SLA *measurement packet* is a single protocol message and corresponding reply that is sent on the network for the purpose of making SLA measurements. These types of measurement packet are supported:

 CFM Delay Measurement (Y.1731 DMM/DMR packets)—CFM delay measurement packets contain timestamps within the packet data that can be used for accurate measurement of frame delay and jitter. These packets can be used to measure round-trip or one-way statistics; however, the size of the DMM/DMR packets cannot be modified.



Note

From Cisco IOS XR Release 4.3.x onwards, you can configure the Ethernet SLA profile to use Y.1731 DMM v1 frames. The restriction of 150 configured Ethernet SLA operations for each CFM MEP is removed not only for profiles using DMM frames, but also for profiles using the other supported Y.1731 frame types, such as loopback measurement and synthetic loss measurement. For interoperability purposes, it is still possible to configure operations to use DMM v0 frames. This is done by specifying a type of **cfm-delay-measurement-v0** on the **ethernet SLA profile** command. The limit of 150 configured operations for each CFM MEP still applies in this case.

- CFM loopback (LBM/LBR)—CFM loopback packets are less accurate, but can be used if the peer device does not support DMM/DMR packets. Only round-trip statistics can be measured because these packets do not contain timestamps. However, loopback packets can be padded, so measurements can be made using frames of a specific size.
- CFM Synthetic Loss Measurement (Y.1731 SLM/SLR packets)—SLM packets contain two sequence numbers; one written by the initiator into the SLM and copied by the responder into the SLR, and the other allocated by the responder and written into the SLR. These are referred to as the source-to-destination (sd) sequence number and the destination-to-source (ds) sequence number respectively.



Note

Because SLM is a statistical sampling technique, there may be some variance of the measured value around the actual loss value. Also, the accuracy of the measurement is improved by using more SLM packets for each FLR calculation.

 CFM Loss Measurement (Y.1731 LMM/LMR packets)— As LMMs and LMRs contain no sequence ID, there is a limited set of data that can be used to distinguish different Loss Measurement operations, limiting the number of concurrent operations for each MEP.

Ethernet SLA Sample

A *sample* is a single result—a number—that relates to a given statistic. For some statistics such as round-trip delay, a sample can be measured using a single measurement packet. For other statistics such as jitter, obtaining a sample requires two measurement packets.

Ethernet SLA Probe

A *probe* is a sequence of measurement packets used to gather SLA samples for a specific set of statistics. The measurement packets in a probe are of a specific type (for example, CFM delay measurement or CFM loopback) and have specific attributes, such as the frame size and priority.



Note

A single probe can collect data for different statistics at the same time, using the same measurement packets (for example, one-way delay and round-trip jitter).

Ethernet SLA Burst

Within a probe, measurement packets can either be sent individually, or in bursts. A *burst* contains two or more packets sent within a short interval apart. Each burst can last up to one minute, and bursts can follow each other immediately to provide continuous measurement within the probe.

For statistics that require two measurement packets for each sample (such as jitter), samples are only calculated based on measurement packets in the same burst. For all statistics, it is more efficient to use bursts than to send individual packets.



Note If bursts are configured back to back, so as to cause a continuous and uninterrupted flow of SLA packets, then packets at the end of one burst and the start of the next are used in Loss Measurement calculations.

Ethernet SLA Schedule

An Ethernet SLA *schedule* describes how often probes are sent, how long each probe lasts, and at what time the first probe starts.



Note

If probes are scheduled back to back, so as to cause a continuous and uninterrupted flow of SLA packets, then packets at the end of one probe and the start of the next are used in Loss Measurement calculations.

Ethernet SLA Bucket

For a particular statistic, a *bucket* is a collection of results that were gathered during a particular period of time. All of the samples for measurements that were initiated during the period of time represented by a bucket are stored in that bucket. Buckets allow results from different periods of time to be compared (for example, peak traffic to off-peak traffic).

By default, a separate bucket is created for each probe; that is, the bucket represents the period of time starting at the same time as the probe started, and continuing for the duration of the probe. The bucket will therefore contain all the results relating to measurements made by that probe.

Ethernet SLA Aggregation Bin

Rather than storing each sample separately within a bucket, an alternative is to aggregate the samples into bins. An *aggregation bin* is a range of sample values, and contains a counter of the number of samples that were received that fall within that range. The set of bins forms a histogram. When aggregation is enabled, each bucket contains a separate set of bins. See this figure.

Ethernet SLA Operation Profile

An *operation profile* is a configuration entity that defines the following aspects of an operation:

- What packet types to send and in what quantities (probe and burst configuration)
- · What statistics to measure, and how to aggregate them
- When to schedule the probes

An operation profile by itself does not cause any packets to be sent or statistics collected, but is used to create operation instances.

Ethernet SLA Operation

An *operation* is an instance of a given operation profile that is actively collecting performance data. Operation instances are created by associating an operation profile with a given source (an interface and MEP) and with a given destination (a MEP ID or MAC address). Operation instances exist for as long as the configuration is applied, and they run for an indefinite duration on an ongoing basis.

Ethernet SLA On-Demand Operation

An *on-demand operation* is a method of Ethernet SLA operation that can be run on an as-needed basis for a specific and finite period of time. This can be useful in situations such as when you are starting a new service or modifying the parameters for a service to verify the impact of the changes, or if you want to run a more detailed probe when a problem is detected by an ongoing scheduled operation.

On-demand operations do not use profiles and have a finite duration. The statistics that are collected are discarded after a finite time after the operation completes (two weeks), or when you manually clear them.

On-demand operations are not persistent so they are lost during certain events such as a card reload or Minimal Disruptive Restart (MDR).

Statistics Measurement and Ethernet SLA Operations Overview

Table 11: Feature History Table

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

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.

By default, there is a separate bucket for each probe. The time period is determined by how long the probe lasts (configured by the **probe**, **send** (**SLA**), and **schedule** (**SLA**) commands). You can modify the size of buckets so that you can have more buckets per probe or fewer buckets per probe (less buckets allows the results from multiple probes to be included in the same bucket). Changing the size of the buckets for a given metric clears all stored data for that metric. All existing buckets are deleted and new buckets are created.

Scheduled SLA operation profiles run indefinitely, according to a configured schedule, and the statistics that are collected are stored in a rolling buffer, where data in the oldest bucket is discarded when a new bucket needs to be recorded.

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.

Configuration Overview of Scheduled Ethernet SLA Operations

When you configure a scheduled Ethernet SLA operation, you perform these basic steps:

- 1. Configure global profiles to define how packets are sent in each probe, how the probes are scheduled, and how the results are stored.
- 2. Configure operations from a specific local MEP to a specific peer MEP using these profiles.



Note Certain Ethernet SLA configurations use large amounts of memory which can affect the performance of other features on the system. For more information, see the "Configuring Ethernet SLA" section on page 128.

Ethernet LMI

The Cisco ASR 9000 Series Router supports the Ethernet Local Management Interface (E-LMI) protocol as defined by the *Metro Ethernet Forum, Technical Specification MEF 16, Ethernet Local Management Interface (E-LMI), January 2006* standard.

E-LMI runs on the link between the customer-edge (CE) device and the provider-edge (PE) device, or User Network Interface (UNI), and provides a way for the CE device to auto-configure or monitor the services offered by the PE device (see this figure).

Figure 11: E-LMI Communication on CE-to-PE Link



E-LMI is an asymmetric protocol whose basic operation involves the User-facing PE (uPE) device providing connectivity status and configuration parameters to the CE using STATUS messages in response to STATUS ENQUIRY messages sent by the CE to the uPE.

E-LMI Messaging

The E-LMI protocol as defined by the MEF 16 standard, defines the use of only two message types—STATUS ENQUIRY and STATUS.

These E-LMI messages consist of required and optional fields called information elements, and all information elements are associated with assigned identifiers. All messages contain the Protocol Version, Message Type, and Report Type information elements, followed by optional information elements and sub-information elements.

E-LMI messages are encapsulated in 46- to 1500-byte Ethernet frames, which are based on the IEEE 802.3 untagged MAC-frame format. E-LMI frames consist of the following fields:

- Destination address (6 bytes)—Uses a standard MAC address of 01:80:C2:00:00:07.
- Source address (6 bytes)-MAC address of the sending device or port.
- E-LMI Ethertype (2 bytes)—Uses 88-EE.
- E-LMI PDU (46–1500 bytes)—Data plus 0x00 padding as needed to fulfill minimum 46-byte length.
- CRC (4 bytes)—Cyclic Redundancy Check for error detection.

For more details about E-LMI messages and their supported information elements, refer to the Metro Ethernet Forum, Technical Specification MEF 16, Ethernet Local Management Interface (E-LMI), January 2006.

Cisco-Proprietary Remote UNI Details Information Element

The E-LMI MEF 16 specification does not define a way to send proprietary information.

To provide additional information within the E-LMI protocol, the Cisco IOS XR software implements a Cisco-proprietary information element called Remote UNI Details to send information to the CE about remote UNI names and states. This information element implements what is currently an unused identifier from the E-LMI MEF 16 specification.

To ensure compatibility for future implementations of E-LMI should this identifier ever be implemented in the standard protocol, or for another reason, you can disable transmission of the Remote UNI information element using the **extension remote-uni disable** command.

E-LMI Operation

The basic operation of E-LMI consists of a CE device sending periodic STATUS ENQUIRY messages to the PE device, followed by mandatory STATUS message responses by the PE device that contain the requested information. Sequence numbers are used to correlate STATUS ENQUIRY and STATUS messages between the CE and PE.

The CE sends the following two forms of STATUS ENQUIRY messages called Report Types:

- E-LMI Check—Verifies a Data Instance (DI) number with the PE to confirm that the CE has the latest E-LMI information.
- Full Status—Requests information from the PE about the UNI and all EVCs.

The CE device uses a polling timer to track sending of STATUS ENQUIRY messages, while the PE device can optionally use a Polling Verification Timer (PVT), which specifies the allowable time between transmission of the PE's STATUS message and receipt of a STATUS ENQUIRY from the CE device before recording an error.

In addition to the periodic STATUS ENQUIRY/STATUS message sequence for the exchange of E-LMI information, the PE device also can send asynchronous STATUS messages to the CE device to communicate changes in EVC status as soon as they occur and without any prompt by the CE device to send that information.

Both the CE and PE devices use a status counter (N393) to determine the local operational status of E-LMI by tracking consecutive errors received before declaring a change in E-LMI protocol status.

Supported E-LMI PE Functions on the Cisco ASR 9000 Series Router

The Cisco ASR 9000 Series Router serves as the PE device for E-LMI on a MEN, and supports the following PE functions:

• Supports the E-LMI protocol on Ethernet physical interfaces that are configured with Layer 2 subinterfaces as Ethernet Flow Points (EFPs), which serve as the EVCs about which the physical interface reports status to the CE. The Cisco IOS XR software does not support a specific manageability context for an Ethernet Virtual Connection (EVC).



• CE functions

Unidirectional Link Detection Protocol

Unidirectional Link Detection (UDLD) is a single-hop physical link protocol for monitoring an ethernet link, including both point-to-point and shared media links. This is a Cisco-proprietary protocol to detect link problems, which are not detected at the physical link layer. This protocol is specifically targeted at possible wiring errors, when using unbundled fiber links, where there can be a mismatch between the transmitting and receiving connections of a port.

UDLD Operation

UDLD works by exchanging protocol packets between the neighboring devices. In order for UDLD to work, both devices on the link must support UDLD and have it enabled on respective ports.

UDLD sends an initial PROBE message on the ports where it is configured. Once UDLD receives a PROBE message, it sends periodic ECHO (hello) messages. Both messages identify the sender and its port, and also contain some information about the operating parameters of the protocol on that port. They also contain the device and port identifiers for any neighbor devices that the local device has heard from, on the port. Similarly, each device gets to know where it is connected and where its neighbors are connected.

This information can then be used to detect faults and miswiring conditions. The protocol operates an aging mechanism by means of which information from neighbors that is not periodically refreshed is eventually timed out. This mechanism can also be used for fault detection.

A FLUSH message is used to indicate that UDLD is disabled on a port, which causes the peers to remove the local device from their neighbor cache, to prevent it from being aged out.

If a problem is detected, UDLD disables the affected interface and also notifies the user. This is to avoid further network problems beyond traffic loss, such as loops which are not detected or prevented by STP.

Types of Fault Detection

UDLD can detect these types of faults:

- Transmit faults These are cases where there has been a failure in transmitting packets from the local port to the peer device, but packets continue to be received from the peer. These faults are caused by failure of the physical link (where notification at layer 1 of unidirectional link faults is not supported by the media) as well as packet path faults on the local or peer device.
- Miswiring faults These are cases where the receiving and transmitting sides of a port on the local device are connected to different peer ports (on the same device or on different devices). This can occur when using unbundled fibers to connect fiber optic ports.
- Loopback faults These are cases where the receiving and transmitting sides of a port are connected to each other, creating a loopback condition. This can be an intentional mode of operation, for certain types of testing, but UDLD must not be used in these cases.
- Receive faults The protocol includes a heartbeat that is transmitted at a negotiated periodic interval to the peer device. Missed heartbeats can therefore be used to detect failures on the receiving side of the link (where they do not result in interface state changes). These could be caused by a unidirectional link with a failure only affecting the receiving side, or by a link which has developed a bidirectional fault. This detection depends on reliable, regular packet transmission by the peer device. For this reason, the UDLD protocol has two (configurable) modes of operation which determine the behavior on a heartbeat timeout. These modes are described in the section UDLD Modes of Operation, page 98.

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

This is a scenario that happens in a receive fault condition. Aging of UDLD information happens when the port that runs UDLD does not receive UDLD packets from the neighbor port for duration of hold time. The hold time for the port is dictated by the remote port and depends on the message interval at the remote side. The shorter the message interval, the shorter is the hold time and the faster the detection. 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 some physical issue or duplex mismatch. Such packet drop does not mean that the link is unidirectional and UDLD in normal mode does not disable such link.

It is important to choose the right message interval in order 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 equal to approximately three times the message interval. So, when using default UDLD timers, UDLD does not time out the link faster than the STP aging time.

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 hellos will continue to be sent and hellos from neighbors 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 TLV 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 those the local device can see and the port will be disabled.
- Bidirectional detected: UDLD hello messages are exchanged successfully in both directions, the port is operating correctly.

Ethernet Data Plane Loopback

The Ethernet Data Plane Loopback feature allows you to test services and throughput of an Ethernet port or a device using a test generator. You can verify the maximum rate of frame transmission with no frame loss. This feature allows bidirectional throughput measurement, and on-demand or out-of-service (intrusive) operation during service turn-ups. This feature can be used for testing during service turn-ups and troubleshooting of services after a turn-up.

If you need to test a service while it is live, you can do this without disrupting any of the live data traffic. To achieve this, you can use test traffic that differs from live data traffic. For example, the traffic from a test generator can contain the source MAC address of the test generator, or test traffic may be assigned a particular Class of Service (CoS). Irrespective of the method used, the device looping back the traffic must be able to filter out the test traffic and leave the data traffic untouched.



Note Configuring Ethernet Data Plane Loopback on a device does not indicate the start of an actual session.

Features Supported for Ethernet Data Plane Loopback

The support that the Ethernet Data Plane Loopback feature provides is:

- Locally-enabled Ethernet Data Plane Loopback on all Ethernet interface types, such as physical and bundle interfaces and sub-interfaces.
- In the case of Layer 2 interfaces, support for these types of looping back of traffic:
 - External loopback All traffic received on the ingress interface is blindly sent out of the egress interface.
 - Internal loopback All traffic received on the egress interface is blindly injected into the ingress interface.
- In the case of Layer 3 interfaces, only external loopback is supported.
- When a Bundle interface is placed into loopback, traffic on all bundle link members are looped back.
- MAC address must always be swapped on looped-back traffic.
- Allows the application of multiple filters to loopback only a subset of traffic received by an interface and only drop the corresponding reverse-direction traffic.
- Provides an option to specify a time period after which the loopback is automatically terminated.
- Supports at least 100 simultaneous loopback sessions across the system.

Limitations of Ethernet Data Plane Loopback

These are the limitations of Ethernet Data Plane Loopback (EDPL):

- Layer 3 interfaces including pseudowires are not supported in internal EDPL.
- The first generation Cisco ASR 9000 Ethernet Line Cards are not supported.
- The fifth generation Cisco ASR 9000 series high density ethernet line cards do not support internal EDPL
- Virtual interfaces such as BVI are not supported.
- Filtering based on LLC-OUI is not supported.
- A maximum of 50 simultaneous loopback sessions are supported for each Network Processor on the linecard.
- LAG bundles that are member of Satellite nV interface over bundle inter-chassis link (also known as LAG over LAG bundles) are not supported.

Configuring Ethernet Data Plane Loopback

Perform these steps to configure Ethernet Data Plane Loopback.

- Configure Ethernet Data Plane Loopback
- Start an Ethernet Data Plane Loopback Session

/* Enable the privileged EXEC mode. Enter your password if prompted and then configure the terminal*/

Router# enable Router# configure

/* Specify the interface on which you want to enable EDPL and specify if the ethernet loopback permit must be internal or external. */

```
Router(config)# interface
TenGigE 0/1/0/0
Router(config-if-srv)# ethernet loopback permit external
or
Router(config-if-srv)# ethernet loopback permit internal
Router# end
Router(config-if-srv)# commit
```

/* Start an EDPL session */

```
RP/0/RSP0/CPU0:router#ethernet loopback start local interface TenGigE0/0/0/29 external destination mac-address 008a.9678.781c
Router#ethernet loopback start local interface TenGigE0/0$
```

```
LC/0/0/CPU0:Jan 11 14:27:57.086 IST: EDPL-MA[354]: %L2-ETH_LB-6-SESSION_STARTED : Session 4 on interface TenGigE0/0/0/29 has successfully started. Session on interface TenGigE0/0/0/29 successfully created with ID 4.
```

Configures ethernet loopback externally or internally on an interface. External loopback allows loopback of traffic from wire. Internal loopback allows loopback of traffic from the bridge domain.

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

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

• 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.

Running Configuration

This example shows external loopback on the TenGig Ethernet 0/0/0/29 interface:

```
interface TenGigE0/0/0/29
ethernet loopback
  permit external
  permit internal
!
!
Router# ethernet loopback start local interface TenGigE0/0/0/29 external destination
mac-address 008a.9678.781c
Router#ethernet loopback start local interface TenGigE0/0$
LC/0/0/CPU0:Jan 11 14:27:57.086 IST: EDPL-MA[354]: %L2-ETH_LB-6-SESSION_STARTED : Session
4 on interface TenGigE0/0/0/29 has successfully started.
Session on interface TenGigE0/0/0/29 successfully created with ID 4.
```

Verification

Verify that ethernet loopback is active on TenGigE0/0/0/29 interface.

Router:ABC#show ethernet loopback permitted Wed Jan 11 14:29:03.503 IST Local Loopback Interface Direction _____ TenGigE0/0/0/29 External, Internal Latching Loopback Interface Direction _____ Loopback Controller Interface _____ Router:ABC#show ethernet loopback active Wed Jan 11 14:29:07.191 IST

Local: TenGigE0/0/0/29, ID 4	
======================================	External
Time out:	0h5m0s
Time left:	0h3m49s
Status:	Active
Filters:	
Dot1Q:	Any
inner-dot1Q:	Any
Source MAC Address:	Any
Destination MAC Address:	008a.9678.781c
Ethertype:	Any
Class of Service:	Any

Router:ABC#ethernet loopback start local interface TenGigE0/0/0/29 external destination mac-addRP/0/RSP0/CPU0:PE3-ASR9901#ethernet loopback stop local interface tenGigE 0/0/0/29 id 4 Wed Jan 11 14:29:31.753 IST

```
LC/0/0/CPU0:Jan 11 14:29:32.022 IST: EDPL-MA[354]: %L2-ETH_LB-6-SESSION_STOPPED : Attempt to stop session 4 on interface TenGigE0/0/0/29 has completed. Session with ID 4 on interface TenGigE0/0/0/29 successfully stopped.
```

Ethernet Data Plane Loopback on Satellite nV System

The Ethernet Data Plane Loopback (EDPL) is implemented on the Satellite nV System as shown in this illustration.



Figure 12: EDPL on Satellite nV System

The internal and external EDPL on satellite are realized as follows:

• **Internal Loopback**: The MAC address swap happens on the host and the frame actually gets looped back from the satellite where Layer 1 loopback needs to be turned on at the port. As the entire port is looped back on the satellite, the internal loopback for satellite ports cannot loopback or filter specific sub-interface sessions on the port. You need to enable both EDPL and port L1 internal loopback on the satellite port for this functionality.
• External Loopback: The external loopback is currently implemented entirely on the host because of the need to perform MAC address swap.

Limitations of Ethernet Data Plane Loopback on nV Satellite System

Following are the limitations of Ethernet Data Plane Loopback (EDPL) on nV Satellite system:

- LAG bundles that are member of Satellite nV interface over bundle inter-chassis link (also known as LAG over LAG bundles) are not supported.
- If ICL is non-redundant(non-bundle), ethernet loopback internal on satellite access interfaces is not supported.

Configuring EDPL on nV Satellite System

To enable and start Ethernet Data Plane Loopback (EDPL) on an nV satellite system, perform the following steps:

/* Enable ethernet internal loopback on satellite interfaces */

```
RP/0/RSP0/CPU0:router#config
RP/0/RSP0/CPU0:router(config)#interface gi200/0/0/8
RP/0/RSP0/CPU0:router(config-if)#ethernet loopback permit internal
RP/0/RSP0/CPU0:router(config-if)#commit
RP/0/RSP0/CPU0:router(config-if)#end
```

/* Enable ethernet external loopback on satellite interfaces */

```
RP/0/RSP0/CPU0:router#config
RP/0/RSP0/CPU0:router(config)#interface gi200/0/0/10
RP/0/RSP0/CPU0:router(config-if)#ethernet loopback permit external
RP/0/RSP0/CPU0:router(config-if)#commit
RP/0/RSP0/CPU0:router(config-if)#end
```

/* Start an EDPL session */

```
RP/0/RSP0/CPU0:router#ethernet loopback start local interface gigabitEthernet 200/0/0/10 external destination mac-address 70b3.1778.ef41 Session on interface GigabitEthernet200/0/0/10 successfully created with ID 2.
```



Note To stop an EDLP session, use the **ethernet loopback stop local interface** *< name > id < id > command.*

Running Configuration

The following configuration displays EDPL on an nv satellite, a bundle interface, and on an interface.

```
RP/0/RSP0/CPU0:router#show run nv satellite 200
nv
satellite 200
type asr9000v2
ip address 100.100.1.3
description sat 200
!
!
RP/0/RSP0/CPU0:router#show run interface bundle-ether 22
interface Bundle-Ether22
ipv4 point-to-point
ipv4 unnumbered Loopback10
nv
```

```
satellite-fabric-link satellite 200
remote-ports GigabitEthernet 0/0/0-39
!
!
!
RP/0/RSP0/CPU0:router#show run interface TenGigE0/3/0/2/2
bundle id 22 mode on
!
RP/0/RSP0/CPU0:router#show run interface TenGigE0/3/0/5/2
interface TenGigE0/3/0/5/2
bundle id 22 mode on
!
```

Verification

Verify that the internal and external loopback are active.

Interface

RP/0/RSP0/CPU0:router#show ethernet loopback active

```
Local: GigabitEthernet200/0/0/10, ID 2
```

Direction:	External
Time out:	0h5m0s
Time left:	0h4m53s
Status:	Active
Filters:	
Dot1Q:	Any
inner-dot1Q:	Any
Source MAC Address:	Any
Destination MAC Address:	70b3.1778.ef41
Ethertype:	Any
Class of Service:	Any

Verify that the satellite is ready.

RP/0/RSP0/CPU0:router#show nv satellite status satellite 200
Satellite 200
-----Status: Connected (Stable)
Type: asr9000v2
Description: sat 200
Displayed device name: Sat200
MAC address: 70b3.1778.ef38
IPv4 address: 100.100.1.3 (VRF: default)
Serial Number: CAT2243U002
Remote version: Compatible (latest version)
ROMMON: 128.1 (Latest)

How to Configure Ethernet OAM

This section provides these configuration procedures:

Configuring Ethernet Link OAM

Custom Ethernet Link OAM (ELO) settings can be configured and shared on multiple interfaces by creating an ELO 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 ELO profile is attached to an interface, individual Ethernet Link OAM features can be configured separately on the interface to override the profile settings when desired.

This section describes how to configure an ELO profile and attach it to an interface.

Configuring an Ethernet Link OAM Profile

Perform these steps to configure an Ethernet Link OAM (ELO) profile.



IOS-XR CLI refers to Ethernet Link OAM as **ethernet oam** in both profile and interface configurations.

SUMMARY STEPS

- 1. configure
- 2. ethernet oam profile profile-name
- 3. link-monitor
- 4. symbol-period window window
- 5. symbol-period threshold low threshold high threshold
- 6. frame window window
- 7. frame threshold low threshold high threshold
- 8. frame-period window window
- 9. frame-period threshold lowthreshold high threshold
- 10. frame-seconds window window
- 11. frame-seconds threshold low threshold high threshold
- 12. exit

- 13. mib-retrieval
- **14. connection timeout** *< timeout>*
- **15.** hello-interval {100ms|1s}
- **16.** mode {active|passive}
- **17.** require-remote mode {active|passive}
- 18. require-remote mib-retrieval
- **19.** action capabilities-conflict {disable | efd | error-disable-interface}
- **20.** action critical-event {disable | error-disable-interface}
- **21.** action discovery-timeout {disable | efd | error-disable-interface}
- **22.** action dying-gasp {disable | error-disable-interface}
- 23. action high-threshold {error-disable-interface | log}
- 24. action session-down {disable | efd | error-disable-interface}
- 25. action session-up disable
- 26. action uni-directional link-fault {disable | efd | error-disable-interface}
- **27.** action wiring-conflict {disable | efd | log}
- 28. uni-directional link-fault detection
- 29. commit
- 30. end

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure terminal	
Step 2	ethernet oam profile profile-name	Creates a new Ethernet Link OAM (ELO) profile and
	Example:	enters Ethernet OAM configuration mode.
	<pre>RP/0/RSP0/CPU0:router(config)# ethernet oam profile Profile_1</pre>	
Step 3	link-monitor	Enters the Ethernet OAM link monitor configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router(config-eoam)# link-monitor	
Step 4	symbol-period window window	(Optional) Configures the window size (in milliseconds)
	Example:	for an Ethernet OAM symbol-period error event. The IEEE 802.3 standard defines the window size as a number of
RP/0/RSP0/CPU0:router(config-eoam-lm) # symbol-period window 60000symbols rathe can be conver interface spectrum	symbols rather than a time duration. These two formats can be converted either way by using a knowledge of the nterface speed and encoding.	

I

	Command or Action	Purpose
		The range is 1000 to 60000.
		The default value is 1000.
Step 5	<pre>symbol-period threshold low threshold high threshold Example: RP/0/RSP0/CPU0:router(config-eoam-lm)#</pre>	(Optional) Configures the thresholds (in symbols) that trigger an Ethernet OAM symbol-period error event. The high threshold is optional and is configurable only in conjunction with the low threshold.
	symbol-period threshold ppm low 1 high 1000000	The range is 0 to 60000000.
		The default low threshold is 1.
Step 6	frame window window	(Optional) Configures the frame window size (in
	Example:	milliseconds) of an OAM frame error event.
	RP/0/RSP0/CPU0:router(config-eoam-lm)# frame	The range is from 1000 to 60000.
	window 6000	The default value is 1000.
Step 7	frame threshold low threshold high threshold	(Optional) Configures the thresholds (in symbols) that
	Example: RP/0/RSP0/CPU0:router(config-eoam-lm)# frame threshold low 10000000 high 60000000	triggers an Ethernet OAM frame error event. The high threshold is optional and is configurable only in conjunction with the low threshold.
		The range is from 0 to 60000000.
		The default low threshold is 1.
Step 8	frame-period window window	(Optional) Configures the window size (in milliseconds)
01000	Example: RP/0/RSP0/CPU0:router(config-eoam-lm)# frame-period window 60000 RP/0/RSP0/CPU0:router(config-coam-lm)#	for an Ethernet OAM frame-period error event. The IEEE 802.3 standard defines the window size as number of frames rather than a time duration. These two formats can be converted either way by using a knowledge of the interface speed. Note that the conversion assumes that all
	frame-period window milliseconds 60000	frames are of the minimum size.
		The range is from 1000 to 60000.
		The default value is 1000.
		Note The only accepted values are multiples of the line card interface module specific polling interval, that is, 1000 milliseconds for most line card interface modules.
Step 9	<pre>frame-period threshold lowthreshold high threshold Example: RP/0/RSP0/CPU0:router(config-eoam-lm)# frame-period threshold ppm low 100 high 1000000</pre>	(Optional) Configures the thresholds (in errors per million frames) that trigger an Ethernet OAM frame-period error event. The frame period window is defined in the IEEE specification as a number of received frames, in our implementation it is x milliseconds. The high threshold is optional and is configurable only in conjunction with the low threshold. The range is from 1 to 1000000.

	Command or Action	Purpose
		The default low threshold is 1.
		To obtain the number of frames, the configured time interval is converted to a window size in frames using the interface speed. For example, for a 1Gbps interface, the IEEE defines minimum frame size as 512 bits. So, we get a maximum of approximately 1.5 million frames per second. If the window size is configured to be 8 seconds (8000ms) then this would give us a Window of 12 million frames in the specification's definition of Errored Frame Window.
		The thresholds for frame-period are measured in errors per million frames. Hence, if you configure a window of 8000ms (that is a window of 12 million frames) and a high threshold of 100, then the threshold would be crossed if there are 1200 errored frames in that period (that is, 100 per million for 12 million).
Step 10	frame-seconds window window	(Optional) Configures the window size (in milliseconds) for the OAM frame-seconds error event.
	Example:	The range is 10000 to 900000.
	RP/0/RSP0/CPU0:router(config-eoam-lm)#	The default value is 60000.
		Note The only accepted values are multiples of the line card interface module specific polling interval, that is, 1000 milliseconds for most line card interface modules.
Step 11	frame-seconds threshold low threshold high threshold	(Optional) Configures the thresholds (in seconds) that
	Example:trigger a fra value can b threshold value can b threshold value can bRP/0/RSP0/CPU0:router(config-eoam-lm)# frame-seconds threshold low 3 high 900The range i	trigger a frame-seconds error event. The high threshold value can be configured only in conjunction with the low threshold value.
		The range is 1 to 900
		The default value is 1.
Step 12	exit	Exits back to Ethernet OAM mode.
	Example:	
	RP/0/RSP0/CPU0:router(config-eoam-lm)# exit	
Step 13	mib-retrieval	Enables MIB retrieval in an Ethernet OAM profile or on
	Example:	an Ethernet OAM interface.
	RP/0/RSP0/CPU0:router(config-eoam)# mib-retrieval	
Step 14	connection timeout <timeout> Example:</timeout>	Configures the connection timeout period for an Ethernet OAM session. as a multiple of the hello interval.

I

	Command or Action	Purpose
	RP/0/RSP0/CPU0:router(config-eoam)# connection timeout 30	The range is 2 to 30. The default value is 5.
Step 15	<pre>hello-interval {100ms 1s} Example: RP/0/RSP0/CPU0:router(config-eoam)# hello-interval 100ms</pre>	Configures the time interval between hello packets for an Ethernet OAM session. The default is 1 second (1s).
Step 16	<pre>mode {active passive} Example: RP/0/RSP0/CPU0:router(config-eoam) # mode passive</pre>	Configures the Ethernet OAM mode. The default is active.
Step 17	<pre>require-remote mode {active passive} Example: RP/0/RSP0/CPU0:router(config-eoam)# require-remote mode active</pre>	Requires that active mode or passive mode is configured on the remote end before the OAM session comes up.
Step 18	<pre>require-remote mib-retrieval Example: RP/0/RSP0/CPU0:router(config-eoam)# require-remote mib-retrieval</pre>	Requires that MIB-retrieval is configured on the remote end before the OAM session comes up.
Step 19	<pre>action capabilities-conflict {disable efd error-disable-interface} Example: RP/0/RSP0/CPU0:router(config-eoam)# action capabilities-conflict efd</pre>	 Specifies the action that is taken on an interface when a capabilities-conflict event occurs. The default action is to create a syslog entry. Note If you change the default, the log keyword option is available in Interface Ethernet OAM configuration mode to override the profile setting and log the event for the interface when it occurs.
Step 20	action critical-event {disable error-disable-interface} Example: RP/0/RSP0/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. Note If you change the default, the log keyword option is available in Interface Ethernet OAM configuration mode to override the profile setting and log the event for the interface when it occurs.

	Command or Action	Purpose
Step 21	<pre>action discovery-timeout {disable efd error-disable-interface} Example: RP/0/RSP0/CPU0:router(config-eoam)# action discovery-timeout efd</pre>	 Specifies the action that is taken on an interface when a connection timeout occurs. The default action is to create a syslog entry. Note If you change the default, the log keyword option is available in Interface Ethernet OAM configuration mode to override the profile setting and log the event for the interface when it occurs.
Step 22	<pre>action dying-gasp {disable error-disable-interface} Example: RP/0/RSP0/CPU0:router(config-eoam)# action dying-gasp error-disable-interface</pre>	 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. Note If you change the default, the log keyword option is available in Interface Ethernet OAM configuration mode to override the profile setting and log the event for the interface when it occurs.
Step 23	<pre>action high-threshold {error-disable-interface log} Example: RP/0/RSP0/CPU0:router(config-eoam)# action high-threshold error-disable-interface</pre>	 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. Note If you change the default, the disable keyword option is available in Interface Ethernet OAM configuration mode to override the profile setting and take no action at the interface when the event occurs.
Step 24	action session-down {disable efd error-disable-interface} Example: RP/0/RSP0/CPU0:router(config-eoam)# action session-down efd	 Specifies the action that is taken on an interface when an Ethernet OAM session goes down. Note If you change the default, the log keyword option is available in Interface Ethernet OAM configuration mode to override the profile setting and log the event for the interface when it occurs.
Step 25	<pre>action session-up disable Example: RP/0/RSP0/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. Note If you change the default, the log keyword option is available in Interface Ethernet OAM configuration mode to override the profile setting and log the event for the interface when it occurs.

	Command or Action	Purpose
Step 26	action uni-directional link-fault {disable efd error-disable-interface}	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.
		 Note If you change the default, the log keyword option is available in Interface Ethernet OAM configuration mode to override the profile setting and log the event for the interface when it occurs.
Step 27	action wiring-conflict {disable efd log} Example:	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.
	RP/0/RSP0/CPU0:router(config-eoam)# action session-down efd	Note • If you change the default, the error-disable-interface keyword option is available in Interface Ethernet OAM configuration mode to override the profile setting and put the interface into error-disable state when the event occurs.
Step 28	uni-directional link-fault detection Example:	Enables detection of a local, unidirectional link fault and sends notification of that fault to an Ethernet OAM peer.
	RP/0/RSP0/CPU0:router(config-eoam)# uni-directional link-fault detection	
Step 29	commit	Saves the configuration changes to the running
	Example:	session.
	RP/0/RSP0/CPU0:router(config-if)# commit	
Step 30	end	Ends the configuration session and exits to the EXEC
	Example:	mode.
	RP/0/RSP0/CPU0:router(config-if)# end	

Attaching an Ethernet OAM Profile to an Interface

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

SUMMARY STEPS

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

6. end

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure terminal	
Step 2	interface [FastEthernet HundredGigE TenGigE] interface-path-id	Enters interface configuration mode and specifies the Ethernet interface name and notation <i>rack/slot/module/port</i> .
	Example:	Note
	<pre>RP/0/RSP0/CPU0:router(config)# interface TenGigE 0/1/0/0</pre>	• The example indicates an 8-port 10-Gigabit Ethernet interface in modular services card slot 1.
Step 3	ethernet oam	Enables Ethernet OAM and enters interface Ethernet OAM
	Example:	configuration mode.
	RP/0/RSP0/CPU0:router(config-if)# ethernet oam	
Step 4	profile profile-name	Attaches the specified Ethernet OAM profile (profile-name),
	Example:	and all of its configuration, to the interface.
	<pre>RP/0/RSP0/CPU0:router(config-if-eoam) # profile Profile_1</pre>	
Step 5	commit	Saves the configuration changes to the running configuration
	Example:	file and remains within the configuration session.
	RP/0/RSP0/CPU0:router(config-if)# commit	
Step 6	end	Ends the configuration session and exits to the EXEC mode.
	Example:	
	RP/0/RSP0/CPU0:router(config-if)# end	

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 [GigabitEthernet | TenGigE] interface-path-id
- 3. ethernet oam
- **4.** *interface-Ethernet-OAM-command* RP/0/RSP0/CPU0:router(config-if-eoam)# action capabilities-conflict error-disable-interface
- 5. commit
- 6. end

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure terminal	
Step 2	interface [GigabitEthernet TenGigE] interface-path-id	Enters interface configuration mode and specifies the
	Example:	Ethernet interface name and notation <i>rack/slot/module/port</i> .
	RP/0/RSP0/CPU0:router(config)# interface TenGigE 0/1/0/0	 Note The example indicates an 8-port 10-Gigabit Ethernet interface in modular services card slot 1.
Step 3	ethernet oam	Enables Ethernet OAM and enters interface Ethernet OAM
	Example:	configuration mode.
	RP/0/RSP0/CPU0:router(config-if)# ethernet oam	
Step 4	<i>interface-Ethernet-OAM-command</i> RP/0/RSP0/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

I

	Command or Action	Purpose
		one of the supported commands on the platform in interface Ethernet OAM configuration mode.
Step 5	commit	Saves the configuration changes to the running configuration
	Example:	file and remains within the configuration session.
	RP/0/RSP0/CPU0:router(config-if)# commit	
Step 6	end	Ends the configuration session and exits to the EXEC mode.
	Example:	
	RP/0/RSP0/CPU0:router(config-if)# end	

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/RSP0/CPU0:router# show ethernet oam conf.	iguration
Thu Aug 5 22:07:06.870 DST	
GigabitEthernet0/4/0/0:	
Hello interval:	1s
Link monitoring enabled:	Y
Remote loopback enabled:	N
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
Remote loopback action:	Log
Require remote mode:	Ignore
Require remote MIB retrieval:	N
Require remote loopback support:	N
Require remote link monitoring:	N

Configuring Ethernet CFM

To configure Ethernet CFM, perform the following tasks:

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	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	ethernet cfm	Enters Ethernet Connectivity Fault Management (CFM)
	Example:	configuration mode.
	RP/0/RSP0/CPU0:router(config)# ethernet cfm	
Step 3	domain domain-name level level-value [id [null] [dnsDNS-name] [mac H.H.H] [string string]]	Creates and names a container for all domain configurations and enters CFM domain configuration mode.
	Example:	The level must be specified.
	RP/0/RSP0/CPU0:router(config-cfm)# domain Domain_One level 1 id string D1	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	traceroute cache hold-time minutes size entries	(Optional) Sets the maximum limit of traceroute cache
	Example:	entries or the maximum time limit to hold the traceroute cache entries. The default is 100 minutes and 100 entries.
	<pre>RP/0/RSP0/CPU0:router(config-cfm)# traceroute cache hold-time 1 size 3000</pre>	
Step 5	end or commit	Saves configuration changes.
	Example:	• When you use the end command, the system prompts you to commit changes:

Purpose
<pre>un)# commit Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre>
• Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
• Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
• Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring Services for a CFM Maintenance Domain

You can configure up to 32000 CFM services for a maintenance domain.

Before you begin

To configure services for 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. service** *service-name* {**bridge group** *bridge-domain-group* **bridge-domain** *bridge-domain-name* | **down-meps** | **xconnect group** *xconnect-group-name* **p2p** *xconnect-name*}[**id** [**icc-based** *icc-string umc-string*] | [**string** *text*] | [**number** *number*] | [**vlan-id** *id-number*] | [**vpn-id** *oui-vpnid*]]
- 5. end or commit

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	

I

	Command or Action	Purpose
Step 2	ethernet cfm	Enters Ethernet CFM configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router(config)# ethernet cfm	
Step 3	domain domain-name level level-value [id [null] [dns DNS-name] [mac H.H.H] [string string]]	Creates and names a container for all domain configurations at a specified maintenance level, and enters CFM domain configuration mode.
	EXAMPLE: RP/0/RSP0/CPU0:router(config-cfm)# domain Domain_One level 1 id string D1	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	<pre>service service-name {bridge group bridge-domain-group bridge-domain bridge-domain-name down-meps xconnect group xconnect-group-name p2p xconnect-name}[id [icc-based icc-string umc-string] [string text] [number number] [vlan-id id-number] [vpn-id oui-vpnid]]</pre>	Configures and associates a service with the domain and enters CFM domain service configuration mode. You can specify that the service is used only for down MEPs, or associate the service with a bridge domain or xconnect where MIPs and up MEPs will be created. The <i>id</i> sets the short MA name
	Example:	The Iu sets the short WA hand.
	RP/0/RSP0/CPU0:router(config-cfm-dmn)# service Bridge_Service bridge group BD1 bridge-domain B1	
Step 5	end or commit	Saves configuration changes.
	Example:	• When you use the end command, the system prompts you to commit changes:
	RP/0/RSP0/CPU0:router(config-cim-dmn-svc)# commit	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.

Enabling and Configuring Continuity Check for a CFM Service

The Cisco ASR 9000 Series Router supports Continuity Check as defined in the IEEE 802.1ag specification, and supports CCMs intervals of 100 ms and longer. The overall packet rates for CCM messages are up to 16000 CCMs-per-second sent, and up to 16000 CCMs-per-second received, per card.

Note If Ethernet SLA is configured, the overall combined packet rate for CCMs and SLA frames is 16000 frames-per-second in each direction, per card.

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

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	ethernet cfm	Enters Ethernet Connectivity Fault Management (CFM)
	Example:	configuration mode.
	RP/0/RSP0/CPU0:router(config)# ethernet cfm	
Step 3	domain domain-name level level-value [id [null] [dnsDNS-name] [mac H.H.H] [string string]]	Creates and names a container for all domain configurations and enters the CFM domain configuration mode.
	Example:	The level must be specified.
	RP/0/RSP0/CPU0:router(config-cfm)# domain Domain_One level 1 id string D1	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.

I

	Command or Action	Purpose
Step 4	<pre>service service-name {bridge group bridge-domain-group bridge-domain bridge-domain-name down-meps xconnect group xconnect-group-name p2p xconnect-name}[id [icc-based icc-string umc-string] [string text] [number number] [vlan-id id-number] [vpn-id oui-vpnid]]</pre>	Configures and associates a service with the domain and enters CFM domain service configuration mode. You can specify that the service is used only for down MEPs, or associate the service with a bridge domain or xconnect where MIPs and up MEPs will be created.
	Example:	The id sets the short MA name.
	RP/0/RSP0/CPU0:router(config-cfm-dmn)# service Bridge_Service bridge group BD1 bridge-domain B1	
Step 5	continuity-check interval <i>time</i> [loss-threshold <i>threshold</i>] Example:	(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.
	RP/0/RSP0/CPU0:router(config-cfm-dmn-svc)# continuity-check interval 100m loss-threshold 10	
Step 6	continuity-check archive hold-time <i>minutes</i> Example:	(Optional) Configures how long information about peer MEPs is stored after they have timed out.
	RP/0/RSP0/CPU0:router(config-cfm-dmn-svc)# continuity-check archive hold-time 100	
Step 7	continuity-check loss auto-traceroute Example:	(Optional) Configures automatic triggering of a traceroute when a MEP is declared down.
	RP/0/RSP0/CPU0:router(config-cfm-dmn-svc)# continuity-check loss auto-traceroute	
Step 8	end or commit	Saves configuration changes.
	Example:	• When you use the end command, the system prompts you to commit changes:
	RP/U/RSPU/CPUU:router(config-cim-amn-svc)# commit	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.

Command or Action	Purpose
	• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring Automatic MIP Creation for a CFM Service

For more information about the algorithm for creating MIPs, see the "MIP Creation" section on page 72. To configure automatic MIP creation for a CFM service, complete the following steps:

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	ethernet cfm	Enters the Ethernet Connectivity Fault Management (CFM)
	Example:	configuration mode.
	RP/0/RSP0/CPU0:router# ethernet cfm	
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>]]	Creates and names a container for all domain configurations and enters the CFM domain configuration mode.
	Example:	The level must be specified.
	RP/0/RSP0/CPU0:router(config-cfm)# domain Domain_One level 1 id string D1	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> { bridge group <i>bridge-domain-group</i> bridge-domain <i>bridge-domain-name</i> down-meps	Configures and associates a service with the domain and enters CFM domain service configuration mode. You can

	Command or Action	Purpose
	<pre>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]] Example: RP/0/RSP0/CPU0:router(config-cfm-dmn)# service Bridge_Service_bridge_group_BD1_bridge=domain_B1</pre>	specify that the service is used only for down MEPs, or associate the service with a bridge domain or xconnect where MIPs and up MEPs will be created. The id sets the short MA name.
Step 5	<pre>mip auto-create {all lower-mep-only} {ccm-learning} Example: RP/0/RSP0/CPU0:router(config-cfm-dmn-svc)# mip auto-create all ccm-learning</pre>	(Optional) Enables the automatic creation of MIPs in a bridge domain or xconnect. ccm-learning option enables CCM learning for MIPs created in this service. This must be used only in services with a relatively long CCM interval of at least 100 ms. CCM learning at MIPs is disabled by default.
Step 6	<pre>end or commit Example: RP/0/RSP0/CPU0:router(config-cfm-dmn-svc)# commit </pre>	 Saves configuration changes. When you use 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 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* {**bridge group** *bridge-domain-group* **bridge-domain** *bridge-domain-name* | **down-meps** | **xconnect group** *xconnect-group-name* **p2p** *xconnect-name*}[**id** [**icc-based** *icc-string umc-string*] | [**string** *text*] | [**number** *number*] | [**vlan-id** *id-number*] | [**vpn-id** *oui-vpnid*]]
- 5. mep crosscheck
- 6. mep-id mep-id-number [mac-address mac-address]
- 7. end or commit

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	ethernet cfm	Enters the Ethernet Connectivity Fault Management (CFM)
	Example:	configuration mode.
	RP/0/RSP0/CPU0:router# ethernet cfm	
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>]]	Creates and names a container for all domain configurations and enters the CFM domain configuration mode.
	Example:	The level must be specified.
	RP/0/RSP0/CPU0:router(config-cfm)# domain Domain_One level 1 id string D1	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 service-name {bridge group bridge-domain-group bridge-domain bridge-domain-name down-meps xconnect group xconnect-group-name p2p xconnect-name}[id [icc-based icc-string umc-string] [string text] [number number] [vlan-id id-number] [vpn-id oui-vpnid]]	Configures and associates a service with the domain and enters CFM domain service configuration mode. You can specify that the service is used only for down MEPs, or associate the service with a bridge domain or xconnect where MIPs and up MEPs will be created. The id sets the short MA name
	Example:	The R sets the short WA hame.
	RP/0/RSP0/CPU0:router(config-cfm-dmn)# service Bridge_Service bridge group BD1 bridge-domain B1	
Step 5	mep crosscheck	Enters CFM MEP crosscheck configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router(config-cfm-dmn-svc)# mep crosscheck mep-id 10	
Step 6	mep-id mep-id-number [mac-address mac-address]	Enables cross-check on a MEP.

L

Command or Action	Purpose
Example: RP/0/RSP0/CPU0:router(config-cfm-xcheck) # mep-id 10	 Note Repeat this command for every MEP that you want included in the expected set of MEPs for cross-check.
Step 7 end or commit	Saves configuration changes.
<pre>Example: RP/0/RSP0/CPU0:router(config-cfm-xcheck)# commit</pre>	 When you use 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

Configuring Other Options for a CFM Service

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

SUMMARY STEPS

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

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	ethernet cfm	Enters the Ethernet Connectivity Fault Management (CFM)
	Example:	configuration mode.
	RP/0/RSP0/CPU0:router# ethernet cfm	
Step 3	domain domain-name level level-value [id [null] [dns DNS-name] [mac H.H.H] [string string]]	Creates and names a container for all domain configurations and enters the CFM domain configuration mode.
	Example:	The level must be specified.
	RP/0/RSP0/CPU0:router(config-cfm)# domain Domain_One level 1 id string D1	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	<pre>service service-name {bridge group bridge-domain-group bridge-domain bridge-domain-name down-meps xconnect group xconnect-group-name p2p xconnect-name}[id [icc-based icc-string umc-string] [string text] [number number] [vlan-id id-number] [vpn-id oui-vpnid]] Example: RP/0/RSP0/CPU0:router(config-cfm-dmn) # service Bridge Service bridge group BD1 bridge-domain B1</pre>	Configures and associates a service with the domain and enters CFM domain service configuration mode. You can specify that the service is used only for down MEPs, or associate the service with a bridge domain or xconnect where MIPs and up MEPs will be created. The id sets the short MA name.
Sten 5	maximum-mens number	(Optional) Configures the maximum number (2 to 8190)
0104 0	Example:	of MEPs across the network, which limits the number of peer MEPs recorded in the database.
	RP/0/RSP0/CPU0:router(config-cfm-dmn-svc)# maximum-meps 1000	
Step 6	log {ais continuity-check errors continuity-check mep changes crosscheck errors efd}	(Optional) Enables logging of certain types of events.
	Example:	
	<pre>RP/0/RSP0/CPU0:router(config-cfm-dmn-svc)# log continuity-check errors</pre>	
Step 7	end or commit	Saves configuration changes.

L

Purpose
• When you use the end command, the system prompts you to commit changes:
<pre># commit 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

When you configure CFM MEPs, consider these guidelines:

- Up to 32000 local MEPs are supported per card.
- CFM maintenance points can be created on these interface types:
 - All physical Ethernet interfaces (except for the RSP Management interfaces).
 - Ethernet bundle interfaces.
 - All physical and bundle Ethernet sub-interfaces, providing the encapsulation is configured according to the following guidelines:

Frames are only matched based on VLAN IDs and CoS bits.

Frames are not matched using VLAN "any."

If frames are untagged, then the interface configuration on the Cisco ASR 9000 Series Router is such that there is no ambiguity on the sub-interface to which the untagged frame must be classified.

- Ethernet bundle member interfaces—Only down MEPs at level 0 can be created.
- CFM maintenance points can be created on both Layer 2 and Layer 3 interfaces. On L3 interfaces, only down MEPs can be created.
- A new configuration under the MEP submode called loss-measurement counters is used to allocate the packet counters used for LMM.

Restrictions

When you configure MEPs, consider these restrictions:

- Maintenance points at level 0 are not supported on bundle interfaces.
- CFM on Cisco IOS XR Software does not support a tag stack of more than two tags.
- If a subinterface is configured that matches untagged Ethernet frames (for example, by configuring the **encapsulation default** command), then you can not create a down MEP on the underlying physical or bundle interface.
- Up MEPs are not supported on Layer 3 interfaces.
- CCM packet must not go through L3VPN cloud.
- LBM/LBR packet must not go through L3VPN cloud.
- LTM/LTR packet must not go through L3VPN cloud.
- DMM/DMR packet must not go through L3VPN cloud.
- SLM/SLR packet must not go through L3VPN cloud.
- LMM/LMR packet must not go through L3VPN cloud.

SUMMARY STEPS

- 1. configure
- 2. interface {GigabitEthernet | TenGigE} interface-path-id
- 3. interface {GigabitEthernet | TenGigE | Bundle-Ether} interface-path-id.subinterface
- 4. vrf vrf-name
- 5. interface {FastEthernet | GigabitEthernet | TenGigE} interface-path-id
- 6. ethernet cfm
- 7. mep domain domain-name service service-name mep-id id-number
- **8.** cos cos
- 9. end or commit

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	interface {GigabitEthernet TenGigE} interface-path-id	Type of Ethernet interface on which you want to create a
	Example:	MEP. Enter GigabitEthernet or TenGigE and the physical interface or virtual interface.
	<pre>RP/0//CPU0:router(config)# interface gigabitethernet 0/1/0/1</pre>	 Note Use the show interfaces command to see a list of all interfaces currently configured on the router.

I

	Command or Action	Purpose
		For more information about the syntax for the router, use the question mark (?) online help function.
Step 3	interface {GigabitEthernet TenGigE Bundle-Ether} interface-path-id.subinterface Example:	Type of Ethernet interface on which you want to create a MEP. Enter GigabitEthernet , TenGigE , or Bundle-Ether and the physical interface or virtual interface followed by the subinterface path ID.
	gigabitethernet 0/1/0/1	Naming notation is <i>interface-path-id.subinterface</i> . The period in front of the subinterface value is required as part of the notation.
		the question mark (?) online help function.
Step 4	vrf vrf-name	Configures a VRF instance and enters VRF configuration
	Example:	mode. For more information on configuring VRF interfaces, refer the <i>Connecting MPLS VPN Customers</i> section in the
	RP/0/RSP0/CPU0:router(config-if)# vrf vrf_A	Cisco ASR 9000 Series MPLS Layer 3 VPN Configuration Guide.
Step 5	interface {FastEthernet GigabitEthernet TenGigE} interface-path-id	Type of Ethernet interface on which you want to create a MEP. Enter FastEthernet , GigabitEthernet or TenGigE
	Example:	and the physical interface or virtual interface.
	<pre>RP/0/RSP0/CPU0:router(config)# interface gigabitethernet 0/1/0/1</pre>	• Use the show interfaces command to see a list of all interfaces currently configured on the router.
		For more information about the syntax for the router, use the question mark (?) online help function.
Step 6	ethernet cfm	Enters interface Ethernet CFM configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router(config-if)# ethernet cfm	
Step 7	mep domain <i>domain-name</i> service <i>service-name</i> mep-id <i>id-number</i>	Creates a maintenance end point (MEP) on an interface and enters interface CFM MEP configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router(config-if-cfm)# mep domain Dm1 service Sv1 mep-id 1	
Step 8	cos cos	(Optional) Configures the class of service (CoS) (from
	Example:	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
	RP/0/RSP0/CPU0:router(config-if-cfm-mep)# cos 7	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 9	end or commit	Saves configuration changes.
	Example:	• When you use the end command, the system prompts you to commit changes:
	RP/0/RSP0/CPU0:router(config-if-cfm-mep)# commit	
		Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:
		• Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring Y.1731 AIS

This section has the following step procedures:

Configuring AIS in a CFM Domain Service

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

SUMMARY STEPS

- 1. configure
- 2. ethernet cfm
- 3. domain name level level
- 4. service name bridge group name bridge-domain name
- 5. service name xconnect group xconnect-group-name p2p xconnect-name
- 6. ais transmission [interval {1s|1m}][cos cos]
- 7. log ais
- 8. end or commit

DETAILED STEPS

Procedure

I

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0//CPU0:router# configure	
Step 2	ethernet cfm	Enters Ethernet CFM global configuration mode.
	Example:	
	RP/0//CPU0:router(config)# ethernet cfm	
Step 3	domain name level level	Specifies the domain and domain level.
	Example:	
	RP/0//CPU0:router(config-cfm)# domain D1 level 1	
Step 4	service name bridge group name bridge-domain name	Specifies the service, bridge group, and bridge domain.
	Example:	
	RP/0/RSP0/CPU0:router(config-cfm-dmn)# service S1 bridge group BG1 bridge-domain BD2	
Step 5	service name xconnect group xconnect-group-name p2p xconnect-name	Specifies the service and cross-connect group and name.
	Example:	
	RP/0//CPU0:router(config-cfm-dmn)# service S1 bridge group BG1 bridge-domain BD2	
Step 6	ais transmission [interval {1s 1m}][cos cos]	Configures Alarm Indication Signal (AIS) transmission for
	Example:	a Connectivity Fault Management (CFM) domain service.
	RP/0//CPU0:router(config-cfm-dmn-svc)# ais transmission interval 1m cos 7	
Step 7	log ais	Configures AIS logging for a Connectivity Fault
	Example:	Management (CFM) domain service to indicate when AIS or LCK packets are received.
	RP/0//CPU0:router(config-cfm-dmn-svc)# log ais	
Step 8	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts
	RP/0//CPU0:router(config-sla-prof-stat-cfg)# commit	you to commit enanges:
		Uncommitted changes found, commit them before

Command or Action	Purpose
	<pre>exiting(yes/no/cancel)? [cancel]:</pre>
	• Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
	• Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
	• Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
	• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring AIS on a CFM Interface

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

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0//CPU0:router# configure	
Step 2	interface gigabitethernet interface-path-id	Enters interface configuration mode.
	Example:	
	RP/0//CPU0:router# interface gigabitethernet 0/1/0/2	

	Command or Action	Purpose
Step 3	ethernet cfm	Enters Ethernet CFM interface configuration mode.
	Example:	
	RP/0//CPU0:router(config)# ethernet cfm	
Step 4	ais transmission up interval 1m cos cos	Configures Alarm Indication Signal (AIS) transmission on a Connectivity Fault Management (CFM) interface.
	Example:	
	<pre>RP/0//CPU0:router(config-if-cfm)# ais transmission up interval 1m cos 7</pre>	
Step 5	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	RP/0//CPU0:router(config-sla-prof-stat-cfg)# commit	
		Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:
		• Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring EFD for a CFM Service

To configure EFD for a CFM service, complete the following steps.

Restrictions

EFD is not supported on up MEPs. It can only be configured on down MEPs, within a particular service.

SUMMARY STEPS

- 1. configure
- **2**. ethernet cfm
- 3. domain domain-name level level-value
- 4. service service-name down-meps
- 5. efd

- 6. log efd
- 7. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0//CPU0:router# configure	
Step 2	ethernet cfm	Enters CFM configuration mode.
	Example:	
	RP/0//CPU0:router(config)# ethernet cfm	
Step 3	domain domain-name level level-value	Specifies or creates the CFM domain and enters CFM
	Example:	domain configuration mode.
	<pre>RP/0//CPU0:router(config-cfm-dmn)# domain D1 level 1</pre>	
Step 4	service service-name down-meps	Specifies or creates the CFM service for down MEPS and
	Example:	enters CFM domain service configuration mode.
	RP/0//CPU0:router(config-cfm-dmn)# service S1 down-meps	
Step 5	efd	Enables EFD on all down MEPs in the down MEPS service.
	Example:	
	RP/0//CPU0:router(config-cfm-dmn-svc)# efd	
Step 6	log efd	(Optional) Enables logging of EFD state changes on an
	Example:	interface.
	RP/0//CPU0:router(config-cfm-dmn-svc)# log efd	
Step 7	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts
	RP/0//CPU0:router(config-cfm-dmn-svc)# commit	you to commit changes:
		Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:

Command or Action	Purpose
	• 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 EFD Configuration

This example shows how to display all interfaces that are shut down because of Ethernet Fault Detection (EFD):

RP/0/RSP0/CPU0:router# show efd interfaces

Server VLAN MA =========== Interface Clients ------GigE0/0/0/0.0 CFM

Configuring Flexible VLAN Tagging for CFM

Use this procedure to set the number of tags in CFM packets from up MEPs to 1, in a CFM domain service.

SUMMARY STEPS

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

DETAILED STEPS

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

	Command or Action	Purpose
	RP/0/RSP0/CPU0:router# configure	
Step 2	ethernet cfm	Enters Ethernet CFM global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router(config)# ethernet cfm	
Step 3	domain name level level	Specifies the domain and domain level.
	Example:	
	<pre>RP/0/RSP0/CPU0:router(config-cfm) # domain D1 level 1</pre>	
Step 4	service name bridge group name bridge-domain name	Specifies the service, bridge group, and bridge domain.
	Example:	
	RP/0/RSP0/CPU0:router(config-cfm-dmn)# service S2 bridge group BG1 bridge-domain BD2	
Step 5	tags number	Specifies the number of tags in CFM packets from up MEPs.
	Example:	Currently, the only valid value is 1.
	RP/0/RSP0/CPU0:router(config-cfm-dmn-svc)# tags 1	
Step 6	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	RP/0/RSP0/CPU0:router(config-cfm-dmn-svc) # commit	
		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 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.
<pre>show ethernet cfm local maintenance-points domain name [service name] interface type interface-path-id] [mep mip]</pre>	Displays a list of local maintenance points.

Troubleshooting Tips

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

SUMMARY STEPS

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

DETAILED STEPS

Procedure

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

```
RP/0/RSP0/CPU0:router# ping ethernet cfm domain D1 service S1 mep-id 16 source
interface GigabitEthernet 0/0/0/0
```

```
Type escape sequence to abort.

Sending 5 CFM Loopbacks, timeout is 2 seconds -

Domain foo (level 2), Service foo

Source: MEP ID 1, interface GigabitEthernet0/0/0/0

Target: 0001.0002.0003 (MEP ID 16):

Running (5s) ...

Success rate is 60.0 percent (3/5), round-trip min/avg/max = 1251/1349/1402 ms

Out-of-sequence: 0.0 percent (0/3)

Bad data: 0.0 percent (0/3)

Received packet rate: 1.4 pps
```

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

RP/0/RSP0/CPU0:router# traceroute ethernet cfm domain D1 service S1 mep-id 16 source interface gigabitethernet 0/0/0/0

```
Traceroutes in domain D1 (level 4), service S1
Source: MEP-ID 1, interface GigabitEthernet0/0/0/0
```

```
_____
Traceroute at 2009-05-18 12:09:10 to 0001.0203.0402,
TTL 64, Trans ID 2:
                   Ingress MAC/name
Hop Hostname/Last
                                  Egress MAC/Name
                                                   Relav
__ ____
                                   _____
 1 ios
                   0001.0203.0400 [Down]
                                                    FDB
   0000-0001.0203.0400 Gi0/0/0/0
                                   0001.0203.0401 [Ok]
 2 abc
                                                   FDB
    ios
                                   Not present
 3 bcd
                   0001.0203.0402 [Ok]
                                                    Hit
                    GigE0/0
    abc
Replies dropped: 0
```

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

If the Relay field contains "MPDB" for any of the hops, then the target MAC address was not found in the bridge MAC learning table at that hop, and the result is relying on CCM learning. This result can occur under normal conditions, but it can also indicate a problem. If you used the **ping ethernet cfm** command before using the **traceroute ethernet cfm** command, then the MAC address should have been learned. If "MPDB" is appearing in that case, then this indicates a problem at that point in the network.

Configuring Ethernet SLA

This section describes how to configure Ethernet SLA.

Ethernet SLA Configuration Guidelines

∕!∖

Caution

n 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.
- The Cisco ASR 9000 Series Router supports SLA packet intervals of 100 ms and longer. If Ethernet SLA is configured, the overall combined packet rate for CCMs and SLA frames is 16000 frames-per-second in each direction, per card.
- 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.

Note When the **once** keyword is used for 'send burst' ('send burst once' rather than 'send burst every'), it stops the collection of statistics with the packets that cross probe boundaries.

The following procedure provides the steps to configure Ethernet Service Level Agreement (SLA) monitoring at Layer 2.

To configure SLA, perform the following tasks:

Configuring an SLA Operation Profile

To configure a profile, perform the following steps:

SUMMARY STEPS

- 1. configure
- 2. ethernet sla
- 3. profile *profile-name* type {cfm-delay-measurement | cfm-loopback | cfm-synthetic-loss-measurement |
- 4. cfm-loss-measurement}
- 5. end or commit

DETAILED STEPS

Step 1	configure
	Example:
	RP/0/RSP0/CPU0:router# configure
	Enters global configuration mode.
Step 2	ethernet sla
	Example:
	RP/0/RSP0/CPU0:router# ethernet sla
	Enters the SLA configuration mode.
Step 3	profile <i>profile-name</i> type {cfm-delay-measurement cfm-loopback cfm-synthetic-loss-measurement
Step 4	cfm-loss-measurement}
	Example:
	RP/0/RSP0/CPU0:router(config-sla)# profile Prof1 type cfm-loopback
	Creates an SLA operation profile and enters the SLA profile configuration mode.
Step 5	end or commit

Example:

RP/0/RSP0/CPU0:router(config-sla) # commit

Saves configuration changes.

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

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

- Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
- Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
- Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
- Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring a Schedule for an SLA Operation Probe in a Profile

This section describes how to configure a schedule for an SLA operation probe on an ongoing basis within an SLA profile. For information about how to configure a schedule for a limited, on-demand SLA operation, see the "Configuring an On-Demand SLA Operation" section on page 137.

To configure a schedule for an SLA operation probe, perform the following steps beginning in SLA profile configuration mode:

SUMMARY STEPS

- schedule every week on day [at hh:mm] [for duration {seconds | minutes | hours | days | week}] or schedule every day [at hh:mm] [for duration {seconds | minutes | hours | days | week}] or schedule every number {hours | minutes}[first at hh:mm[.ss]] [for duration {seconds | minutes | hours | days | week}]
- 2. end or commit

DETAILED STEPS

Procedure

 Step 1
 schedule every week on day [at hh:mm] [for duration {seconds | minutes | hours | days | week}] or schedule every day [at hh:mm] [for duration {seconds | minutes | hours | days | week}] or schedule every number {hours | minutes][first at hh:mm[.ss]] [for duration {seconds | minutes | hours | days | week}]

Example:

RP/0/RSP0/CPU0:router(config-sla-prof) # schedule every week on Monday at 23:30 for 1 hour
```
or
RP/0/RSP0/CPU0:router(config-sla-prof)# schedule every day at 11:30 for 5 minutes
or
RP/0/RSP0/CPU0:router(config-sla-prof)# schedule every 2 hours first at 13:45:01
or
```

RP/0/RSP0/CPU0:router(config-sla-prof)# schedule every 6 hours for 2 hours

Schedules an operation probe in a profile. A profile may contain only one schedule.

Note

The schedule start time starts after the configuration is committed and not at the time when the operation is configured.

For information on the *schedule* command behavior and usage guidelines, see *Interface and Hardware Component* Command Reference for Cisco ASR 9000 Series Routers, chapter Ethernet OAM Commands.

Step 2 end or commit

Example:

RP/0/RSP0/CPU0:router(config-sla-prof-stat-cfg) # commit

Saves configuration changes.

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

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

- Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
- Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
- Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
- Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring SLA Probe Parameters in a Profile

To configure SLA probe parameters in a profile, perform these steps beginning in SLA profile configuration mode:

SUMMARY STEPS

- 1. probe
- 2. send burst {every number {seconds | minutes | hours} | once} packet count packets interval number {seconds | milliseconds}
- **3**. or
- 4. send packet {every number {milliseconds | seconds | minutes | hours} | once}
- 5. packet size *bytes* [test pattern {hex 0xHHHHHHHH | pseudo-random}]

I

- 6. priority priority
- 7. synthetic loss calculation packets number
- 8. end or commit

DETAILED STEPS

Procedure

Step 1	probe
	Example:
	RP/0/RSP0/CPU0:router(config-sla-prof)# probe
	Enters the SLA profile probe configuration mode.
Step 2	send burst {every <i>number</i> {seconds minutes hours} once} packet count <i>packets</i> interval <i>number</i> {seconds milliseconds}
Step 3	or
Step 4	send packet {every number {milliseconds seconds minutes hours} once}
	Example:
	RP/0/RSP0/CPU0:router(config-sla-prof-pb)# send burst every 60 seconds packet count 100 interval 100 milliseconds
	or RP/0/RSP0/CPU0:router(config-sla-prof-pb)# send burst once packet count 2 interval 1 second or
	RP/0/RSP0/CPU0:router(config-sla-prof-pb)# send packet every 100 milliseconds
	Configures the number and timing of packets sent by a probe in an operations profile.
	Note When the once keyword for 'send burst' ('send burst once' rather than 'send burst every') is used, it stops the collection of statistics with the packets that cross probe boundaries.
Step 5	<pre>packet size bytes [test pattern {hex 0xHHHHHHHH pseudo-random}]</pre>
	Example:
	RP/0/RSP0/CPU0:router(config-sla-prof-pb)# packet size 9000
	Configures the minimum size (in bytes) for outgoing probe packets, including padding when necessary. Use the test pattern keyword to specify a hexadecimal string to use as the padding characters, or a pseudo-random bit sequence. The default padding is 0's. The packet size can be configured for SLM, loopback, and DMM/R probes.
Step 6	priority priority
	Example:
	RP/0/RSP0/CPU0:router(config-sla-prof-pb)# priority 7
	Configures the priority of outgoing SLA probe packets.

If the operation is running on an interface, which matches tagged traffic, then a priority value must be configured for the probe. This priority value must match the "on-the-wire" CoS value of the packets to be counted (after any tag rewrites).LMM packets are sent with this priority value as the CoS-value, and LMR packets must be received with the same CoS-value; otherwise, all LMRs are dropped. Note that this is the case even when aggregate counters are being collected.

If the operation is running on an interface which matches untagged traffic, then configuring a priority value is not permitted. In this case, only aggregate counters can be collected. When configuring data-loss measurement operations, configuration must also be applied to allocate the correct packet counters (matching the CoS values to be collected) on the interface, using the "loss-measurement counters" configuration under the MEP properties submode.

Step 7 synthetic loss calculation packets number

Example:

RP/0/RSP0/CPU0:router(config-sla-prof-pb)# synthetic loss calculation packets 25

Configures the number of packets that must be used to make each FLR calculation in the case of synthetic loss measurements. This item can only be configured for packet types that support synthetic loss measurement.

An FLR value is calculated for each discrete block of packets. For instance, if a value of 10 is configured, the first FLR value would be calculated based on packets 0 - 9 and the second FLR value based on packets 10 - 19, and so on.

Step 8 end or commit

Example:

RP/0/RSP0/CPU0:router(config-sla-prof-pb)# commit

Saves configuration changes.

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

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

- Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
- Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
- Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
- Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring SLA Statistics Measurement in a Profile

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

Before you begin

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 need to 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.

For more information about frequency and time synchronization, refer to the *Configuring Frequency* Synchronization on the Cisco ASR 9000 Series Router and the Configuring PTP on the Cisco ASR 9000 Series Router modules in the Cisco ASR 9000 Series Aggregation Services Router System Management Configuration Guide.

Restrictions

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

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

SUMMARY STEPS

- 1. 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}
- **2.** aggregate {bins count width [usec] width | none}
- 3. buckets size number probes
- 4. buckets archive *number*
- 5. end or commit

DETAILED STEPS

Procedure

 Step 1
 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}

Example:

RP/0/RSP0/CPU0:router(config-sla-prof) # statistics measure round-trip-delay

Enables the collection of SLA statistics, and enters SLA profile statistics configuration mode.

Step 2 aggregate {bins count width [usec] width | none}

Example:

RP/0/RSP0/CPU0:router(config-sla-prof-stat-cfg)# aggregate bins 100 width 10000

Configures 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, the default is aggregation disabled.

- 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.

Step 3 buckets size *number* probes

Configures the size of the buckets in which statistics are collected.

Step 4 buckets archive *number*

Example:

RP/0/RSP0/CPU0:router(config-sla-prof-stat-cfg) # buckets archive 50

Configures the number of buckets to store in memory.

Step 5 end or commit

Example:

RP/0/RSP0/CPU0:router(config-sla-prof-stat-cfg)# 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 displays aggregate bins configured with a range of 123 microseconds:

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

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:
                Samples Cum. Count
   Range
                                     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%)
                                        _
```

Configuring an SLA Operation

This section describes how to configure an ongoing SLA operation on a MEP using an SLA profile.

SUMMARY STEPS

- 1. interface [FastEthernet
- **2**. ethernet cfm
- 3. mep domain domain-name service service-name mep-id id-number
- 4. sla operation profile *profile-name* target {mep-id *id* | mac-address *mac-address*}
- 5. end or commit

DETAILED STEPS

Procedure

Step 1 interface [FastEthernet

Example:

RP/0/RSP0/CPU0:router(config-if) # interface gigabitethernet 0/1/0/1

Physical interface or virtual interface.

Note

• Use the **show interfaces** command to see a list of all interfaces currently configured on the router.

For more information about the syntax for the router, use the question mark (?) online help function.

Step 2 ethernet cfm

Example:

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

Enters interface CFM configuration mode.

Step 3 mep domain domain-name service service-name mep-id id-number

Example:

RP/0/RSP0/CPU0:router(config-if-cfm) # mep domain Dm1 service Sv1 mep-id 1

Creates a MEP on an interface and enters interface CFM MEP configuration mode.

Step 4 sla operation profile *profile-name* **target** {**mep-id** *id* | **mac-address** *mac-address*}

Example:

RP/0/RSP0/CPU0:router(config-if-cfm-mep)# sla operation profile Profile_1 target mac-address
01:23:45:67:89:ab

Creates an operation instance from a MEP to a specified destination.

Step 5 end or commit

Example:

RP/0/RSP0/CPU0:router(config-sla-prof-stat-cfg) # commit

Saves configuration changes.

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

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

- Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
- Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
- Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
- Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring an On-Demand SLA Operation

The Cisco ASR 9000 Series Router supports configuration of on-demand SLA operations to run on an as-needed basis for a finite period of time.

This section includes the following topics:

Configuration Guidelines

When you configure on-demand SLA operations, consider the following guidelines:

- Each MEP supports up to 50 on-demand operations.
- Each card supports up to 250 on-demand operations.
- On-demand Ethernet SLA operations can be run in addition to any other ongoing scheduled SLA operations that you might have configured, and use similar amounts of CPU and router memory. When configuring an on-demand Ethernet SLA operation, you should consider your existing SLA operation configuration and the potential impact of additional packet processing to your normal operations.
- If you do not specify a schedule for the on-demand operation, the probe defaults to running one time beginning two seconds from the execution of the command, and runs for a ten-second duration.
- If you do not specify the statistics for the probe to measure, it defaults to measuring all statistics, inlcuding these statistics by probe type:
 - CFM loopback—Two-way delay and jitter is measured by default.
 - CFM delay measurement—One-way delay and jitter in both directions, in addition to two-way delay
 and jitter is measured by default.
 - CFM synthetic loss measurement—One-way FLR in both directions is measured by default.
- The default operation mode is synchronous, where progress of the operation is reported to the console and the output of the statistics collection is displayed.

L

Note When the **once** keyword is used for 'send burst' ('send burst once' rather than 'send burst every'), it stops the collection of statistics with the packets that cross probe boundaries.

Configuring an On-Demand Ethernet SLA Operation for CFM Delay Measurement

To configure an on-demand Ethernet SLA operation for CFM delay measurement, use the following command in privileged EXEC configuration mode:

ethernet sla on-demand operation type cfm-delay-measurement probe	Configures an on-demand Ethernet S
<pre>[priority number][send {packet {once every number {miniseconds seconds minutes / hours}} burst {once every number {seconds minutes hours}} packet count number interval number {milliseconds seconds}] domain domain-name source interface type interface-path-id target {mac-address H.H.H.H mep-id id-number} [statistics measure {one-way-delay-ds </pre>	The example shows a minimum confispecifies the local domain and source target MEP, using the following defau
one-way-delay-sd one-way-jitter-ds one-way-jitter-sd round-trip-delay round-trip-jitter}][aggregate {none bins number width milliseconds}] [buckets {archive number probes size number probes}] [schedule {now at hh:mm[.ss] [day [month [year]]] in number {seconds minutes hours}][for duration {seconds minutes hours}][repeat every number (seconds minutes hours}]]	 Send a burst once for a packet continuerval of 1 second (10-second) Use default class of service (CoS) interface.
<pre>{seconds minutes hours} count probes]] [asynchronous] RP/0/RSP0/CPU0:router# ethernet sla on-demand operation type cfm-delay-measurement probe domain D1 source interface TenGigE 0/6/1/0 target mep-id 100</pre>	 Measure all statistics, including and round-trip delay and jitter st Aggregate statistics into one bin Schedule now.
	• Display results on the console.

Configuring an On-Demand Ethernet SLA Operation for CFM Loopback

To configure an on-demand Ethernet SLA operation for CFM loopback, use the following command in privileged EXEC configuration mode:

I

	1
ethernet sla on-demand operation type cfm-loopback probe [packet size bytes [test pattern {hex 0xHHHHHHHH pseudo-random}]] [priority number] [send {packet {once every number {milliseconds seconds minutes hours}} packet count number] burst {once every number {seconds minutes hours}} packet count number interval number {milliseconds seconds}] domain domain-name source interface type interface-path-id target {mac-address H.H.H.H mep-id id-number} [statistics measure {round-trip-delay round-trip-jitter}][aggregate {none bins number width milliseconds}][buckets {archive number probes size number probes}] [schedule {now at hh:mm[.ss] [day [month [year]]] in number {seconds minutes hours}}[for duration {seconds minutes hours}][repeat every number {seconds minutes hours} count probes]] [asynchronous] RP/0/RSP0/CPU0:router# ethernet sla on-demand operation type cfm-loopback probe packet size 1500 domain D1 source interface TenGigE 0/6/1/0 target mep-id 100	 Configures an on-demand Ethernet Sloperation for CFM loopback. The example shows a minimum confibut specifies the option of a minimum size, and specifies the local domain an interface and target MEP, using the for defaults: Send a burst once for a packet coand interval of 1 second (10-second) Use default test pattern of 0's for Use default class of service (Cosenders) Measure all statistics. Aggregate statistics into one bin. Schedule now. Display results on the console.

Configuring an On-Demand Ethernet SLA Operation for CFM Synthetic Loss Measurement

To configure an on-demand Ethernet SLA operation for CFM synthetic loss measurement, use this command in privileged EXEC configuration mode:

ethernet sla on-demand operation type cfm-synthetic-loss-measurement	Configures an on-demand Ethernet SLA	
probe [priority number] [send {packet {once every number {milliseconds	for CFM synthetic loss measurement.	
<pre> seconds minutes / hours}} burst {once every number {seconds minutes hours}} packet count number interval number {milliseconds seconds}] domain domain-name source interface type interface-path-id target {mac-address H.H.H. mep-id id-number} [synthetic loss calculation packets number] [statistics measure {one-way-loss-ds one-way-loss-sd}][aggregate {none bins number width milliseconds}] [buckets {archive number probes size number probes}] [schedule {now at hh:mm[.ss] [day [month [year]]] in number {seconds minutes hours}}[for duration {seconds minutes hours}][repeat every number</pre>	The example shows a minimum configure specifies the local domain and source interact target MEP.	
{seconds minutes hours} count probes]] [asynchronous]		
RP/0/RSP0/CPU0:router# ethernet sla on-demand operation type cfm-synthetic-loss-measurement probe domain D1 source interface TenGigE 0/6/1/0 target mac-address 2.3.4		

Verifying SLA Configuration

To verify SLA configuration, use one or more of these commands:

show ethernet sla configuration-errors [domain	Displays information about errors that are preventing
domain-name] [interface interface-path-id] [profile	configured SLA operations from becoming active, as
profile-name]	well as any warnings that have occurred.
show ethernet sla operations [detail] [domain domain-name] [interface interface-path-id] [profile profile-name]	Displays information about configured SLA operations.

Bit Error Rate

In network transmission, data streaming over communication channels is susceptible to unplanned alterations during transmission. Such alterations are due to noise, interference, or synchronization errors. The number of bits thus received with alterations is measured as the number of bit errors.

Bit Error Rate (BER) is the number of bit errors per unit time or time window. For example, consider a scenario where the bit rate reaching the receiver is 10 bits per second, and the bit error is 1 bit per second. In this example, the BER is bit errors/unit time or time window = 1 bit/second.

Using this feature, you can test cables and diagnose signal problems in the field. You can display and analyze the total number of error bits transmitted and the total received on the link. Your router supports BER on 10/40/100 GE interfaces.

The error range measurement that your router supports is 10E-8 through 10E-12 bits, where $E = *10^{-12}$. Thus, the error range is from:

 $10*10^{-8} = 10 \ge 0.00000001 = 0.0000001$ bits

through

 $10*10^{-12} = 10 \times 0.00000000001 = 0.0000000001$ bits

Bit errors usually occur because of:

- · Faulty or bad cables
- · Loose cable connections at one or both ends

How is Bit Error Rate Measured?

BER algorithm polls the hardware counters periodically for bit errors, every 500ms.

For 40 GE and 100GE interfaces, your router uses a physical coding sublayer (PCS) bit interleaved parity (BIP) error counter.

For 10 GE interfaces, your router employs a sync header error counter. (BIP counters aren't supported for 10GE interfaces.)

What are Bit Error Rate Error States and Thresholds?

BER has the following error conditions for which you must configure threshold values at the interface:

- Signal Degradation (SD): there's a reduction in the signal quality but no loss of service, referred to as 'graceful error'.
- Signal Failure (SF): there's a loss of service because of a link-state change, referred to as 'catastrophic error'. The SF threshold state is enabled by default.

A switch uses the BER threshold value to detect an increased error rate before performance degradation seriously affects traffic. If the polling indicates the reaching of the error threshold value:

- For SD BER: the console generates an IOS message.
- For SF BER: the console generates an IOS message. Plus, you can bring down the interface transmission at the device under test (DUT) end.

Sliding Window for Polling

BER employs the concept of a sliding window to measure bit performance while polling happens in a small-length sequence of several windows. Here, 'window' refers to the BIP period or duration defined for different threshold levels. Consider a scenario where the BIP period is 2.5 seconds and the software polls the hardware counter every 500 ms. In this example, the 2.5 seconds BIP period is complete after five polls, and the window completely deploys. For the next round of polling, the window slides to the following sequence, thus ensuring better error performance while consuming lesser memory.

Alarm Raise

If errors above the configured threshold accumulate in the first poll, an alarm is raised right away instead of waiting for the completion of the BIP period. For example, if there are errors above the threshold value in the first poll of 500 ms, an alarm is raised immediately and not after completing 2.5 seconds (five polls) of the BIP period.

Alarm Clearance

The SD and SF alarm clearance is automatic once the error value is below a certain threshold level. Your router uses the configured error threshold value to measure the errors and generates IOS messages at that threshold.

Your router waits till the last poll of window deployment before clearing the alarm. The alarm is cleared as soon as the error value goes below the configured threshold value. This ensures that no new errors accumulate during the last poll of the completed window, which might keep the error count above the threshold.

Configure BER

To configure BER thresholds:

- 1. Enter the configuration mode for your interface.
- 2. Enable the Signal Degrade Bit Error Rate (SD-BER) on the interface.



Note SD-BER is disabled by default.

- 3. Configure the SD-BER threshold.
- 4. Configure the Signal Fail Bit Error Rate (SF-BER) threshold.



Note SF-BER is enabled by default.

L

5. Enable remote fault signaling when SF BER is triggered.



Note Remote signaling for SF BER is disabled by default.

```
Router#config
Router(config)#int TenGigE 0/1/0/3
/*Enable SD-BER*/
Router(config-if)#report sd-ber
/*Configure SD-BER threshold*/
Router(config-if)#threshold sd-ber 12
/*Configure SF-BER threshold*/
Router(config-if)#threshold sf-ber 8
Router(config-if)#commit
Router(config-if)#exit
```

Running Configuration

```
int TenGigE 0/1/0/3
!
report sd-ber
!
threshold sd-ber 12
!
threshold sf-ber 8
!
```

Verification

Run the **show controllers** <**interface**> **all** command to verify the BER default value as well as the configured threshold values.

```
BER monitoring:
Signal Degrade: 1e-11 (report-alarm)
Signal Fail: 1e-9 (report-alarm, signal-rf)
Current SD BER: 0
Current SF BER: 0
BER-SD Threshold: 1e-12
BER-SD Report: Enabled
BER-SF Threshold: 1e-8
BER-SF Report: Not configured (Enabled)
```

Cyclic Redundancy Check

Table 12: Feature History Table

Feature Name	Release	Description
Cyclic Redundancy Check based Bit Error Ratio on A9K-20HG-FLEX, A9K-8HG-FLEX, and A9K-4HG-FLEX Line cards	Release 7.10.1	 Cyclic Redundancy Check (CRC) based Bit Error Ratio (BER) measures the error rate in a received message by dividing the number of errors by the total number of bits transmitted. This metric allows you to evaluate the performance of the communication system and ensure the integrity of transmitted data. A high BER indicates a higher rate of errors and potential issues with the communication system, whereas a low BER indicates a high-quality system with minimal errors. This release extends CRC-based BER support for the following line cards: A9K-20HG-FLEX A9K-8HG-FLEX A9K-4HG-FLEX

The Cyclic Redundancy Check (CRC) based Bit Error Ratio (BER) is an active measure of the error rate in a communication system that utilizes CRC as an error detection method. CRC is a widely adopted error-detection technique that ensures the integrity of data transmissions. It involves appending a fixed number of check bits to the transmitted data, which are then utilized to identify any errors that may occur during the transmission process.

The CRC based BER calculates the number of errors in a received message, divided by the total number of transmitted bits. This measurement allows you to evaluate the quality of the communication system and identify any potential issues that require attention. A low BER indicates a high-quality system with minimal errors, while a high BER signifies a higher error rate and potential concerns with the communication system.

BER represents the number of bit errors per unit of time. The BER ratio denotes the number of Cyclic Redundancy Check (CRC) errors divided by the total number of transferred bits during a specific time interval. Furthermore, BER utilizes CRC for error detection within a network, enabling you to promptly identify and address faulty links.

The CRC based BER feature is available on the following line cards: [List line cards here].

- A9K-24X10GE-1G-SE
- A9K-24X10GE-1G-TR
- A9K-48X10GE-1G-SE
- A9K-48X10GE-1G-TR

- A99-48X10GE-1G-SE
- A99-48X10GE-1G-TR
- A9K-20HG-FLEX
- A9K-8HG-FLEX
- A9K-4HG-FLEX

Configure CRC BER

```
Router#config
Router(config)#int TenGigE 0/1/0/3
/*Enable CRC BER
Router(config-if)#report crc-ber
/*Enable SD-BER*/
Router(config-if)#report sd-ber
/*Configure SD-BER threshold*/
Router(config-if)#threshold sd-ber 12
/*Configure SF-BER threshold*/
Router(config-if)#threshold sf-ber 8
/*Enable crc-ber autorecovery*/
Router(config-if)#crc-ber auto-recover 2
Router(config-if)#commit
Router(config-if)#exit
```

Running Configuration

```
interface TenGigE0/1/0/3
ipv4 address 11.1.13.1 255.255.255.0
report crc-ber ---- > mandatory config to report crc-ber
report sd-ber ----- > To report sd-ber
threshold sd-ber 12 ---- > sd-ber threshold set to 12
threshold sf-ber 8 ---- > sf-ber threshold set to 8
crc-ber auto-recover 2 ---- > ber is cleared within configured time
```

Verification

Run the **show controllers <interface> all** command to verify the BER default value as well as the configured threshold values.

```
RP/0/RSP0/CPU0:ios#show controllers tenGigE0/1/0/3 all | inc BER
BER-SD Threshold: 1e-6
BER-SD Report: Enabled
BER-SF Threshold: 1e-7
BER-SF Report: Not configured (Enabled)
BER-CRC Report: Enabled
```

Associated Commands

- report crc-ber
- crc-ber auto-recover
- report sd-ber
- report sf-ber disable
- threshold sd-ber

• threshold sf-ber

Configuring Ethernet LMI

To configure Ethernet LMI, complete the following tasks:

Prerequisites for Configuring E-LMI

Before you configure E-LMI on the Cisco ASR 9000 Series Router, be sure that you complete the following requirements:

- Identify the local and remote UNIs in your network where you want to run E-LMI, and define a naming convention for them.
- Enable E-LMI on the corresponding CE interface link on a device that supports E-LMI CE operation, such as the Cisco Catalyst 3750 Metro Series Switches.

Restrictions for Configuring E-LMI

When configuring E-LMI, consider the following restrictions:

- E-LMI is not supported on subinterfaces or bundle interfaces. E-LMI is configurable on Ethernet physical interfaces only.
- E-LMI is not supported on nV satellite access interfaces when the inter-chassis links are configured as a bundle.

Creating EVCs for E-LMI

EVCs for E-LMI on the Cisco ASR 9000 Series Router are established by first configuring EFPs (Layer 2 subinterfaces) on the local UNI physical Ethernet interface link to the CE where E-LMI will be running, and also on the remote UNI link. Then, the EFPs need to be assigned to an L2VPN bridge domain to create the EVC.

To create EVCs, complete the following tasks:

Configuring EFPs

This section describes the basic configuration of an EFP. For more information about configuration of other supported Layer 2 services, see the *Cisco ASR 9000 Series Aggregation Services Routers L2VPN and Ethernet Services Configuration Guide*.

To configure an EFP, complete these tasks:

SUMMARY STEPS

- 1. configure
- 2. interface [GigabitEthernet | TenGigE] interface-path-id.subinterface l2transport
- **3.** encapsulation dot1q vlan-id [, untagged |, vlan-id | –vlan-id] [exact | ingress source-mac mac-address | second-dot1q vlan-id]
- 4. end or commit

DETAILED STEPS

Procedure

I

Step 1	configure
	Example:
	RP/0/RSP0/CPU0:router# configure
	Enters global configuration mode.
Step 2	interface [GigabitEthernet TenGigE] interface-path-id.subinterface l2transport
	Example:
	RP/0/RSP0/CPU0:router(config)# interface gigabitethernet 0/0/0/0.0 l2transport
	Creates a VLAN subinterface in Layer 2 transport mode and enters Layer 2 subinterface configuration mode.
Step 3	encapsulation dot1q vlan-id [, untagged , vlan-id –vlan-id] [exact ingress source-mac mac-address second-dot1q vlan-id] vlan-id]
	Example:
	RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 1-20
	Defines the matching criteria to map 802.1Q frames ingress on an interface to the appropriate service instance.
Step 4	end or commit
	Example:
	RP/0/RSP0/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.

Configuring a Bridge Group and Assigning EFPs to a Bridge Domain

To configure a bridge group and assign EFPs to a bridge domain to create an EVC, complete the following steps:

SUMMARY STEPS

- 1. configure
- 2. l2vpn
- 3. bridge group bridge-group-name
- 4. bridge-domain bridge-domain-name
- 5. interface [GigabitEthernet | TenGigE] interface-path-id.subinterface
- 6. end or commit

DETAILED STEPS

Procedure

Step 1	configure
	Example:
	RP/0/RSP0/CPU0:router# configure
	Enters global configuration mode.
Step 2	l2vpn
	Example:
	RP/0/RSP0/CPU0:router(config)# 12vpn
	Enters L2VPN configuration mode.
Step 3	bridge group bridge-group-name
	Example:
	RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group BG1
	Creates a bridge group and enters L2VPN bridge group configuration mode.
Step 4	bridge-domain bridge-domain-name
	Example:
	RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain BD1
	Creates a bridge domain and enters L2VPN bridge group bridge domain configuration mode.
Step 5	interface [GigabitEthernet TenGigE] interface-path-id.subinterface
	Example:

RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# interface GigabitEthernet 0/0/0/0.0

Associates the EFP (EVC) with the specified bridge domain and enters L2VPN bridge group bridge domain attachment circuit configuration mode, where *interface-path-id* is specified as the *rack/slot/module/port* location of the interface and *.subinterface* is the subinterface number.

Repeat this step for as many EFPs (EVCs) as you want to associate with the bridge domain.

Step 6 end or commit

Example:

RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)# end

or

```
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac) # commit
```

Saves configuration changes.

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

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

- Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
- Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
- Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
- Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring Ethernet CFM for E-LMI

The Cisco ASR 9000 Series Router uses Ethernet CFM to monitor EVC status for E-LMI. To use CFM for E-LMI, a CFM maintenance domain and service must be configured on the router and the EFPs must be configured as CFM Up MEPs.

To configure Ethernet CFM for E-LMI, complete the following tasks:

Configuring Ethernet CFM

The minimum configuration to support E-LMI using Ethernet CFM is to configure a CFM maintenance domain and service on the router. Other CFM options can also be configured.

For more information about the tasks to configure Ethernet CFM, see the "Configuring Ethernet CFM" section on page 108.

Configuring EFPs as CFM Up MEPs

This section describes the minimum tasks required to configure EFPs as CFM MEPs. For more information about configuring CFM MEPs, see the "Configuring CFM MEPs" section on page 118.

To configure EFPs as CFM MEPs, complete the following tasks for each E-LMI EFP:

SUMMARY STEPS

- 1. configure
- 2. interface gigabitethernet interface-path-id.subinterface l2transport
- 3. ethernet cfm
- 4. mep domain domain-name service service-name mep-id id-number
- 5. end or commit

DETAILED STEPS

Procedure

Step 1	configure
	Example:
	RP/0/RSP0/CPU0:router# configure
	Enters global configuration mode.
Step 2	interface gigabitethernet interface-path-id.subinterface l2transport
	Example:
	RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet0/0/0/0.0 l2transport
	Enters Layer 2 subinterface configuration mode for the EFP.
Step 3	ethernet cfm
	Example:
	RP/0/RSP0/CPU0:router(config-subif) # ethernet cfm
	Enters Ethernet CFM interface configuration mode.
Step 4	mep domain domain-name service service-name mep-id id-number
	Example:
	RP/0/RSP0/CPU0:router(config-if-cfm)# mep domain GLOBAL service CustomerA mep-id 22
	Creates a MEP on an interface and enters interface CFM MEP configuration mode.
Step 5	end or commit
	Example:
	RP/0/RSP0/CPU0:router(config-if-cfm-mep)# commit
	Saves configuration changes.
	• When you issue the end command, the system prompts you to commit changes:

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

- Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
- Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
- Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
- Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring UNI Names on the Physical Interface

It is recommended that you configure UNI names on the physical interface links to both the local and remote UNIs to aid in management for the E-LMI protocol. To configure UNI names, complete the following tasks on the physical interface links to both the local and remote UNIs:

SUMMARY STEPS

- 1. configure
- 2. interface [GigabitEthernet | TenGigE] interface-path-id
- 3. ethernet uni id name
- 4. end or commit

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:
	<pre>RP/0/RSP0/CPU0:router(config)# interface gigabitethernet 0/0/0/0</pre>
	Enters interface configuration mode for the physical interface.
Step 3	ethernet uni id name
	Example:

RP/0/RSP0/CPU0:router(config-if) # ethernet uni id PE1-CustA-Slot0-Port0

Specifies a name (up to 64 characters) for the Ethernet UNI interface link.

Step 4 end or commit

Example:

RP/0/RSP0/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.

Enabling E-LMI on the Physical Interface

The Cisco ASR 9000 Series Router supports the E-LMI protocol only on physical Ethernet interfaces. To enable E-LMI, complete the following tasks on the physical Ethernet interface link to the local UNI:

SUMMARY STEPS

- 1. configure
- 2. interface [GigabitEthernet | TenGigE] interface-path-id
- 3. ethernet lmi
- 4. end or commit

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# interface gigabitethernet 0/0/0/0

Enters interface configuration mode for the physical interface.

Step 3 ethernet lmi

Example:

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

Enables Ethernet Local Managment Interface operation on an interface and enters interface Ethernet LMI configuration mode.

Step 4 end or commit

Example:

RP/0/RSP0/CPU0:router(config-if-lmi)# commit

Saves configuration changes.

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

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

- Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
- Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
- Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
- Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring the Polling Verification Timer

The MEF T392 Polling Verification Timer (PVT) specifies the allowable time between transmission of a STATUS message and receipt of a STATUS ENQUIRY from the UNI-C before recording an error. The default value is 15 seconds.

To modify the default value or disable the PVT altogether, complete the following tasks:

SUMMARY STEPS

- 1. configure
- 2. interface [GigabitEthernet | TenGigE] interface-path-id

I

- 3. ethernet lmi
- 4. polling-verification-timer {*interval* | disable}
- 5. end or commit

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# interface gigabitethernet 0/0/0/0
	Enters interface configuration mode for the physical interface.
Step 3	ethernet lmi
	Example:
	RP/0/RSP0/CPU0:router(config-if)# ethernet lmi
	Enables Ethernet Local Managment Interface operation on an interface and enters interface Ethernet LMI configuration mode.
Step 4	polling-verification-timer {interval disable}
	Example:
	RP/0/RSP0/CPU0:router(config-if-lmi)# polling-verification-timer 30
	Sets or disables the MEF T392 Polling Verification Timer for E-LMI operation, which specifies the allowable time (in seconds) between transmission of a STATUS message and receipt of a STATUS ENQUIRY from the UNI-C before recording an error. The default is 15.
Step 5	end or commit
	Example:
	RP/0/RSP0/CPU0:router(config-if-lmi)# commit
	Saves configuration changes.
	• When you issue the end command, the system prompts you to commit changes:
	Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:

- Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
- Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
- Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
- Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring the Status Counter

The MEF N393 Status Counter value is used to determine E-LMI operational status by tracking receipt of consecutive good packets or successive expiration of the PVT on packets. The default counter is four, which means that while the E-LMI protocol is in Down state, four good packets must be received consecutively to change the protocol state to Up, or while the E-LMI protocol is in Up state, four consecutive PVT expirations must occur before the state of the E-LMI protocol is changed to Down on the interface.

To modify the status counter default value, complete the following tasks:

SUMMARY STEPS

- 1. configure
- 2. interface [GigabitEthernet | TenGigE] interface-path-id
- 3. ethernet lmi
- 4. status-counter threshold
- 5. end or commit

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# interface gigabitethernet 0/0/0/0
	Enters interface configuration mode for the physical interface.
Step 3	ethernet lmi

Example:

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

Enables Ethernet Local Managment Interface operation on an interface and enters interface Ethernet LMI configuration mode.

Step 4 status-counter threshold

Example:

RP/0/RSP0/CPU0:router(config-if-lmi)# status-counter 5

Sets the MEF N393 Status Counter value that is used to determine E-LMI operational status by tracking receipt of consecutive good and bad packets from a peer. The default is 4.

Step 5 end or commit

Example:

RP/0/RSP0/CPU0:router(config-if-lmi)# 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 Syslog Messages for E-LMI Errors or Events

The E-LMI protocol tracks certain errors and events whose counts can be displayed using the **show ethernet Imi interfaces** command.

To disable syslog messages for E-LMI errors or events, complete the following tasks:

SUMMARY STEPS

- 1. configure
- 2. interface [GigabitEthernet | TenGigE] interface-path-id
- 3. ethernet lmi
- 4. log {errors | events} disable

5. end or commit

DETAILED STEPS

Procedure

I

configure			
Example:			
RP/0/RSP0/CPU0:router# configure			
Enters global configuration mode.			
interface [GigabitEthernet TenGigE] interface-path-id			
Example:			
RP/0/RSP0/CPU0:router# interface gigabitethernet 0/0/0/0			
Enters interface configuration mode for the physical interface.			
ethernet lmi			
Example:			
RP/0/RSP0/CPU0:router(config-if)# ethernet lmi			
Enables Ethernet Local Managment Interface operation on an interface and enters interface Ethernet LMI configuration mode.			
log {errors events} disable			
Example:			
RP/0/RSP0/CPU0:router(config-if-lmi)# log events disable			
Turns off syslog messages for E-LMI errors or events.			
end or commit			
Example:			
RP/0/RSP0/CPU0:router(config-if-lmi)# 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 **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 Use of the Cisco-Proprietary Remote UNI Details Information Element

To provide additional information within the E-LMI protocol, the Cisco IOS XR software implements a Cisco-proprietary information element called Remote UNI Details to send information to the CE about remote UNI names and states. This information element implements what is currently an unused identifier from the E-LMI MEF 16 specification.

To disable use of the Remote UNI Details information element, complete the following tasks:

SUMMARY STEPS

- 1. configure
- 2. interface [GigabitEthernet | TenGigE] interface-path-id
- **3**. ethernet lmi
- 4. extension remote-uni disable
- 5. end or commit

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# interface gigabitethernet 0/0/0/0

Enters interface configuration mode for the physical interface.

Step 3 ethernet lmi

Example:

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

Enables Ethernet Local Managment Interface operation on an interface and enters interface Ethernet LMI configuration mode.

Step 4 extension remote-uni disable

Example:

RP/0/RSP0/CPU0:router(config-if-lmi)# extension remote-uni disable

Disables transmission of the Cisco-proprietary Remote UNI Details information element in E-LMI STATUS messages.

Step 5 end or commit

Example:

RP/0/RSP0/CPU0:router(config-if-lmi)# 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 Ethernet LMI Configuration

Use the **show ethernet lmi interfaces detail** command to display the values for the Ethernet LMI configuration for a particular interface, or for all interfaces. The following example shows sample output for the command:

```
RP/0/RSP0/CPU0:router# show ethernet lmi interfaces detail
Interface: GigabitEthernet0/0/0/0
  Ether LMI Link Status: Up
 UNI Id: PE1-CustA-Slot0-Port0
  Line Protocol State: Up
  MTU: 1514 (1 PDU reqd. for full report)
  CE-VLAN/EVC Map Type: Bundling (1 EVC)
  Configuration: Status counter 4, Polling Verification Timer 15 seconds
  Last Data Instance Sent: 0
  Last Sequence Numbers: Sent 0, Received 0
  Reliability Errors:
    Status Enq Timeouts
                                         0 Invalid Sequence Number
                                                                                0
    Invalid Report Type
                                         0
  Protocol Errors:
   Malformed PDUs
                                         0 Invalid Procotol Version
                                                                                0
    Invalid Message Type
                                         0 Out of Sequence IE
                                                                                0
    Duplicated IE
                                         0 Mandatory IE Missing
                                                                                0
```

0

Invalid Mandatory IE Unrecognized IE		0 Invalid non-Mandatory 0 Unexpected IE	/ IE			
Full Status Enq Received PDU Received LMI Link Status Changed Counters cleared	never never 00:00:03 ago never	Full Status Sent PDU Sent Last Protocol Error	never never never			
<pre>Sub-interface: GigabitEthernet0/0/0/0.0 VLANs: 1-20 EVC Status: Active EVC Type: Point-to-Point OAM Protocol: CFM CFM Domain: Global (level 5) CFM Service: CustomerA Remote UNI Count: Configured = 1, Active = 1</pre>						
Remote UNI Id			Status			
PE1-CustA-Slot0-Port1			 Up			

Troubleshooting Tips for E-LMI Configuration

This section describes some basic information for troubleshooting your E-LMI configuration in the following topics:

Ethernet LMI Link Status Troubleshooting

The E-LMI protocol operational status is reported in the "Ether LMI Link Status" or "ELMI state" fields in the output of forms of the **show ethernet lmi interfaces** command. To investigate a link status other than "Up," consider the following guidelines:

- Unknown (PVT disabled)—Indicates that the Polling Verification Timer has been configured as disabled, so no status information can be provided. To see an "Up" or "Down" status, you must enable the PVT. For more information, see the "Configuring the Polling Verification Timer" section on page 146.
- Down-The E-LMI link status can be Down for the following reasons:
 - The PVT has timed out the number of times specified by the **status-counter** command. This indicates that STATUS ENQUIRY messages have not been received from the CE device. This can be for the following reasons:

— The CE device is not connected to the PE device. Check that the CE device is connected to the interface on which E-LMI is enabled on the PE device.

— The CE device is not sending Status Enquiries. Check that E-LMI is enabled on the CE interface which is connected to the PE device.

— Protocol errors are causing the PVT to expire. The PVT is only reset when a valid (unerrored) STATUS ENQUIRY message is received.

- The Line Protocol State is "Down" or "Admin Down."
- The protocol has not yet started on the interface because it does not have useful information to provide, such as the UNI Id or details about EVCs. This is a symptom of provisioning misconfiguration.



Note If the protocol is started, then E-LMI still responds to STATUS ENQUIRY messages when it is in "Down" state.

Ethernet LMI Line Protocol State Troubleshooting

The E-LMI line protocol state is reported in the "Line Protocol State" or "LineP State" fields in the output of forms of the **show ethernet lmi interfaces** command. The line protocol state is the state of the E-LMI protocol on the physical interface.

To investigate a line protocol state other than Up, consider the following guidelines:

- Admin-Down—The interface is configured with the shutdown command. Use the no shutdown command to bring the interface up.
- Down—Indicates a fault on the interface. Run the **show interfaces** command to display both the interface state and the interface line protocol state for more information, and take the following actions to investigate further:
 - If both states are Down, this suggests a physical problem with the link (for example, the cable is not plugged into either the PE or CE device).
 - If the interface state is Up but the line protocol state is Down, this suggests that an OAM protocol has brought the line protocol state down due to a fault. Use the **show efd interface** command for more information.

Ethernet LMI Error Counter Troubleshooting

The show ethernet lmi interfaces command displays two sections of error counters:

- Reliability Errors—Can indicate that messages are being lost between the PE and CE devices. The timers in the last block of the output should indicate that messages are being sent and received by the PE device.
- Protocol Errors—Indicates that the CE device is sending packets to the PE device, but the PE does not
 understand those packets. This suggests an incorrect configuration of the E-LMI protocol on the CE side,
 or corruption of the packets on the path between the CE and PE. E-LMI packets have a strictly defined
 structure in the MEF 16 standard, and any deviation from that results in a protocol error. The PE will
 not respond to any packets that are malformed and result in a protocol error.

Immediately after configuring E-LMI, all of the error counters should be zero, with the possible exception of the Status Enq Timeouts counter. The Status Enq Timeouts counter can be non-zero if the E-LMI protocol was started on the PE interface before being started on the corresponding CE interface. However, once the protocol is started on both devices, this counter should stop increasing.

If the Status Enq Timeouts counter is non-zero and is increasing, this indicates that enquiries are not being received from the CE device. This can be due to the following conditions:

- The CE device is not connected or not sending STATUS ENQUIRY messages. For more information, see also the "Ethernet LMI Link Status Troubleshooting" section on page 153.
- The Polling Timer on the CE device is configured to a value greater than the PVT on the PE device. Verify that the value of the **polling-verification-timer** command on the PE device is larger than the value of the CE's Polling Timer.

For more information, see also the documentation for the show ethernet lmi interfaces command in the *Cisco* ASR 9000 Aggregation Services Router Interfaces and Hardware Component Command Reference.

Ethernet LMI Remote UNI Troubleshooting

Information about the Remote UNIs is reported in the output of the **show ethernet lmi interfaces detail** command. The Remote UNI ID field displays the name of the UNI as configured by the **ethernet uni id** command, or it displays the CFM MEP ID of the UNI when the UNI name has not been configured.

If the Remote UNI is missing from the table altogether, this is can be due to the following conditions:

- The remote UNI's EFP is missing from the bridge-domain in L2VPN configuration. Use the **show ethernet cfm configuration-errors** command to verify the configuration.
- A CFM MEP has not been configured on the remote UNI's EFP.

Configuring UDLD

UDLD is configured for each interface. The interface must be a physical ethernet interface.

Perform these steps to configure UDLD protocol on an interface.

SUMMARY STEPS

- 1. configure
- 2. interface [GigabitEthernet | TenGigE] interface-path-id
- 3. ethernet udld
- 4. mode {normal |aggressive}
- 5. message-time [7-90]
- 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 function 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 [7-90]			
	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 between 7 to 90 seconds.			
Step 6	logging disable			
	Example:			
	RP/0/RSP0/CPU0:router(config-if-udld)# loggig disable			
	(Optional) This command 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.			

Configuration Examples for Ethernet OAM

This section provides the following configuration examples:

Configuration Examples for EOAM Interfaces

This section provides the following configuration examples:

Configuring an Ethernet OAM Profile Globally: Example

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

```
configure terminal
ethernet oam profile Profile 1
 link-monitor
   symbol-period window 60000
  symbol-period threshold low 10000000 high 6000000
   frame window 60
  frame threshold low 10000000 high 60000000
   frame-period window 60000
   frame-period threshold low 100 high 12000000
  frame-seconds window 900000
  frame-seconds threshold 3 threshold 900
  exit
 mib-retrieval
  connection timeout 30
  require-remote mode active
 require-remote link-monitoring
  require-remote mib-retrieval
 action dying-gasp error-disable-interface
 action critical-event error-disable-interface
 action discovery-timeout error-disable-interface
 action session-down error-disable-interface
 action capabilities-conflict error-disable-interface
 action wiring-conflict error-disable-interface
 action remote-loopback error-disable-interface
  commit
```

Configuring Ethernet OAM Features on an Individual Interface: Example

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

```
configure terminal
interface TenGigE 0/1/0/0
 ethernet oam
   link-monitor
   symbol-period window 60000
   symbol-period threshold low 10000000 high 6000000
   frame window 60
   frame threshold low 10000000 high 60000000
    frame-period window 60000
    frame-period threshold low 100 high 12000000
   frame-seconds window 900000
   frame-seconds threshold 3 threshold 900
   exit
  mib-retrieval
   connection timeout 30
  require-remote mode active
  require-remote link-monitoring
   require-remote mib-retrieval
  action link-fault error-disable-interface
  action dying-gasp error-disable-interface
   action critical-event error-disable-interface
  action discovery-timeout error-disable-interface
  action session-down error-disable-interface
  action capabilities-conflict error-disable-interface
  action wiring-conflict error-disable-interface
   action remote-loopback error-disable-interface
   commit.
```

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

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

```
configure terminal
ethernet oam profile Profile_1
 mode passive
 action dying-gasp disable
 action critical-event disable
 action discovery-timeout disable
 action session-up disable
 action session-down disable
 action capabilities-conflict disable
 action wiring-conflict disable
 action remote-loopback disable
 action uni-directional link-fault error-disable-interface
 commit
configure terminal
 interface TenGigE 0/1/0/0
 ethernet oam
  profile Profile 1
   mode active
   action dying-gasp log
   action critical-event log
   action discovery-timeout log
   action session-up log
   action session-down log
   action capabilities-conflict log
   action wiring-conflict log
    action remote-loopback log
   action uni-directional link-fault log
   uni-directional link-fault detection
    commit
```

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
```

Configuring a Remote Loopback on an Ethernet OAM Peer: Example

This example shows how to configure a remote loopback on an Ethernet OAM peer:

```
configure terminal
interface gigabitethernet 0/1/5/6
ethernet oam
profile Profile_1
remote-loopback
```

This example shows how to start a remote loopback on a configured Ethernet OAM interface:

ethernet oam loopback enable TenGigE 0/6/1/0

Clearing Ethernet OAM Statistics on an Interface: Example

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

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

Enabling SNMP Server Traps on a Router: Example

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

```
configure terminal
  ethernet oam profile Profile_1
  snmp-server traps ethernet oam events
```

Configuration Examples for Ethernet CFM

This section includes the following examples:

Ethernet CFM Domain Configuration: Example

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

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

Ethernet CFM Service Configuration: Example

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

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

Flexible Tagging for an Ethernet CFM Service Configuration: Example

This example shows how to set the number of tags in CFM packets from up MEPs in a CFM domain service:
```
configure
  ethernet cfm
  domain D1 level 1
   service S2 bridge group BG1 bridge-domain BD2
   tags 1
   commit
```

Continuity Check for an Ethernet CFM Service Configuration: Example

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

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

MIP Creation for an Ethernet CFM Service Configuration: Example

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

```
RP/0/RSP0/CPU0:router(config-cfm-dmn-svc)# mip auto-create all ccm-learning
RP/0/RSP0/CPU0:router(config-cfm-dmn-svc)# commit
```

Cross-check for an Ethernet CFM Service Configuration: Example

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

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

Other Ethernet CFM Service Parameter Configuration: Example

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

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

MEP Configuration: Example

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

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

Ethernet CFM Show Command: Examples

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

Example 1

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

RP/0/RSP0/CPU0:router# show ethernet cfm local maintenance-points

Domain/Level	Service	Interface	Туре	ID	MAC
fig/5	bay	Gi0/10/0/12.23456	Dn MEP	2	44:55:66
fig/5	bay	Gi0/0/1/0.1	MIP		55:66:77
fred/3	barney	Gi0/1/0/0.1	Up MEP	5	66:77:88!

Example 2

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

RP/0/RSP0/CPU0:router# show ethernet cfm configuration-errors

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

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

* An Up MEP is configured for this domain on interface GigabitEthernet0/1/2/3.234 and an

Up MEP is also configured for domain blort, which is at the same level (5).

* A MEP is configured on interface GigabitEthernet0/3/2/1.1 for this domain/service, which

has CC interval 100ms, but the lowest interval supported on that interface is 1s
```

Example 3

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

RP/0/RSP0/CPU0:router# show ethernet cfm local meps

 A - AIS received R - Remote Defect received L - Loop (our MAC received) C - Config (our ID received) X - Cross-connect (wrong MAID) P - Peer port down 	I V T M) U	- Wrong i - Wrong I - Timed c - Missing - Unexpec	lnte Geve out g (o	erval el (archive cross-che d (cross-	ed) eck) -check)
Domain foo (level 6), Service b ID Interface (State)	bar Dir	MEPs/Err	RD	Defects	AIS
100 Gi1/1/0/1.234 (Up)	Up	0/0	N	A	L7
Domain fred (level 5), Service ID Interface (State)	barn Dir	NEPs/Err	RD	Defects	AIS
2 Gi0/1/0/0.234 (Up) Domain foo (level 6), Service b	Up bar	3/2	Y	RPC	L6
ID Interface (State)	Dir	MEPs/Err	RD	Defects	AIS
100 Gil/1/0/1.234 (Up)	Up	0/0	Ν	А	
Domain fred (level 5), Service ID Interface (State)	barn Dir	ney MEPs/Err	RD	Defects	AIS
2 Gi0/1/0/0.234 (Up)	Up	3/2	Y Y	RPC	

Example 4

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

RP/0/RSP0/CPU0:router# show ethernet cfm peer meps

Flags: > - Ok R - Rem L - Loo: C - Con X - Cro Domain f Up MEP o.	ote Defect rece: p (our MAC rece: fig (our ID rece ss-connect (wro) red (level 7), s n GigabitEtherne	ived ived) eived) ng MAID) Service H et0/1/0/(I - Wrong V - Wrong T - Timed M - Missin U - Unexpe Darney 0.234, MEP-ID	interval level out g (cross-c cted (cros	check) ss-check)		
St ID	MAC address	Port	Up/Downtime	CcmRcvd	SeqErr	RDI	Error	
> 1	0011.2233.4455	α	00:00:01	1234	0	0	0	
R> 4	4455.6677.8899	qU	1d 03:04	3456	0	234	0	
L 2	1122.3344.5566	qU	3w 1d 6h	3254	0	0	3254	
C 2	7788.9900.1122	Test	00:13	2345	6	20	2345	
х 3	2233.4455.6677	Up	00:23	30	0	0	30	
I З	3344.5566.7788	Down	00:34	12345	0	300	1234	
V 3	8899.0011.2233	Blocked	00:35	45	0	0	45	
т 5	5566.7788.9900		00:56	20	0	0	0	
м 6				0	0	0	0	
U> 7	6677.8899.0011	Up	00:02	456	0	0	0	
Domain f Down MEP	red (level 7), s on GigabitEthe:	Service f rnet0/10,	fig /0/12.123, ME	P-ID 3				
St ID	MAC address	Port	Up/Downtime	CcmRcvd S	GeqErr	RDI E	Error	
> 1	9900.1122.3344	Up	03:45	4321	0	0	0	

Example 5

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

```
RP/0/RSP0/CPU0:router# show ethernet cfm peer meps detail
Domain dom3 (level 5), Service ser3
Down MEP on GigabitEthernet0/0/0/0 MEP-ID 1
_____
Peer MEP-ID 10, MAC 0001.0203.0403
  CFM state: Wrong level, for 00:01:34
  Port state: Up
  CCM defects detected: V - Wrong Level
  CCMs received: 5
    Out-of-sequence:
                             0
                            5
    Remote Defect received:
                             0
    Wrong Level:
    Cross-connect (wrong MAID): 0
    Wrong Interval:
                             5
    Loop (our MAC received):
                             0
    Config (our ID received):
                             0
Last CCM received 00:00:06 ago:
    Level: 4, Version: 0, Interval: 1min
    Sequence number: 5, MEP-ID: 10
    MAID: String: dom3, String: ser3
```

Port status: Up, Interface status: Up Domain dom4 (level 2), Service ser4 Down MEP on GigabitEthernet0/0/0/0 MEP-ID 1 _____ Peer MEP-ID 20, MAC 0001.0203.0402 CFM state: Ok, for 00:00:04 Port state: Up CCMs received: 7 Out-of-sequence: 1 Remote Defect received: 0 Wrong Level: 0 Cross-connect (wrong MAID): 0 Wrong Interval: 0 Loop (our MAC received): 0 Config (our ID received): 0 Last CCM received 00:00:04 ago: Level: 2, Version: 0, Interval: 10s Sequence number: 1, MEP-ID: 20 MAID: String: dom4, String: ser4 Chassis ID: Local: ios; Management address: 'Not specified' Port status: Up, Interface status: Up Peer MEP-ID 21, MAC 0001.0203.0403 CFM state: Ok, for 00:00:05 Port state: Up CCMs received: 6 Out-of-sequence: Ο Remote Defect received: 0 Wrong Level: 0 Cross-connect (wrong MAID): 0 Wrong Interval: 0 Loop (our MAC received): 0 Config (our ID received): 0 Last CCM received 00:00:05 ago: Level: 2, Version: 0, Interval: 10s Sequence number: 1, MEP-ID: 21 MAID: String: dom4, String: ser4 Port status: Up, Interface status: Up Domain dom5 (level 2), Service ser5 Up MEP on Standby Bundle-Ether 1 MEP-ID 1 _____ Peer MEP-ID 600, MAC 0001.0203.0401 CFM state: Ok (Standby), for 00:00:08, RDI received Port state: Down CCM defects detected: Defects below ignored on local standby MEP I - Wrong Interval R - Remote Defect received CCMs received: 5 Out-of-sequence: 0 5 Remote Defect received: Wrong Level: 0 Cross-connect W(wrong MAID): 0 Wrong Interval: 5 Loop (our MAC received): 0 Config (our ID received): 0 Last CCM received 00:00:08 ago: Level: 2, Version: 0, Interval: 10s Sequence number: 1, MEP-ID: 600 MAID: DNS-like: dom5, String: ser5 Chassis ID: Local: ios; Management address: 'Not specified'

Port status: Up, Interface status: Down Peer MEP-ID 601, MAC 0001.0203.0402 CFM state: Timed Out (Standby), for 00:15:14, RDI received Port state: Down CCM defects detected: Defects below ignored on local standby MEP I - Wrong Interval R - Remote Defect received T - Timed Out P - Peer port down CCMs received: 2 Out-of-sequence: 0 Remote Defect received: 2 Wrong Level: 0 Cross-connect (wrong MAID): 0 2 Wrong Interval: Loop (our MAC received): 0 Config (our ID received): 0 Last CCM received 00:15:49 ago: Level: 2, Version: 0, Interval: 10s Sequence number: 1, MEP-ID: 600 MAID: DNS-like: dom5, String: ser5 Chassis ID: Local: ios; Management address: 'Not specified' Port status: Up, Interface status: Down

AIS for CFM Configuration: Examples

Example 1

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

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# ethernet cfm
RP/0/RSP0/CPU0:router(config-cfm)# domain D1 level 1
RP/0/RSP0/CPU0:router(config-cfm-dmn)# service S1 bridge group BG1 bridge-domain BD2
RP/0/RSP0/CPU0:router(config-cfm-dmn-svc)# ais transmission interval 1m cos 7
```

configure

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

Example 2

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

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# ethernet cfm
RP/0/RSP0/CPU0:router(config-cfm)# domain D1 level 1
RP/0/RSP0/CPU0:router(config-cfm-dmn)# service S2 bridge group BG1 bridge-domain BD2
RP/0/RSP0/CPU0:router(config-cfm-dmn-svc)# log ais
```

configure

```
RP/0//CPU0:router(config)# ethernet cfm
RP/0//CPU0:router(config-cfm)# domain D1 level 1
RP/0//CPU0:router(config-cfm-dmn)# service Cross_Connect_1 xconnect group XG1 p2p
RP/0//CPU0:router(config-cfm-dmn-svc)# log ais
```

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

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface gigabitethernet 0/1/0/2
RP/0/RSP0/CPU0:router(config-if)# ethernet cfm
RP/0/0RSP0/CPU0:router(config-if-cfm)# ais transmission up interval 1m cos 7
```

AIS for CFM Show Commands: Examples

This section includes the following examples:

show ethernet cfm interfaces ais Command: Example

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

RP/0/RSP0/CPU0:router# show ethernet cfm interfaces ais

Defects (from at least o	ne pe	er MEP):				
A - AIS received		I - Wro	ng interv	val		
R - Remote Defect recei	ved	V - Wro	ng Level			
L - Loop (our MAC recei	ved)	T - Tim	ed out (a	rchive	d)	
C - Config (our ID rece	ived)	M - Mis	sing (cro	ss-che	ck)	
X - Cross-connect (wron	g MAI	D) U - Une	xpected ((cross-	check)	
P - Peer port down		D - Loc	al port d	lown		
		Trigger			Transmission	ı
	AIS		Via			
Interface (State)	Dir	L Defects	Levels	L Int	Last started	Packets
Gi0/1/0/0.234 (Up)	Dn	5 RPC	6	7 1s	01:32:56 ago	5576
Gi0/1/0/0.567 (Up)	Up	0 M	2,3	5 1s	00:16:23 ago	983
Gi0/1/0/1.1 (Dn)	αU	D		7 60s	01:02:44 ago	3764

show ethernet cfm local meps Command: Examples

Example 1: Default

Gi0/1/0/2 (Up)

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

1!

RP/0/RSP0/CPU0:router# show ethernet cfm local meps

Dn

0 RX

A - AIS received	I - Wrong interval
R - Remote Defect received	V - Wrong Level
L - Loop (our MAC received)	T - Timed out (archived)
C - Config (our ID received)	M - Missing (cross-check)
X - Cross-connect (wrong MAID)	U - Unexpected (cross-check)
P - Peer port down	
Domain foo (level 6), Service k	bar
ID Interface (State)	Dir MEPs/Err RD Defects AIS
100 Gi1/1/0/1.234 (Up)	Up 0/0 N A 7
Domain fred (level 5), Service	barney
ID Interface (State)	Dir MEPs/Err RD Defects AIS
2 Gi0/1/0/0.234 (Up)	Up 3/2 Y RPC 6

Example 2: Domain Service

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

RP/0/RSP0RP0/CPU0:router# show ethernet cfm local meps domain foo service bar detail

```
Domain foo (level 6), Service bar
Up MEP on GigabitEthernet0/1/0/0.234, MEP-ID 100
_____
 Interface state: Up MAC address: 1122.3344.5566
 Peer MEPs: 0 up, 0 with errors, 0 timed out (archived)
 CCM generation enabled: No
 AIS generation enabled: Yes (level: 7, interval: 1s)
 Sending AIS:
                      Yes (started 01:32:56 ago)
 Receiving AIS:
                     Yes (from lower MEP, started 01:32:56 ago)
Domain fred (level 5), Service barney
Up MEP on GigabitEthernet0/1/0/0.234, MEP-ID 2
_____
 Interface state: Up MAC address: 1122.3344.5566
 Peer MEPs: 3 up, 2 with errors, 0 timed out (archived)
 Cross-check defects: 0 missing, 0 unexpected
 CCM generation enabled: Yes (Remote Defect detected: Yes)
                      R - Remote Defect received
 CCM defects detected:
                     P - Peer port down
                      C - Config (our ID received)
 AIS generation enabled: Yes (level: 6, interval: 1s)
 Sending AIS:
                      Yes (to higher MEP, started 01:32:56 ago)
                     No
 Receiving AIS:
```

Example 3: Verbose

The following example shows how to display verbose statistics for MEPs in a domain service:

```
Note
```

The Discarded CCMs field is not displayed when the number is zero (0). It is unusual for the count of discarded CCMs to be any thing other than zero, since CCMs are only discarded when the limit on the number of peer MEPs is reached.

RP/0/RSP0RP0/CPU0:router# show ethernet cfm local meps domain foo service bar verbose

```
Domain foo (level 6), Service bar
Up MEP on GigabitEthernet0/1/0/0.234, MEP-ID 100
_____
 Interface state: Up
                   MAC address: 1122.3344.5566
 Peer MEPs: 0 up, 0 with errors, 0 timed out (archived)
 CCM generation enabled: No
 AIS generation enabled: Yes (level: 7, interval: 1s)
 Sending AIS:
                     Yes (started 01:32:56 ago)
 Receiving AIS:
                    Yes (from lower MEP, started 01:32:56 ago)
 Packet
           Sent Received
 _____ ____
           20
5576
                        20 (out of seq: 0)
0
 CCM
 ATS
```

```
Domain fred (level 5), Service barney
Up MEP on GigabitEthernet0/1/0/0.234, MEP-ID 2
_____
 Interface state: Up
                   MAC address: 1122.3344.5566
 Peer MEPs: 3 up, 2 with errors, 0 timed out (archived)
 Cross-check defects: 0 missing, 0 unexpected
 CCM generation enabled: Yes (Remote Defect detected: Yes)
 CCM defects detected: R - Remote Defect received
                    P - Peer port down
                    C - Config (our ID received)
 AIS generation enabled: Yes (level: 6, interval: 1s)
                    Yes (to higher MEP, started 01:32:56 ago)
 Sending AIS:
 Receiving AIS:
                    No
                   Received
 Packet
           Sent
  -----
                 _____
                   67890 (out of seq: 6, discarded: 10)
 CCM
          12345
                     0
5 (out of seq: 0, with bad data: 0)
            5
 T'BW
              0
 LBR
              0
                     46910
 ATS
 LCK
                         0
```

Example 4: Detail

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

RP/0/RSP0/CPU0:router# show ethernet cfm local meps detail

```
Domain foo (level 6), Service bar
Up MEP on GigabitEthernet0/1/0/0.234, MEP-ID 100
_____
 Interface state: Up MAC address: 1122.3344.5566
 Peer MEPs: 0 up, 0 with errors, 0 timed out (archived)
 CCM generation enabled: No
 AIS generation enabled: Yes (level: 7, interval: 1s)
                      Yes (started 01:32:56 ago)
 Sending AIS:
                     Yes (from lower MEP, started 01:32:56 ago)
 Receiving AIS:
Domain fred (level 5), Service barney
Up MEP on GigabitEthernet0/1/0/0.234, MEP-ID 2
_____
 Interface state: Up
                    MAC address: 1122.3344.5566
 Peer MEPs: 3 up, 2 with errors, 0 timed out (archived)
 Cross-check defects: 0 missing, 0 unexpected
 CCM generation enabled: Yes (Remote Defect detected: Yes)
 CCM defects detected: R - Remote Defect received
                     P - Peer port down
                      C - Config (our ID received)
 AIS generation enabled: Yes (level: 6, interval: 1s)
 Sending AIS:
                      Yes (to higher MEP, started 01:32:56 ago)
                     No
 Receiving AIS:
```

EFD Configuration: Examples

This example shows how to enable EFD:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# ethernet cfm
```

RP/0/RSP0/CPU0:router(config-cfm) # domain D1 level 1 RP/0/RSP0/CPU0:router(config-cfm-dmn) # service S1 down-meps RP/0/RSP0/CPU0:router(config-cfm-dmn-svc) # efd

This example shows how to enable EFD logging:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# ethernet cfm
RP/0/RSP0/CPU0:router(config-cfm)# domain D1 level 1
RP/0/RSP0/CPU0:router(config-cfm-dmn)# service S1 down-meps
RP/0/RSP0/CPU0:router(config-cfm-dmn-svc)# log efd
```

Displaying EFD Information: Examples

The following examples show how to display information about EFD:

show efd interfaces Command: Example

This example shows how to display all interfaces that are shut down in response to an EFD action:

RP/0/RSP0/CPU0:router# show efd interfaces Server VLAN MA ============ Interface Clients ------GigE0/0/0/0.0 CFM

show ethernet cfm local meps detail Command: Example

EFD triggered:

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

```
RP/0/RSP0/CPU0:router# show ethernet cfm local meps detail
Domain foo (level 6), Service bar
Up MEP on GigabitEthernet0/1/0/0.234, MEP-ID 100
_____
 Interface state: Up
                    MAC address: 1122.3344.5566
 Peer MEPs: 0 up, 0 with errors, 0 timed out (archived)
 Cross-check errors: 2 missing, 0 unexpected
 CCM generation enabled: No
 AIS generation enabled: Yes (level: 7, interval: 1s)
 Sending AIS: Yes (started 01:32:56 ago)
                    Yes (from lower MEP, started 01:32:56 ago)
 Receiving AIS:
 EFD triggered:
                     Yes
Domain fred (level 5), Service barney
Up MEP on GigabitEthernet0/1/0/0.234, MEP-ID 2
_____
 Interface state: Up
                     MAC address: 1122.3344.5566
 Peer MEPs: 3 up, 0 with errors, 0 timed out (archived)
 Cross-check errors: 0 missing, 0 unexpected
 CCM generation enabled: Yes (Remote Defect detected: No)
 AIS generation enabled: Yes (level: 6, interval: 1s)
 Sending AIS:
                      No
 Receiving AIS:
                      No
```

No



Note

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

Configuration Examples for Ethernet SLA

This section includes the following examples:

Ethernet SLA Profile Type Configuration: Examples

These examples show how to configure the different profile types supported by Ethernet SLA.

Example 1

This example configures a profile named "Prof1" for CFM loopback measurements:

```
configure
ethernet sla
profile Profl type cfm-loopback
commit
```

Example 2

This example configures a profile named "Prof1" for CFM delay measurements. Setting this type allows you to configure the probe to measure additional one-way delay and jitter statistics:

```
configure
  ethernet sla
   profile Prof1 type cfm-delay-measurement
   commit
```

Ethernet SLA Probe Configuration: Examples

These examples show how to configure some of the packet options for an Ethernet CFM loopback probe.

Example 1

This example shows how to configure sending a group of 100 packets in 100 ms intervals and repeat that burst every 60 seconds. Packets are padded to a size of 9000 bytes as needed using a hexadecimal test pattern of "abcdabcd," and with a class of service value of 7:



Note The total length of a burst (packet count multiplied by the interval) must not exceed 1 minute.

```
configure
  ethernet sla
  profile Prof1 type cfm-loopback
  probe
    send burst every 60 seconds packet count 100 interval 100 milliseconds
```

```
packet size 9000 test pattern hex 0xabcdabcd
priority 7
commit
```

Example 2

This example has the same characteristics as the configuration in Example 1, but sends a single burst of 50 packets, one second apart:

```
configure
ethernet sla
profile Prof1 type cfm-loopback
probe
send burst once packet count 50 interval 1 second
packet size 9000 test pattern hex 0xabcdabcd
priority 7
commit
```

Example 3

This example shows how to configure a continuous stream of packets at 100 ms intervals for the duration of the probe. Packets are padded to a size of 9000 bytes as needed using a pseudo-random test pattern, and with a class of service value of 7:

```
configure
  ethernet sla
   profile Prof1 type cfm-loopback
   probe
    send burst every 60 seconds packet count 600 interval 100 milliseconds
   packet size 9000 test pattern pseudo-random
   priority 7
   commit
```

Profile Statistics Measurement Configuration: Examples

These examples show how to configure the different types of statistics measurement.

Example 1

This example shows the two available types of statistics that can be measured by a CFM loopback SLA profile type:

```
configure
  ethernet sla
   profile Prof1 type cfm-loopback
  statistics measure round-trip-delay
  statistics measure round-trip-jitter
   commit
```

Example 2

This example shows how to configure measurement of round-trip delay and one-way jitter (from destination to source) for a CFM delay measurement SLA profile type:

Note

The CFM delay measurement profile type supports measurement of all round-trip and one-way delay and jitter statistics.

```
configure
ethernet sla
profile Prof1 type cfm-delay-measurement
statistics measure round-trip-delay
statistics measure one-way-jitter-ds
commit
```

Scheduled SLA Operation Probe Configuration: Examples

These examples show how to configure different schedules for an SLA operation probe.

Example 1

This example shows how to configure a probe to run hourly for a specified duration:

```
configure
ethernet sla
profile Prof1 type cfm-delay-measurement
schedule every 1 hours for 15 minutes
commit
```

Example 2

This example shows how to configure a probe to run daily for a specified period of time:

```
configure
  ethernet sla
   profile Prof1 type cfm-delay-measurement
  schedule every day at 11:30 for 5 minutes
   commit
```

Example 3

This example shows how to configure a probe to run weekly beginning at a specified time and for a specified duration:

```
configure
  ethernet sla
  profile Prof1 type cfm-delay-measurement
  schedule every week on Monday at 23:30 for 1 hour
  commit
```

Ethernet SLA Operation Probe Scheduling and Aggregation Configuration: Example

This figure shows a more comprehensive example of how some of the probe scheduling and measurement configuration works using aggregation. The following configuration supports some of the concepts shown in the figure:

```
configure
ethernet sla profile Prof1 type cfm-loopback
```

```
probe
send packet every 60 seconds
schedule every 6 hours for 2 hours
statistics measure round-trip-delay
aggregate bins 3 width 30
buckets size 1 probes
buckets archive 4
commit
```

Figure 13: SLA Probe Scheduling Operation With Bin Aggregation



This example schedules a probe with the following characteristics:

- Sends packets 60 seconds apart (for a 2-hour probe, this results in sending 120 individual packets).
- Probe runs every 6 hours for 2 hours duration.
- Collects data into 1 bucket for every probe, so each bucket covers 2 hours of the 2-hour probe duration.
- Aggregates statistics within the buckets into 3 bins each in the following ranges:
 - Bin 1 contains samples in the range 0 to < 30 ms.
 - Bin 2 contains samples in the range 30 ms to < 60 ms.
 - Bin 3 contains samples in the range 60 ms or greater (unbounded).
- The last 4 buckets are saved in memory.

Ongoing Ethernet SLA Operation Configuration: Example

This example shows how to configure an ongoing Ethernet SLA operation on a MEP:

```
interface gigabitethernet 0/1/0/1
ethernet cfm
mep domain Dm1 service Sv1 mep-id 1
sla operation profile Profile_1 target mac-address 01:23:45:67:89:ab s
commit
end
```

On-Demand Ethernet SLA Operation Basic Configuration: Examples

These examples show how to configure on-demand Ethernet SLA operations.

Example 1

This example shows how to configure a basic on-demand Ethernet SLA operation for a CFM loopback probe that by default will measure round-trip delay and round-trip jitter for a one-time, 10-second operation to the target MEP:

RP/0//CPU0:router# ethernet sla on-demand operation type cfm-loopback probe domain D1 source interface TenGigE 0/6/1/0 target mep-id 1

Example 2

This example shows how to configure a basic on-demand Ethernet SLA operation for a CFM delay measurement probe that by default will measure one-way delay and jitter in both directions, as well as round-trip delay and round-trip jitter for a one-time, 10-second operation to the target MEP:

RP/0//CPU0:router# ethernet sla on-demand operation type cfm-delay-measurement probe domain
D1 source interface TenGigE 0/6/1/0 target mep-id 1

Ethernet SLA Y.1731 SLM Configuration: Examples

These examples show how to configure the synthetic loss measurement statistics.

Example 1

This example shows the default configuration for Y.1731 SLM:

```
ethernet sla
profile sl1 type cfm-synthetic-loss-measurement
statistic measure one-way-loss-sd
statistic measure one-way-loss-ds
```

Example 2

This example configures a profile named "Sl2" for synthetic loss measurements. with the parameters to configure the probe and SLM statistics:

```
ethernet sla
profile sl2 type cfm-synthetic-loss-measurement
probe
send burst every 5 seconds packet count
100 interval 50 milliseconds
packet size 400 test pattern hex 0xABDC1234
synthetic loss calculation packets 200
schedule every 1 hours for 1 minute
```

```
statistic measure one-way-loss-sd
statistic measure one-way-loss-ds
aggregate bins 3 width 30
bucket size 24 probes
```

Ethernet SLA Show Commands: Examples

These examples show how to display information about configured SLA operations:

show ethernet sla operations Command: Example 1

RP/0/RSP0/CPU0:router# show ethernet sla operations interface gigabitethernet 0/1/0/1.1

show ethernet sla configuration-errors Command: Example 2

RP/0/RSP0/CPU0:router# show ethernet sla configuration-errors

```
Errors:
-----
Profile 'gold' is not defined but is used on Gi0/0/0/0.0
Profile 'red' defines a test-pattern, which is not supported by the type
```

These examples show how to display the contents of buckets containing SLA metrics collected by probes:

show ethernet sla statistics current Command: Example 3

RP/0/RSP0/CPU0:router# show ethernet sla statistics current interface GigabitEthernet 0/0/0/0.0

```
Interface GigabitEthernet 0/0/0/0.0
Domain mydom Service myser to 00AB.CDEF.1234
Profile 'business-gold', packet type 'cfm-loopback'
Scheduled to run every Sunday at 4am for 2 hours
Round Trip Delay
2 buckets per probe
Bucket started at 04:00 Sun 17 Feb 2008 lasting 1 hour:
Pkts sent: 2342; Lost 2 (0%); Corrupt: 0 (0%); Misordered: 0 (0%)
Min: 13ms; Max: 154ms; Mean: 28ms; StdDev: 11ms
Round Trip Jitter
2 buckets per probe
```

Bucket started at 04:00 Sun 17 Feb 2008 lasting 1 hour: Pkts sent: 2342; Lost: 2 (0%); Corrupt: 0 (0%); Misordered: 0 (0%) Min: -5ms; Max: 8ms; Mean: 0ms; StdDev: 3.6ms
Bucket started at 05:00 Sun 17 Feb 2008 lasting 1 hour: Pkts sent: 2342; Lost: 2 (0%); Corrupt: 0 (0%); Misordered: 0 (0%) Min: 0; Max: 4; Mean: 1.4; StdDev: 1

show ethernet sla statistics history detail Command: Example 4

```
RP/0/RSP0/CPU0:router# show ethernet sla history detail GigabitEthernet 0/0/0/0.0
Interface GigabitEthernet 0/0/0/0.0
Domain mydom Service myser to 00AB.CDEF.1234
_____
Profile 'business-gold', packet type 'cfm-loopback'
Scheduled to run every Sunday at 4am for 2 hours
Round Trip Delay
2 buckets per probe
Bucket started at 04:00 Sun 17 Feb 2008 lasting 1 hour:
   Pkts sent: 2342; Lost: 2 (0%); Corrupt: 0 (0%); Misordered: 0 (0%)
   Min: 13ms, occurred at 04:43:29 on Sun 22 Aug 2010 UTC
   Max: 154ms, occurred at 05:10:32 on Sun 22 Aug 2010 UTC
   Mean: 28ms; StdDev: 11ms
   Results suspect as more than 10 seconds time drift detected
   Results suspect as scheduling latency prevented some packets being sent
   Samples:
                Result Notes
   Time sent
   ----- -----
   04:00:01.324 23ms
               36ms
- Timed Out
   04:00:01.425
   04:00:01.525
   . . .
Round Trip Jitter
2 buckets per probe
Bucket started at 04:00 Sun 17 Feb 2008, lasting 1 hour:
   Pkts sent: 2342; Lost: 2 (0%); Corrupt: 0 (0%); Misordered: 0 (0%)
   Min: -5ms; Max: 10ms; Mean: 0ms; StdDev: 3.6ms
   Samples:
   Time sent
               Result Notes
   -----
                        _____
   04:00:01.324
                     _
               13ms
   04:00:01.425
   04:00:01.525
                   - Timed out
```

show ethernet sla statistics history detail on-demand: Example 5

This example shows how to display statistics for all full buckets for on-demand operations in detail:

RP/0//CPU0/router #show ethernet sla statistics history detail on-demand

```
Interface GigabitEthernet0/0/0/0.1
Domain mydom Service myser to 0123.4567.890A
_____
On-demand operation ID #1, packet type 'cfm-delay-measurement'
Started at 15:38 on 06 July 2010 UTC, runs every 1 hour for 1 hour
Round Trip Delay
1 bucket per probe
Bucket started at 15:38 on Tue 06 Jul 2010 UTC, lasting 1 hour:
   Pkts sent: 1200; Lost: 4 (0%); Corrupt: 600 (50%); Misordered: 0 (0%)
   Min: 13ms, occurred at 15:43:29 on Tue 06 Jul 2010 UTC
   Max: 154ms, occurred at 16:15:34 on Tue 06 Jul 2010 UTC
   Mean: 28ms; StdDev: 11ms
   Bins:
                 Samples Cum. Count
   Range
                                        Mean
   0 - 20 ms 194 (16%) 194 (16%)
                                         17ms
   20 - 40 ms
                 735 (61%)
                            929 (77%)
                                         27ms
   40 - 60 ms
                 212 (18%)
                            1141 (95%)
                                         45ms
                          1196
   > 60 ms
                 55 (5%)
                                         70ms
Bucket started at 16:38 on Tue 01 Jul 2008 UTC, lasting 1 hour:
   Pkts sent: 3600; Lost: 12 (0%); Corrupt: 1800 (50%); Misordered: 0 (0%)
   Min: 19ms, occurred at 17:04:08 on Tue 06 Jul 2010 UTC
   Max: 70ms, occurred at 16:38:00 on Tue 06 Jul 2010 UTC
   Mean: 28ms; StdDev: 11ms
   Bins:
   Range
                  Samples
                          Cum. Count
                                        Mean
   _____ ____
               194 (16%) 194 (16%)
                                       19ms
   0 - 20 ms
   20 - 40 ms
               735 (61%)
                           929 (77%)
                                        27ms
   40 - 60 ms
                212 (18%)
                          1141 (95%)
                                         45ms
                 55 (5%)
   > 60 ms
                           1196
                                         64ms
```

show ethernet sla statistics profile Command: Example 6

These examples show how to display statistics for synthetic loss measurement in detail:

RP/0/RSP0/CPU0:router#show ethernet sla statistics profile sl2 statistic one-way-loss-sd detail

Source: Interface GigabitEthernet0/0/0/0, Domain dom1
Destination: Target MAC Address 0002.0003.0005
Profile 'sl1', packet type 'cfm-synthetic-loss-measurement'
Scheduled to run every 1hr first at 00:50:00 UTC for 1min
Frame Loss Ratio calculated every 10s
One-way Frame Loss (Source->Dest)
One-way Frame Loss (Source->Dest)
I probes per bucket
Bucket started at 04:50:00 PDT Thu 15 September 2012 lasting 1hr
Pkts sent: 1200; Lost: 27 (2.25%); Corrupt: 0 (0.0%);
Min: 0.00%, occurred at 04:50:50 PDT Thu 15 September 2011
Max: 5.50%, occurred at 04:50:20 PDT Thu 15 September 2011
Mean: 2.08%; StdDev: 1.99%; Overall: 2.08%

Measurements:			
Time		Result	Notes
04:50:00.0	1.50%	(3 of 200)	
04:50:10.0	2.00%	(4 of 200)	
04:50:20.0	5.50%	(11 of 200)	
04:50:30.0	3.00%	(6 of 200)	
04:50:40.0	0.50%	(1 of 200)	
04:50:50.0	0.00%	(0 of 200)	

In the example 6, the description of the statistics that indicate the lost count and overall FLR are Lost: 27 (2.25%) and Overall: 2.08%. The lost count means that 27 SLMs were lost out of 1200, but it might not be possible to determine in which direction they were lost. The overall FLR reports the overall loss in the Source to Destination direction.

show ethernet sla statistics profile Command: Example 7

```
RP/0/RSP0/CPU0:ios#show ethernet sla statistics profile sl2 statistic one-way-loss-ds detail
Source: Interface GigabitEthernet0/0/0/0, Domain dom1
Destination: Target MAC Address 0002.0003.0005
_____
Profile 'sl2', packet type 'cfm-synthetic-loss-measurement'
Scheduled to run every 1hr first at 00:55:00 UTC for 1min
Frame Loss Ratio calculated every 10s
One-way Frame Loss (Dest->Source)
24 probes per bucket
Bucket started at 04:55:00 PDT Thu 15 September 2012 lasting 1 day
   Pkts sent: 28800; Lost: 14691 (51.01%); Corrupt: 0 (0.0%);
                  Misordered: 0 (0.0%); Duplicates: 0 (0.0%)
   Min: 10.00%, occurred at 04:55:00 PDT Thu 15 September 2011
   Max: 68.80%, occurred at 06:55:00 PDT Thu 15 September 2011
   Mean: 52.5%; StdDev: 0.00%; Overall: 51.00%
   Bins:
   Range
                  Count Cum. Count
                                    Mean
   _____
              _____
                         _____
                                   ____
    0 to 30% 20 (13.9%) 20 (13.9%) 21.00%
    30 to 60% 71 (49.3%) 91 (63.2%) 57.90%
    60 to 100% 49 (34.0%) 144 (100.0%) 62.00%
```

Configuration Example for Ethernet LMI

Figure 16 shows a basic E-LMI network environment with a local UNI defined on a Cisco ASR 9000 Series Router functioning as the PE using Gigabit Ethernet interface 0/0/0/0, and connectivity to a remote UNI over Gigabit Ethernet interface 0/0/0/1.

Figure 14: Basic E-LMI UNI and Remote UNI Diagram



The following configuration provides a basic E-LMI configuration for the environment shown in Figure 16, for the Cisco ASR 9000 Series Router as the PE device on the local UNI with physical Gigabit Ethernet interfaces 0/0/0/0 and 0/0/0/1:

```
RP/0/RSP0/CPU0:router# configure
1
! Configure the Local UNI EFPs
RP/0/RSP0/CPU0:router(config) # interface GigabitEthernet0/0/0/0.0 l2transport
RP/0/RSP0/CPU0:router(config-subif) # #encapsulation dot1q 1-20
RP/0/RSP0/CPU0:router(config-subif) # exit
RP/0/RSP0/CPU0:router(config) # interface GigabitEthernet0/0/0/1.1 l2transport
RP/0/RSP0/CPU0:router(config-subif)# #encapsulation dot1g 1-20
RP/0/RSP0/CPU0:router(config-subif) # exit
! Create the EVC
1
RP/0/RSP0/CPU0:router(config)# l2vpn
RP/0/RSP0/CPU0:router(config-l2vpn) # bridge group BG1
RP/0/RSP0/CPU0:router(config-l2vpn-bg) # bridge-domain BD1
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# interface GigabitEthernet0/0/0/0.0
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# interface GigabitEthernet0/0/0/1.1
RP/0/RSP0/CPU0:router(config=l2vpn-bg-bd)# exit
RP/0/RSP0/CPU0:router(config-12vpn-bg)# exit
RP/0/RSP0/CPU0:router(config-l2vpn)# exit
1
! Configure Ethernet CFM
1
RP/0/RSP0/CPU0:router(config) # ethernet cfm
RP/0/RSP0/CPU0:router(config-cfm) # domain GLOBAL level 5
RP/0/RSP0/CPU0:router(config-cfm-dmn) # service CustomerA bridge group BG1 bridge-domain BD1
RP/0/RSP0/CPU0:router(config-cfm-dmn-svc) # continuity-check interval 100ms
RP/0/RSP0/CPU0:router(config-cfm-dmn-svc)# mep crosscheck mep-id 22
RP/0/RSP0/CPU0:router(config-cfm-dmn-svc) # mep crosscheck mep-id 11
RP/0/RSP0/CPU0:router(config-cfm-dmn-svc) # exit
RP/0/RSP0/CPU0:router(config-cfm-dmn)# exit
RP/0/RSP0/CPU0:router(config-cfm) # exit
!
! Configure EFPs as CFM MEPs
1
RP/0/RSP0/CPU0:router(config) # interface GigabitEthernet0/0/0/0.0 l2transport
RP/0/RSP0/CPU0:router(config-subif) # ethernet cfm
RP/0/RSP0/CPU0:router(config-if-cfm) # mep domain GLOBAL service CustomerA mep-id 22
RP/0/RSP0/CPU0:router(config-if-cfm) # exit
RP/0/RSP0/CPU0:router(config-subif) # exit
!
! Configure the Local UNI Name
RP/0/RSP0/CPU0:router(config) # interface GigabitEthernet 0/0/0/0
```

```
RP/0/RSP0/CPU0:router(config-if)# ethernet uni id PE1-CustA-Slot0-Port0
RP/0/RSP0/CPU0:router(config-if)# exit
! Enable E-LMI on the Local UNI Physical Interface
!
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/0/0/0
RP/0/RSP0/CPU0:router(config-if)# ethernet lmi
RP/0/RSP0/CPU0:router(config-if)# exit
RP/0/RSP0/CPU0:router(config)# commit
```

Configuration Examples for Ethernet Data Plane Loopback

This example shows how to configure Ethernet Data Plane Loopback:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/1/0/1
RP/0/RSP0/CPU0:router(config-if-srv)# ethernet loopback permit external
```

This example shows how to start an Ethernet Data Plane Loopback:

```
RP/0/RSP0/CPU0:router# ethernet loopback start local interface gigabitEthernet
0/1/0/1
```

external

```
[source mac-address <addr>]
[destination mac-address <addr>]
[ether-type <etype>]
[{dot1q <vlan-ids> [second-dot1q <vlan-ids>] |
dot1ad <vlan-ids> [dot1q <vlan-ids>]}]
[cos <cos>]
[llc-oui <oui>]
[timeout {<length> | none}]
```

This example shows how to stop an Ethernet Data Plane Loopback session:

RP/0/RSP0/CPU0:router# ethernet loopback stop local interface <name> id <id>

This example shows how to extend an Ethernet Data Plane Loopback session:

RP/0/RSP0/CPU0:router# ethernet loopback extend local interface <name> id <id>
length
<length>

Verification

• Use the **show ethernet loopback permitted** command to display all the permitted interfaces which run Ethernet Data Plane Loopback sessions:

```
RP/0/RSP0/CPU0:router# show ethernet loopback permitted
Interface Direction
GigabitEthernet0/0/0/0 External
GigabitEthernet0/0/0/1.100 Internal
TenGigE0/1/0/0.200 External, Internal
```

• Use the show ethernet loopback active command to view active sessions:

RP/0/RSP0/CPU0:router# show ethernet loopback active interface TenGigE0/1/0/0.200

```
Local: TenGigE0/1/0/0.200, ID 1
_____
Direction: Internal
Time out: 2 hours
Time left: 00:01:17
Status: Active
Filters:
Dot1ad: 100-200
Dotlg: Anv
Source MAC Address: aaaa.bbbb.cccc
Destination MAC Address: Any
Ethertype: 0x8902
Class of Service: Any
LLC-OUT: Any
Local: TenGigE0/1/0/0.200, ID 2
_____
Direction: External
Time out: 10 minutes
Time left: 00:00:00
Status: Stopping
Filters:
Dot1q: 500
Second-dot1q: 200
Source MAC Address: Any
Destination MAC Address: Any
Ethertype: Any
Class of Service: 4
LLC-OUI: Any
```

For each loopback session listed, this information is displayed:

- Header containing the Interface name and session ID, which uniquely identify the local loopback session,
- · Direction which specifies the direction of the loopback,
- Time out the time out period specified when the loopback was started,
- Time left the amount of time left until the loopback session is automatically stopped,
- Status the status of the loopback session,
- Filters details of the filters specified when the loopback session was started. Similar to the start CLI, only the filters supported by the platform are displayed.

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 on the Cisco ASR 9000 Series Router" module 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

CFM Adaptive Bandwidth Notifications

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

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



A generic solution to this traffic drop issue is Connectivity Fault Management (CFM) extension to send Bandwidth Notifications Messages (BNM) to Maintenance Endpoints (MEPs) on the corresponding interface at the head-end router. To be flexible in the actions that are to be taken, the solution uses Embedded Event Manager (EEM) to invoke operator written TCL scripts. For information on EEM, see Embedded Event Manager, on page 220.

Note

This feature is supported only on 64-bit Linux-based IOS XR ASR 9000 operating system.

Bandwidth Notification Messages

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

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

Both message types contain the following information:

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

- · Nominal Bandwidth
- Current Bandwidth

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



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

Restrictions for CFM Bandwidth Notifications

These are the restrictions of CFM Bandwidth Notifications:

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

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

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

Bandwidth Reporting

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

Devices sending BNMs can detect changes in bandwidth many times in a second. For example, changes caused by an object passing through a microwave link's line of sight. The protocol for sending BNMs is designed to mitigate fluctuating current bandwidth by sampling across a 'monitoring-interval' and applying basic damping to degradation events. To help mitigate this further, a damping algorithm is used. This algorithm is applied on the receiving device, and is distinct from any damping performed by the sender. For more information, see Damping Algorithm, on page 218.

An operator may be interested in more than one BNM enabled link, and needs the ability to register on a set of BNM enabled links that affect the path to a node in the network. To do this, the state and RBW for each link of interest are put into a conformance testing algorithm, which both filters and changes the rate-limits to publish events notifying the operator only of significant changes. For more information, see Conformance Testing Algorithm, on page 220.

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



• If a degraded link having bandwidth higher than the threshold receives BNM with bandwidth less than the threshold, it doesn't wait for the hold-off timer and instantly changes the bandwidth by invoking EEM script.

Damping Algorithm

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

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

The following internal state transition diagram shows how the damping algorithm executes:



Conformance Testing Algorithm

The conformance testing algorithm comprises of two parts:

1. Filtering bandwidth changes.

Filtering is done so that events are published whenever either:

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

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

Embedded Event Manager

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

Typical actions taken in response to signal degradation events include:

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

The following variables can be queried within the TCL script:

EEM Variables	Comment
interface, level, direction	Identify the MEP in the registration
min_reported_bandwidth	Minimum reported bandwidth across all links in the registration that are currently in the DEGRADED state, and below the specified threshold.
<pre>bnm_enabled_links [{ MAC address Port ID }]</pre>	Array of BNM enabled links, with each one containing the following elements:
	• reported_bw: Reported Bandwidth
	• nominal_bw: Nominal BW in last BNM
event_type	Either 'DEGRADED' or 'OK'
	DEGRADED indicates that at least one BNM enabled link in the registration is in the DEGRADED state with a reported bandwidth less than the threshold value.
	This means that the event_type could be 'OK' if all BNM enabled links in the registration, which are in the DEGRADED state have a reported bandwidth greater than or equal to the threshold.

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

To configure EEM, use the following commands:

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

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

event manager policy <script_name> username lab persist-time <time>

Event Publishing

CFM publishes events for a given EEM registration after applying the damping and conformance testing algorithms as described in Damping Algorithm, on page 218 and Conformance Testing Algorithm, on page 220 respectively. The set of BNM Enabled Links published in an event are those in the DEGRADED state and whose RBW is less than the specified threshold.

Configuring CFM Bandwidth Notifications

Use the following steps to configure CFM bandwidth notifications:

- Configure a CFM domain at the level BNMs are expected to be received at, and a CFM service in the direction (either up or down-MEPs) the BNMs are expected to be received.
- Configure a CFM MEP on the interface expected to receive BNMs in the domain and service.

Configuration consists of two parts:

Configuring global CFM. This is similar to Continuity Check Message (CCM) and other CFM configurations.

Global CFM configuration:

```
ethernet cfm
domain DM1 level 2 id null
  service SR1 down-meps
  !
!
domain dom1 level 1
  service ser1 down-meps
  !
!
```

 Configuration related to CFM-BNMs under interfaces. This is optional and used for changing default values.

Interface configuration:

```
Interface TenGigE0/0/1/1
ethernet cfm
 mep domain DM1 service SR1 mep-id 3001
 bandwidth-notifications
  hold-off 0
   wait-to-restore 60
   loss-threshold 10
   log changes
  1
!
12transport
1
interface TenGigE0/0/0/3
ethernet cfm
 mep domain dom1 service ser1 mep-id 11
  1
 bandwidth-notifications
 hold-off 10
  wait-to-restore 40
   log changes
  1
1
12transport
1
1
```

Running Configuration

RP/0/RP0/CPU0:router#show running-configuration
!! IOS XR Configuration 7.1.1.104I

```
!! Last configuration change at Mon Jun 24 21:26:46 2019 by root
1
hostname R2 cXR
logging console debugging
logging buffered 12500000
event manager directory user policy harddisk:/tcl/
event manager directory user library harddisk:/tcl/
event manager policy EEMmac lev1.tcl username root persist-time 3600
event manager policy EEMport lev1.tcl username root persist-time 3600
aaa authorization exec default local group tacacs+
aaa authorization eventmanager default local
1
ethernet cfm
domain DMO level 1 id null
 service SR0 down-meps
  continuity-check interval 1m
   mep crosscheck
   mep-id 1003
   1
   ais transmission interval 1s cos 4
   log ais
   log continuity-check errors
   log crosscheck errors
   log continuity-check mep changes
  1
 1
 domain DM1 level 2 id null
 service SR1 down-meps id number 1
   continuity-check interval 1m
   mep crosscheck
   mep-id 431
   1
   ais transmission interval 1m
   log ais
   log continuity-check errors
   log crosscheck errors
   log continuity-check mep changes
  1
 domain dom1 level 3 id string domain3
  service ser1 xconnect group XG1 p2p XC1 id number 2300
  mip auto-create all
   continuity-check interval 1m
   mep crosscheck
   mep-id 2030
   1
interface Loopback0
ipv4 address 30.30.30.30 255.255.255.255
1
interface MgmtEth0/RSP0/CPU0/0
ipv4 address 5.18.9.102 255.255.0.0
1
interface MgmtEth0/RSP0/CPU0/1
shutdown
1
interface TenGigE0/0/0/0
shutdown
1
interface TenGigE0/0/0/3.1 l2transport
 encapsulation dot1q 6
 ethernet cfm
 mep domain DM1 service SR1 mep-id 231
 bandwidth-notifications
  hold-off 50
```

wait-to-restore 50
loss-threshold 100
log changes
!

Verification

RP/0/RP0/CPU0:router#show ethernet cfm interfaces b	andwidth-notifications detail
BNM Enabled Links at Level 3 (Down MEP) for Gigabit	Ethernet/1
MAC Address 000a.000a.000a	
State (OK):	
Nominal Bandwidth:	3000 Mbps
Reported Bandwidth:	1000 Mbps
Elapsed time in this state:	00:00:13.000
Transitions into degraded state:	5000
Hold-off:	111s remaining
Last BNM received 00:00:10 ago	
Nominal Bandwidth:	1000 Mbps
Current Bandwidth:	2000 Mbps
Interval:	10s
Packet-type:	Cisco BW-VSM
Packets received:	20000
Port ID 7 (MAC Address 000c.000c.000c)	
State (DEGRADED):	
Nominal Bandwidth:	6000 Mbps
Reported Bandwidth:	2000 Mbps
Elapsed time in this state:	00:00:39.000
Transitions into degraded state:	10000
Wait-to-restore:	111s remaining
Last BNM received 00:00:33 ago	
Nominal Bandwidth:	2000 Mbps
Current Bandwidth:	4000 Mbps
Interval:	1min
Packet-type:	Cisco BW-VSM
Packets received:	40000



Configuring Integrated Routing and Bridging

This module describes the configuration of Integrated Routing and Bridging (IRB) on the Cisco ASR 9000 Series Aggregation Services Routers. IRB provides the ability to exchange traffic between bridging services on the Cisco ASR 9000 Series Router and a routed interface using a Bridge-Group Virtual Interface (BVI).

Modification
This feature was introduced on the Cisco ASR 9000 Series Router for the following line cards:
• 2-Port 10-Gigabit Ethernet, 20-Port Gigabit Ethernet Combination Line Cards (A9K-2T20GE-B and A9K-2T20GE-L)
• 4-Port 10-Gigabit Ethernet Line Cards (A9K-4T-B, -E, -L)
• 8-Port 10-Gigabit Ethernet DX Line Cards (A9K-8T/4-B, -E, -L)
• 8-Port 10-Gigabit Ethernet Line Cards (A9K-8T-B, -E, -L)
• 16-Port 10-Gigabit Ethernet Line Cards (A9K-16T/8-B, -E, -L)
• 40-Port Gigabit Ethernet Line Cards (A9K-40GE-B, -E, -L)

Feature History for IRB

• Support for the following IRB environment using the Cisco ASR 9000 SIP-700 with any supported SPA as the core-facing interface was added:
• Layer 3 routed traffic from the Cisco ASR 9000 SIP-700 to Layer 2 bridged interfaces on Gigabit Ethernet line cards supporting IRB.
• IPv4 unicast traffic only.
• Support for IRB/BVI was included on the Cisco ASR 9000 Enhanced Ethernet Line Card.
• Support for IPv6 unicast addressing for IRB and 6PE/6VPE support with BVI interfaces was added for the following line cards:
• 2-Port 10-Gigabit Ethernet, 20-Port Gigabit Ethernet Combination Line Cards (A9K-2T20GE-B and A9K-2T20GE-L)
• 4-Port 10-Gigabit Ethernet Line Cards (A9K-4T-B, -E, -L)
• 8-Port 10-Gigabit Ethernet DX Line Cards (A9K-8T/4-B, -E, -L)
• 8-Port 10-Gigabit Ethernet Line Cards (A9K-8T-B, -E, -L)
• 16-Port 10-Gigabit Ethernet Line Cards (A9K-16T/8-B, -E, -L)
• 40-Port Gigabit Ethernet Line Cards (A9K-40GE-B, -E, -L)
Support for IRB with 802.1ah BVI and Provider Backbone Bridge (PBB)



Note Bridge-Group Virtual Interface (BVI) is not supported on ASR 9000 16-port 100GE QSFP TR (A9K-16X100GE-TR), ASR 9900 16-port 100GE QSFP SE (A99-16X100GE-X-SE) and ASR 9900 32-port 100GE QSFP TR (A99-32X100GE-TR) line cards.

- Prerequisites for Configuring IRB, on page 226
- Guidelines and Restrictions for Configuring IRB, on page 227
- Information About Configuring IRB, on page 228
- How to Configure IRB, on page 234
- Configuration Examples for IRB, on page 242

Prerequisites for Configuring IRB

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

Before configuring IRB, be sure that these tasks and conditions are met:

• Confirm that you are configuring only the following types of Gigabit Ethernet 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:

- 2-Port 10-Gigabit Ethernet, 20-Port Gigabit Ethernet Combination Line Cards (A9K-2T20GE-B and A9K-2T20GE-L)
- 4-Port 10-Gigabit Ethernet Line Cards (A9K-4T-B, -E, -L)
- 8-Port 10-Gigabit Ethernet DX Line Cards (A9K-8T/4-B, -E, -L)
- 8-Port 10-Gigabit Ethernet Line Cards (A9K-8T-B, -E, -L)
- 16-Port 10-Gigabit Ethernet Line Cards (A9K-16T/8-B, -E, -L)
- 40-Port Gigabit Ethernet Line Cards (A9K-40GE-B, -E, -L)
- 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 for Configuring IRB

Before configuring IRB, consider these restrictions:

- Only one BVI can be configured in any bridge domain.
- The same BVI can not be configured in multiple bridge domains.
- The following areas are not supported on the BVI:
 - IP fast reroute (FRR)
 - MoFRR
 - Traffic mirroring
 - Unnumbered interface for BVI
 - Video monitoring (Vidmon)
- IRB is supported on the following line cards:
 - Cisco ASR 9000 High Density 100GE Ethernet line cards
 - A9K-16X100GE-TR
 - A99-32X100GE-TR
 - A99-32X100GE-X-SE
 - A99-32X100GE-X-TR
 - A9K-8HG-FLEX-TR
 - A9K-4HG-FLEX-FC

- A99-4HG-FLEX-FC
- A9K-400GE-SE/-TR
- A99-400GE-SE/-TR
- A99-10X400GE-X-SE
- A99-10X400GE-X-TR
- Multi protocol Label Switching (MPLS) on BVI is supported on Cisco ASR 9000 Enhanced Ethernet Line Cards but not on Cisco ASR 9000 Ethernet Line Cards.
- IRB with 802.1ah (BVI and Provider Backbone Bridge (PBB) should not be configured in the same bridge domain).
- PIM snooping. (Need to use selective flood.)
- VRF-aware DHCP relay 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

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 ASR 9000 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.





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 a 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
- Quality of Service (QoS)
- 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 ASR 9000 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 16: 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 ASR 9000 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), Routing Information Protocol Version 2 (RIPv2), 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 VRRF/HSRP VMAC per 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 ASR 9000 Series Router.
- IP SLA support as currently supported on Layer 3 subinterfaces on the Cisco ASR 9000 Series Router.
- Load balancing of BVIs as ECMP paths (up to 32 paths).
- Interface-MIB.
- · Packet counters for BVI interfaces.
- Multi-chassis link aggregation (LAG) on link bundles that are members of a bridge domain that uses a BVI.

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

Additional IPv4-Specific Environments Supported for IRB

- Configuration of up to a maximum of 2000 BVIs.
- Up to a maximum of128k IPv4 adjacencies.
- 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

- Configuration of up to a maximum of 2000 BVIs, with up to 512 of these BVIs that can support IPv6 addressing.
- Up to a maximum of 5k IPv6 adjacencies.
- 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 ASR 9000 Series Router as the PE device with the following restrictions:
 - Supported by the following line cards on the PE devices:

- 2-Port 10-Gigabit Ethernet, 20-Port Gigabit Ethernet Combination Line Cards (A9K-2T20GE-B and A9K-2T20GE-L)

- 4-Port 10-Gigabit Ethernet Line Cards (A9K-4T-B, -E, -L)
- 8-Port 10-Gigabit Ethernet DX Line Cards (A9K-8T/4-B, -E, -L)
- 8-Port 10-Gigabit Ethernet Line Cards (A9K-8T-B, -E, -L)
- 16-Port 10-Gigabit Ethernet Line Cards (A9K-16T/8-B, -E, -L)
- 40-Port Gigabit Ethernet Line Cards (A9K-40GE-B, -E, -L)
- A99-32X100GE-X-SE
- A99-32X100GE-X-TR
- A9K-8HG-FLEX-TR
- A9K-4HG-FLEX-FC
- A99-4HG-FLEX-FC
- A9K-400GE-SE/-TR
- A99-400GE-SE/-TR
- A99-10X400GE-X-SE
- A99-10X400GE-X-TR
- Up to 512 BVIs with IPv6 addressing can be supported.
- 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:

Configuring the Bridge Group Virtual Interface

To configure a BVI, complete the following steps.

Configuration Guidelines

Consider the following guidelines when configuring the BVI:

- The BVI must be assigned an IPv4 or IPv6 address that is in the same subnet as the hosts in the bridged segments.
- If the bridged network has multiple IP networks, then the BVI must be assigned secondary IP addresses for each network.

SUMMARY STEPS

- **1**. configure
- 2. interface bvi identifier
- **3.** ipv4 address *ipv4-address mask* [secondary] ipv6 address *ipv6-prefix/prefix-length* [eui-64] [route-tag *route-tag value*]
- 4. arp purge-delay seconds
- 5. arp timeout seconds
- 6. bandwidth rate
- 7. mac-address value1.value2.value3
- 8. mtu bytes
- 9. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters the global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	interface bvi identifier	Specifies or creates a BVI, where <i>identifier</i> is a number
	Example:	from 1 to 65535.
	<pre>RP/0/RSP0/CPU0:router(config)# interface bvi 1</pre>	
Step 3	ipv4 address <i>ipv4-address mask</i> [secondary] ipv6 address <i>ipv6-prefix/prefix-length</i> [eui-64] [route-tag <i>route-tag value</i>]	Specifies a primary or secondary IPv4 address or an IPv6 address for an interface.
	Example:	
	<pre>RP/0/RSP0/CPU0:router(config-if)# ipv4 address 10.10.0.4 255.255.255.0</pre>	

	Command or Action	Purpose
Step 4	arp purge-delay seconds Example:	(Optional) Specifies the amount of time (in <i>seconds</i>) to delay purging of Address Resolution Protocol (ARP) table entries when the interface goes down.
	<pre>RP/0/RSP0/CPU0:router(config-if)#arp purge-delay 120</pre>	The range is 1 to 65535. By default purge delay is not configured.
Step 5	arp timeout seconds Example:	(Optional) Specifies how long dynamic entries learned on the interface remain in the ARP cache.
	RP/0/RSP0/CPU0:router(config-if)# arp timeout 12200	The range is 30 to 2144448000 seconds. The default is 14,400 seconds (4 hours).
Step 6	bandwidth <i>rate</i>	(Optional) Specifies the amount of bandwidth (in kilobits
	Example:	per second) to be allocated on the interface. This number is used as the cost metric in routing protocols for the BVI.
	RP/0/RSP0/CPU0:router(config-if)# bandwidth 1000000	The range is 0 to 4294967295. The default is 10000000 (10 Gbps).
Step 7	mac-address value1.value2.value3	(Optional) Specifies the 48-bit MAC address for the BVI
	Example:	as three dotted-hexadecimal values, and overrides use of the default MAC address. The range for each value is 0000
	<pre>RP/0/RSP0/CPU0:router(config-if)# mac-address 1111.2222.3333</pre>	to ffff. A MAC address of all 0s is not supported.
Step 8	mtu bytes	(Optional) Specifies the maximum transmission unit (MTU)
	Example:	Size for packets on the interface. The range is 64 to 65535. The default is 1514.
	RP/0/RSP0/CPU0:router(config-if)# mtu 2000	
Step 9	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	RP/0/RSP0/CPU0:router(config-if)# end	
	or	exiting(yes/no/cancel)? [cancel]:
	RP/0/RSP0/CPU0:router(config-if)# commit	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
	• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring the Layer 2 AC Interfaces

To configure the Layer 2 AC interfaces for routing by a BVI, complete the following steps.

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 following types of cards supporting IRB on the Cisco ASR 9000 Series Router:

- 2-Port 10-Gigabit Ethernet, 20-Port Gigabit Ethernet Combination Line Cards (A9K-2T20GE-B and A9K-2T20GE-L)
- 4-Port 10-Gigabit Ethernet Line Cards (A9K-4T-B, -E, -L)
- 8-Port 10-Gigabit Ethernet DX Line Cards (A9K-8T/4-B, -E, -L)
- 8-Port 10-Gigabit Ethernet Line Cards (A9K-8T-B, -E, -L)
- 40-Port Gigabit Ethernet Line Cards (A9K-40GE-B, -E, -L)

SUMMARY STEPS

- 1. configure
- 2. interface [GigabitEthernet | TenGigE] interface-path-id[.subinterface] l2transport
- 3. encapsulation dot1q vlan-id [exact]orencapsulation dot1ad vlan-id dot1q vlan-id
- **4**. rewrite ingress tag pop {1 | 2} symmetric
- 5. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	<pre>interface [GigabitEthernet TenGigE] interface-path-id[.subinterface] l2transport</pre>	Enables Layer 2 transport mode on a Gigabit Ethernet or 10-Gigabit Ethernet interface or subinterface and enters
	Example:	interface or subinterface configuration mode, where <i>interface-path-id</i> is specified as the <i>rack/slot/module/port</i>
	<pre>RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/1/0/0.1 l2transport</pre>	location of the interface and <i>.subinterface</i> is the optional subinterface number.

I

	Command or Action	Purpose
Step 3	encapsulation dot1q vlan-id [exact]orencapsulation dot1ad vlan-id dot1q vlan-id	(Optional) Specifies IEEE 802.1q encapsulation on the specified VLAN only.
	Example:	
	<pre>RP/0/RSP0/CPU0:router(config-if)# encapsulation dot1q 1 exact</pre>	
Step 4	rewrite ingress tag pop {1 2} symmetric	(Required if VLAN tagging configured) Specifies that one
	Example:	be removed from frames arriving at the ingress interface to
	RP/0/RSP0/CPU0:router(config-if)# rewrite ingress	the bridge domain.
	tag pop i Symmetric	 If configuring double tags using dot1ad and dot1q encapsulation, you need to use the rewrite ingress tag pop 2 symmetric command.
Step 5	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	RP/0/RSP0/CPU0:router(config-if)# end	
	or	<pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre>
	RP/0/RSP0/CPU0:router(config-if)# commit	• Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring a Bridge Group and Assigning Interfaces to a Bridge Domain

To configure a bridge group and assign interfaces to a bridge domain, complete the following steps.

SUMMARY STEPS

- 1. configure
- 2. l2vpn

- **3.** bridge group bridge-group-name
- 4. bridge-domain bridge-domain-name
- 5. interface [GigabitEthernet | TenGigE] interface-path-id[.subinterface]
- 6. end or commit

DETAILED STEPS

Procedure

I

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	l2vpn	Enters L2VPN configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router(config)# l2vpn	
Step 3	bridge group bridge-group-name	Creates a bridge group and enters L2VPN bridge group
	Example:	configuration mode.
	RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group 10	
Step 4	bridge-domain bridge-domain-name	Creates a bridge domain and enters L2VPN bridge group
	Example:	bridge domain configuration mode.
	RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain BD_1	
Step 5	interface [GigabitEthernet TenGigE]	Associates the Gigabit Ethernet and 10-Gigabit Ethernet
	interface-path-ia[.subinterface]	L2VPN bridge group bridge domain attachment circuit
	Example.	configuration mode, where <i>interface-path-id</i> is specified as the <i>rack/slot/module/port</i> location of the interface and
	RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# interface GigabitEthernet 0/1/0/0.1	<i>subinterface</i> is the optional subinterface number.
		Repeat this step for as many interfaces as you want to associate with the bridge domain.
Step 6	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	RP/0/RSP0/CPU0:router(config=12vpn-bg-bd-ac)# end	
	or	Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:

 Command or Action	Purpose
 RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-ac)# commit	• Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
	• Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
	• Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
	• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Associating the BVI as the Routed Interface on a Bridge Domain

To associate the BVI as the routed interface on a bridge domain, complete the following steps.

SUMMARY STEPS

- 1. configure
- 2. l2vpn
- 3. bridge group bridge-group-name
- 4. bridge-domain bridge-domain-name
- 5. routed interface bvi identifier
- 6. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	l2vpn	Enters L2VPN configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router(config)# 12vpn	
Step 3	bridge group bridge-group-name	Creates a bridge group and enters L2VPN bridge group
	Example:	configuration mode.

	Command or Action	Purpose
	RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group BG_test	
Step 4	bridge-domain bridge-domain-name Example:	Creates a bridge domain and enters L2VPN bridge group bridge domain configuration mode.
	RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain 1	
Step 5	routed interface bvi identifier	Associates the specified BVI as the routed interface for the
	Example:	interfaces assigned to the bridge domain.
	<pre>RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# routed interface bvi 1</pre>	
Step 6	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	RP/0/RSP0/CPU0:router(config-12vpn-bg-bd)# end	
	or	<pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre>
	RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# commit	• Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Displaying Information About a BVI

To display information about BVI status and packet counters, use the following commands:

show interfaces byi <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
!
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config) # interface bvi 1
RP/0/RSP0/CPU0:router(config-if))# ipv4 address 10.10.0.4 255.255.255.0
RP/0/RSP0/CPU0:router(config-if))# exit
! Configure the Layer 2 AC interface
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/1/0/0 l2transport
RP/0/RSP0/CPU0:router(config-if))# exit
Т
! Configure the L2VPN bridge group and bridge domain and assign interfaces
RP/0/RSP0/CPU0:router(config)# 12vpn
RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group 10
RP/0/RSP0/CPU0:router(config=l2vpn-bg)# bridge-domain 1
RP/0/RSP0/CPU0:router(config-12vpn-bg-bd)# interface GigabitEthernet 0/1/0/0
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-if)# exit
! Associate a BVI to the bridge domain
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# routed interface bvi 1
RP/0/RSP0/CPU0: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
!
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface bvi 1
RP/0/RSP0/CPU0:router(config-if))# ipv4 address 10.10.0.4 255.255.255.0
RP/0/RSP0/CPU0:router(config-if))# exit
!
! Configure the Layer 2 AC interfaces using dot1q encapsulation on a VLAN
!
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/1/0/0.1 l2transport
RP/0/RSP0/CPU0:router(config-if))# no ip address
RP/0/RSP0/CPU0:router(config-if))# no ip address
RP/0/RSP0/CPU0:router(config-if))# rewrite ingress tag pop 1 symmetric
RP/0/RSP0/CPU0:router(config-if))# exit
```

```
RP/0/RSP0/CPU0:router(config) # interface GigabitEthernet 0/1/0/1.1 l2transport
RP/0/RSP0/CPU0:router(config-if))# no ip address
RP/0/RSP0/CPU0:router(config-if))# encapsulation dot1q 1 exact
RP/0/RSP0/CPU0:router(config-if)) # rewrite ingress tag pop 1 symmetric
RP/0/RSP0/CPU0:router(config-if)) # exit
! Configure the L2VPN bridge group and bridge domain and assign interfaces
RP/0/RSP0/CPU0:router(config) # 12vpn
RP/0/RSP0/CPU0:router(config-l2vpn) # bridge group 10
RP/0/RSP0/CPU0:router(config-l2vpn-bg) # bridge-domain 1
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# interface GigabitEthernet 0/1/0/0.1
RP/0/RSP0/CPU0:router(config-12vpn-bg-bd)# interface GigabitEthernet 0/1/0/1.1
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-if)# exit
1
! Associate a BVI to the bridge domain
RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# routed interface bvi 1
RP/0/RSP0/CPU0:router(config-12vpn-bg-bd) # commit
```

IPv4 Addressing on a BVI Supporting Multiple IP Networks: Example

The following example shows how to configure secondary IPv4 addresses on a BVI that supports bridge domains for the 10.10.10.0/24, 10.20.20.0/24, and 10.30.30.0/24 networks. In this example, the BVI must have an address on each of the bridge domain networks:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface bvi 1
RP/0/RSP0/CPU0:router(config-if))# ipv4 address 10.10.10.4 255.255.255.0
RP/0/RSP0/CPU0:router(config-if))# ipv4 address 10.20.20.4 255.255.255.0 secondary
RP/0/RSP0/CPU0:router(config-if))# ipv4 address 10.30.30.4 255.255.255.0 secondary
RP/0/RSP0/CPU0: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
Т
interface Loopback0
ipv4 address 10.5.5.5 255.255.255.255
1
interface GigabitEthernet0/0/0/1
negotiation auto
1
interface GigabitEthernet0/0/0/1.1 l2transport
encapsulation dot1q 1
rewrite ingress tag pop 1 symmetric
interface GigabitEthernet0/0/0/1.2 l2transport
encapsulation dot1g 2
rewrite ingress tag pop 1 symmetric
1
```

```
interface GigabitEthernet0/0/0/9
```

```
bundle id 25 mode active
1
interface GigabitEthernet0/0/0/19
bundle id 25 mode active
T.
interface GigabitEthernet0/0/0/29
bundle id 25 mode active
1
interface GigabitEthernet0/0/0/39
bundle id 25 mode active
interface BVI1
ipv4 address 10.1.1.1 255.255.255.0
T.
interface BVI2
ipv4 address 10.1.2.1 255.255.255.0
router ospf 100
router-id 10.5.5.5
area O
 interface Bundle-Ether25
    interface Loopback0
   interface BVI1
   interface BVI2
  1
12vpn
bridge group IRB
 bridge-domain IRB1
   igmp snooping profile IRB SNOOP
   interface GigabitEthernet0/0/0/1.1
   1
   routed interface BVI1
  I
  bridge-domain IRB2
   igmp snooping profile IRB SNOOP
   interface GigabitEthernet0/0/0/1.2
   1
   routed interface BVI2
multicast-routing
address-family ipv4
 interface all enable
igmp snooping profile IRB SNOOP
report-suppression disable
I.
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.

12vpn

```
bridge group IRB
bridge-domain IRB-EDGE
interface GigabitEthernet0/0/0/8
!
routed interface BVI 100
!
interface GigabitEthernet0/0/0/8
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/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 ASR 9000 Series Router as the PE devices. For more information about Cisco 6PE/6VPE and its configuration, see the "Implementing IPv6 VPN Provider Edge Transport Over MPLS" chapter of the Cisco ASR 9000 Series Aggregation Services Router MPLS Layer 3 VPN Configuration Guide.

This figure shows the location of the BVI interfaces (green icons) on the Cisco ASR 9000 Series Routers as the PE1 and PE2 devices.

Figure 17: BVI Interfaces on the CE-Facing Sides in an MPLS 6PE/6VPE Network



This is a sample configuration only for the Cisco ASR 9000 Series Router (PE1) device with a BVI interface numbered 1 on the CE-facing side, and a non-BVI interface (Gigabit Ethernet 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 Gigabit Ethernet CE-facing interface to the
! L2VPn bridge domain where the routed BVI interface is also associated.
12vpn
bridge group 1
 bridge-domain 1
  interface Gigabit Ethernet 0/1/0/11
routed interface BVI1
! Configure OSPF routing for the BVI interface for
! advertisement of its IPv6 address.
Т
router ospfv3 1
graceful-restart
 redistribute bgp 1
area 1
 interface BVI1
 interface Loopback0
1
! 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.
1
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 vpnv6 unicast
 1
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
  1
  address-family ipv6 labeled-unicast
  !
  address-family vpnv6 unicast
   route-policy pass-all in
   route-policy pass-all out
  1
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 Gigabit Ethernet 0/1/0/37



Configuring Link Bundling

This module describes the configuration of link bundle interfaces on the Cisco ASR 9000 Series Aggregation Services Routers.

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). POS link bundles do not have mac address, only ethernet link bundles have mac address.

V	
---	--

Note The Cisco ASR 9000 Series Router supports both Layer 2 and Layer 3 Link Bundles. If the Link Bundle is a Layer 3 interface, an IP address is required. If the Link Bundle is a Layer 2 interface, an IP address is not required. A Link Bundle on the Cisco ASR 9000 Series 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. POS Link bundling is supported only on Layer 3 link bundles.

The Cisco ASR 9000 Series Router supports bundling for these types of interfaces:

• Ethernet interfaces and

Feature History for Configuring Link Bundling

Release	Modification	
Release 3.7.2	This feature was introduced on the Cisco ASR 9000 Series Router.	
Release 3.9.0	Support for load balancing was added.	
	Bundle member links are put into new err-disable link interface status and admin-down protocol state when a bundle interface is shut down.	
Release 3.9.1	Support for Layer 3 load balancing on Layer 2 link bundles was added.	
Release 4.0.0	The following support was added:	
	• Up to a maximum of 64 member links per bundle.	
	• IPv6 addressing.	
	Multichassis Link Aggregation.	

Release 4.0.1	Support for Dynamic Load Balancing for Link Aggregation (LAG) members was added.	
	The hw-module load-balance bundle l2-service l3-params command is replaced by the load-balancing flow command in L2VPN configuration mode. For more information see the Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Configuration Guide and Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Command Reference.	
Release 4.1.0	Support for Multi-Gigabit Service Control Point was added.	
Release 4.2.0	Support for Link bundling for POS interfaces was added.	
Release 4.3.1	Support for ICCP Based Service Multihoming was included.	
Release 5.1.1	Support for Multichassis Link Aggregation on IPv6 traffic was included.	
Release 5.1.2	Support for mixed speed member links in a bundle interface was included.	
Release 6.0.0	Bundle scale support increases to 1600 bundles on A9K-RSP880-SE, A99-RP2-SE and the third generation of ASR 9000 Ethernet line card.	
Release 6.2.2	Support for Layer 3 Multicast traffic over mixed speed bundles was added.	

- Prerequisites for Configuring Link Bundling, on page 250
- Information About Configuring Link Bundling, on page 251
- How to Configure Link Bundling, on page 268
- Configuring Multichassis Link Aggregation, on page 281
- How to Configure MGSCP, on page 297
- Configuration Examples for Link Bundling, on page 305
- Configuration Examples for MGSCP, on page 312

Prerequisites for Configuring Link Bundling

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.

The prerequisites for link bundling depend on the platform on which you are configuring this feature. This section includes the following information:

Prerequisites for Configuring Link Bundling on a Cisco ASR 9000 Series Router

Before configuring Link Bundling, be sure that the following tasks and conditions are met:

- You know the interface IP address (Layer 3 only).
- You know which links should be included in the bundle you are configuring.
- If you are configuring an Ethernet link bundle, you must have Ethernet line cards installed in the router.



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. A virtual interface is assigned to the bundled link. The component links can be dynamically added and deleted from the virtual interface.

The virtual interface is treated as a single interface on which one can configure an IP address and other software features used by the link bundle. Packets sent to the link bundle are forwarded to one of the links in the bundle.

A link bundle is simply a group of ports that are bundled together and act as a single link. The advantages of link bundles are as follows:

- Multiple links can span several line cards to form a single interface. Thus, the failure of a single link does not cause a loss of connectivity.
- Bundled interfaces increase bandwidth availability, because traffic is forwarded over all available members of the bundle. Therefore, traffic can flow on the available links if one of the links within a bundle fails. Bandwidth can be added without interrupting packet flow.

All the individual links within a single bundle must be of the same type.

For example, a bundle can contain all Ethernet interfaces, or it can contain all POS interfaces, but it cannot contain Ethernet and POS interfaces at the same time.

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. Links that are incompatible or have failed are automatically removed from a bundle.
- Ether Channel or POS Channel—Cisco proprietary technology that allows the user to configure links to join a bundle, but has no mechanisms to check whether the links in a bundle are compatible.(EtherChannel applies to Ethernet interfaces, and POS Channel applies to POS interfaces.)

Features and Compatible Characteristics of Ethernet Link Bundles

This list describes the properties and limitations of ethernet link bundles:

- Any type of Ethernet interfaces can be bundled, with or without the use of LACP (Link Aggregation Control Protocol).
- Bundle membership can span across several line cards that are installed in a single router or multiple routers in the case of MC-LAG.
- An ethernet link bundle can support a maximum of 64 physical links. If you add more than 64 links to a bundle, only 64 of the links are in distributing state, and the remaining links are in waiting state.
- A single Cisco ASR 9000 Series Router supports a maximum of 1600 bundles.
- Cisco ASR 9000 Series Router supports mixed speed bundles. Mixed speed bundles allow member links of different bandwidth to be configured as active members in a single bundle. The ratio of the bandwidth for bundle members must not exceed 10. Also, the total weight of the bundle must not exceed 64. For example, 100Gbps link and 10Gbps links can be active members in a bundle. Mixed speed bundles also allow:
 - · Load-balancing on member links based on bandwidth weightage.
 - Support of proportional load-balancing for all unicast flows.
 - Support of load-balancing based on bandwidth weightage for all VPLS flooding and Layer 2 multicast flows.
- Multi-Gigabit Service Control Point (MGSCP) is not supported for mixed speed bundles.
- With mixed speed bundles, the total weight of a bundle is greater than the number of members in the bundle, as the weight represents the smallest active number bandwidth.
- 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.
- Mixed speed bundles are supported in MC-LAG.
- 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. Beginning in Cisco IOS XR Release 3.9.0, when you shut down a bundle interface, the member links are put into err-disable link interface status and

admin-down line protocol state. You can show the status of a bundle interface and its members using the **show interfaces** command.

- 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.
- When LACP configured, each link within a bundle can be configured to allow different keepalive periods on different members.
- 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.
- Link layer protocols, such as CDP and HDLC keepalives, work independently on each link within a bundle.
- Upper layer protocols, such as routing updates and hellos, are sent over any member link of an ethernet interface bundle.
- All links within a single bundle must terminate on the same two systems. Both systems must be directly connected except in the case of MC-LAG.
- Bundled interfaces are point-to-point.
- A link must be in the up state before it can be in distributing state in a bundle.
- All links within a single bundle must be configured either to run 802.3ad (LACP) or Etherchannel (non-LACP). Mixed links within a single bundle are not supported.
- A bundle interface can contain physical links and VLAN subinterfaces only. Tunnels cannot be bundle members.
- Access Control List (ACL) configuration on link bundles is identical to ACL configuration on regular interfaces.
- Multicast traffic is load balanced over the members of a bundle. For a given flow, the control plane selects the member link, and all traffic for that flow is sent over that member. The system supports Layer 3 Multicast traffic over mixed speed bundles.
- When the LACP system receives priority value as zero (0) from a remote device, bundle does not come up.

Characteristics of POS Link Bundles in Cisco ASR 9000 Series Router

This section lists the properties of POS link bundles that are specific to Cisco ASR 9000 Series Router:

- Each bundle has to be configured between a pair of directly connected systems.
- All members of a bundle must be POS.
- POS link bundling can support up to 32 physical links if they are in the same speed. If links are in different speed, it cannot reach 32 physical links.
- Only physical interfaces can become bundle members.

- All bundles must be statically configured.
- Only cHDLC encapsulation type is currently supported on POS Link Bundle.
- Only POS SPA is supported for POS Link Bundling and not channelized SPA.
- Upper layer protocols, such as routing updates and hellos, are sent over through the bundle interface.
- Bandwidths for policers and queues must be in percentage and not in absolute values.
- Queue-limit must be in time unit and not in bytes.
- For POS link bundles, different link speeds are allowed within a single bundle, with a maximum of four times the speed difference between the members of the bundle. This means that only up to 4 times the bandwidth ratio is supported.

Restrictions of POS Link Bundles in Cisco ASR 9000 Series Router

This section lists the limitations of POS link bundles that are specific to Cisco ASR 9000 Series Router:

- LACP is not supported for POS link bundles in Cisco IOS XR Release 4.2.0 and later releases.
- IPv6 and ACL are not supported for POS link bundels in Cisco IOS XR Release 4.2.0.
- Multicast routing is not supported for POS link bundles in Cisco IOS XR Release 4.2.0.

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 Cisco ASR 9000 Series 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. These frames are analyzed to ensure both 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 bundle member, the following information is exchanged between the systems that host each end of the link bundle:

- A globally unique local system identifier
- An identifier (operational key) for the bundle of which the link is a member
- An identifier (port ID) for the link
- The current aggregation status of the link

This information is used to form the link aggregation group identifier (LAG ID). Links that share a common LAG ID can be aggregated. Individual links have unique LAG IDs.

The system identifier distinguishes one router from another, and its uniqueness is guaranteed through the use of a MAC address from the system. The bundle and link identifiers have significance only to the router assigning them, which must guarantee that no two links have the same identifier, and that no two bundles have the same identifier.

The information from the peer system is combined with the information from the local system to determine the compatibility of the links configured to be members of a bundle.

Bundle MAC addresses in the Cisco ASR 9000 Series 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 the user configures a different MAC address. The bundle MAC address is used by all member links when passing bundle traffic. Any unicast or multicast addresses set on the bundle are also set on all the member links.



Note We recommend that you avoid modifying the MAC address, because changes in the MAC address can affect packet forwarding.

Non Revertive Behavior for LACP Bundle Interface

In LACP, by default, a higher priority port would become the active port after it becomes operational again. To avoid this reversion, you can run the **lacp non-revertive** command. This configures the lower priority port to continue as the active port even after the higher priority port is capable of being operational. This avoids the traffic disruption that may happen in putting the currently active but lower priority port into standby and diverting traffic through the higher priority port that is now capable of being operational.

Multichassis Link Aggregation

The Multichassis Link Aggregation (MC-LAG) feature provides an end to end interchassis redundancy solution for the Carrier Ethernet Networks. MC-LAG involves two devices collaborating to act as a single LAG from the perspective of a (third) connected device, thus providing device-level as well as link-level redundancy.

To achieve this, two devices co-ordinate with each other to present a single LACP bundle (spanning the two devices) to a partner device. Only one of the devices forwards traffic at any one time, eliminating the risk of forwarding loops. When a failure occurs, these devices coordinate to perform a switchover, changing the device on which traffic is being forwarded by manipulating the link LACP states.

The existing pseudowire redundancy in the core network coordinates with the redundancy in the access network based on:

- Multichassis Link Aggregation Control Protocol (mLACP)
- Interchassis Communication Protocol (ICCP)

The mLACP protocol defines the expected behavior between the two devices and uses the Interchassis Control Protocol (ICCP) to exchange TLVs and identify peer devices to operate with. At the edge of a provider's network, a simple customer edge (CE) device that only supports standard LACP is connected to two provider edge (PE) devices. Thus the CE device is dual-homed, providing better L2 redundancy from the provider's side. In mLACP terminology, the CE device is referred to as a dual-homed device (DHD) and each PE device

is known as a point of attachment (POA). The POA forwarding traffic for the bundle is the active device for that bundle, while the other POA is the standby device.



MC-LAG in EVPN Multihoming-ELAN is not supported and alternative EVPN port-active should be used.

Failure Cases

MC-LAG provides redundancy, switching traffic to the unaffected POA while presenting an unchanged bundle interface to the DHD, for these failure events:

- Link failure: A port or link between the DHD and one of the POAs fails.
- Device failure: Meltdown or reload of one of the POAs, with total loss of connectivity (to the DHD, the core and the other POA).
- Core isolation: A POA loses its connectivity to the core network, and therefore is of no value, being unable to forward traffic to or from the DHD.

A loss of connectivity between the POAs leads both devices to assume that the other has experienced device failure, causing them to attempt to take on the Active role. This is known as a split brain scenario and can happen in either of the following cases:

- All other connectivity remains; only the link between POAs is lost.
- One POA is isolated from the core network (i.e. a core isolation scenario where the connection between the two POAs was over the core network).

MC-LAG by itself does not provide a means to avoid this situation; resiliency in the connection between the POAs is a requirement. The DHD is given the responsibility of mitigating the problem by setting a limit on the number of links, within the bundle, that can be active. As such only the links connected to one of the POAs can be active at any one point of time.

Interchassis Communication Protocol

This figure shows the graphical representation of the Interchassis Communication Protocol (ICCP).

Figure 18: ICCP Protocol



Two POAs communicate with each other over an LDP link using the Interchassis Communication Protocol (ICCP). ICCP is an LDP based protocol wherein an LDP session is created between the POAs in a redundancy group, and the ICCP messages are carried over that LDP session. The PE routers in a redundancy group may be a single-hop (directly connected) or a multi-hop away from one another. The ICCP protocol manages the setup and controls the redundancy groups. It also establishes, maintains, and tears down ICCP connections. The ICCP protocol uses route-watch to monitor the connectivity to the PEs in a given redundancy group. It is also responsible for tracking core isolation failures. It notifies all client applications of failure (core isolation and active PE failure).

To operate ICCP, the devices are configured as members of redundancy groups (RGs).

Note

In the mLACP configuration, two devices are configured to be members of each RG (until a device-level failure occurs leaving only a single member). However, each device can be a member of more than one RG.

In each redundancy group, a POA's mLACP peer is the other POA in that group, with which it communicates using mLACP over ICCP. For each bundle, the POA and DHD at each end are LACP partners, communicating using the standard LACP protocol.

Access Network Redundancy Model

The Multichassis Link Aggregation Control Protocol (mLACP) based redundancy between the customer edge device (CE) or access network and the provider edge (PE) device is achieved by allowing the CE to be connected to two PE routers. The two PE routers synchronize the data through ICCP; therefore they appear as a single device to the CE.

Figure 19: mLACP/ICCP Redundancy Model



The CE is also called dual-homed device (DHD) and the PE is also called point of attachment (POA). The pair of POAs that is connected to the single DHD forms a redundancy group (RG).

At any given time, only one POA is active for a bundle. Only the set of links between the DHD and the active POA actively sends traffic. The set of links between the DHD and the standby POA does not forward traffic. When the multichassis link bundle software detects that the connection to the active POA has failed, the

software triggers the standby POA to become the active POA, and the traffic flows using the links between the DHD and newly active POA.

Note When the connection to the active POA fails and the active POA is in the Negotiation state, the standby POA does not take over.

The ICCP protocol operates between the active and the standby POAs, and allows the POAs to coordinate their configuration, determine which POA is active, and trigger a POA to become active. Applications running on the two POAs (mLACP, IGMP snooping, DHCP snooping or ANCP) synchronize their state using ICCP.



Note

While upgrading to a newer version of Cisco ASR 9000 router, ensure that you upgrade the active and standby nodes to the same Cisco IOS XR version.

ICCP Based Service Multihoming

In the case of ICCP based Service Multihoming (ICCP-SM), the CE device uses two independent bundle interfaces to connect to the PoAs. Although bundle interfaces are used, they are not aggregated across the two chassis, and mLACP is not involved in the communication. The CE device configures the bundle interfaces in such a manner that all VLANs are allowed on both bundles. You can manually configure the PoAs to distribute the VLANs across the two bundles in order that individual VLANs are active(forwarding) on one bundle or PoA, and standby (blocked) on the other. The CE device initially floods a traffic flow on both bundles and learns the MAC address on the interface where it receives the response.

With ICCP-SM, you are not limited to a dual homed device. The access links can connect to a dual homed network (DHN) that are separate devices in the access network. The two bundles on the DHD or the DHN must be in a bridge domain so that L2 learning selects the link with the active set of VLANs.

Figure 20: ICCP Based Service Multihoming

If a bundle interface between the CE and the PoA fails, ICCP-SM on the PoA with the failed bundle communicates through ICCP to the other PoA's ICCP-SM. This activates the standby VLANs on the remaining bundle. A MAC flush is sent to the CE so that packets destined to hosts on the failed bundle are again flooded, in order to be learned on the newly activated bundle. The MAC flush is required because it is possible that the bundle interface failure is not detected by the CE.

In ICCP Based Service Multihoming, the total set of VLANs are split into a primary set and a secondary set and are configured on each PoA such that the primary set on one PoA is configured as secondary on the other. On each PoA, the VLANs are associated with ACs. If the VLANs are primary on a PoA and there are no faults, the associated ACs are set to forwarding. If the VLANs are secondary on a PoA, the associated ACs are blocked. ICCP-SM is only supported in VPLS cores.

ICCP-SM Redundancy Group

Prior to Release 6.4.1, ICCP-SM configurations were required to match on both peers for traffic flow. Primary VLANs on one PoA were configured as secondary VLANs on the other. If there was a mismatch in VLAN configuration, all primary and secondary VLANs were blocked. Also, when a user configured a new VLAN, there was a temporary mismatch which blocked other services until the VLAN configuration on both sides matched.

Starting from Release 6.4.1, when there is a mismatch in VLAN configuration on PoAs, instead of blocking all primary and secondary VLANs, only unmatched VLANs are blocked. That is, if at least one primary VLAN on one PoA matches the secondary VLAN on the other PoA, traffic flows. If misconfiguration is detected while there is no port failure, each PoA re-evaluates the configuration. The PoAs move misconfigured VLANs to the secondary VLAN list while the primary VLANs forward traffic. The misconfigured VLAN list also includes those VLANs that are only configured remotely. For example, if VLAN 1 is only configured on remote PoA and is not configured locally, VLAN 1 will still be in the secondary list locally.

When two PoAs are connected, if one PoA has been configured with Release 6.4.1 image and the other PoA has been configured with an image earlier to Release 6.4.1, the behavior of the PoAs with respect to forwarding and blocking the traffic is different from each other. The PoA with an image earlier to Release 6.4.1 blocks all VLANs when there is a mismatch in VLAN configuration. The PoA with Release 6.4.1 image forwards primary VLANs that are matched in secondary list on the peer PoA.

Advantages of Pseudo mLACP:

Pseudo mLACP has these three major advantages over mLACP:

- Pseudo mLACP can support a Dual Homed Network (DHN), while mLACP can only support a Dual Homed Device (DHD).
- Pseudo mLACP supports per-VLAN active/active redundancy without any load-balancing requirements on the CE.
- Pseudo mLACP does not require LACP support from the DHD, or DHN. It is independent of the access redundancy mechanism; therefore, it provides a network based redundancy solution. It allows maximum flexibility for the PE-CE interoperability in terms of dual-homing redundancy and recovery.

Failure Modes

The mLACP feature provides network resiliency by protecting against port, link, and node failures. This figure depicts the various failure modes.

Figure 21: Failure Modes



These are the failure categories:

• A—DHD uplink port failure. The port on the DHD that is connected to the POA fails.

- B—DHD uplink failure. The connection between the DHD and the POA fails.
- C—Active POA downlink port failure.
- D—Active POA node failure.
- E—Active POA uplink failure (network isolation). The links between the active POA and the core network fails

Note

ICCP Based Service Multihoming is similar to MC-LAG in the case of core network failures. It is revertive in nature. In the case of a failure, the PoA whose link has been restored activates the VLANs that are configured as primary.

Core Network Redundancy Model

This section explains:

One-way Pseudowire Redundancy

This figure shows the VPWS one-way pseudowire redundancy model. Only one end of the pseudowire is protected by a backup pseudowire.

Figure 22: VPWS one-way Pseudowire Redundancy



Two-way Pseudowire Redundancy

This figure shows the VPWS two-way pseudowire redundancy model. In this topology, each T-PE at the end of a PW has a primary and a backup PW. The state of the PW is coordinated with the state of the mLACP link between the DHD and the PE.



Figure 23: VPWS two-way Pseudowire Redundancy

Switchovers

Switchovers, which is changing the Active/Standby roles of the POAs, are performed using dynamic priority management or brute force behavior.

Dynamic Priority Management

Dynamic Priority Management involves co-ordination between the POAs to manipulate the LACP port priorities of their member links. Two priority values are tracked for each links:

- A configured priority which can either be configured explicitly, or defaults to 32768
- An operational priority used in LACP negotiations, which may differ from the configured priority if switchovers have occurred.

Higher priority LACP links are always selected ahead of lower priority LACP links. This means the operational priorities can be manipulated to force the standard LACP Selection Logic (on the POAs and on the DHD) to select desired links on both ends.

For example, consider a case where the DHD has two links to each POA, and each POA is configured with minimum-active links is 2. (This means the bundle goes down on the POA if the number of active links falls below 2.) The operational priorities for the member links are 1 on POA-1 and 2 on POA-2. This means that POA-1 is active (being higher priority) and the links on POA-2 are held in Standby state. The sequence of events in a switchover is as follows:

- **1.** A link fails on POA-1, causing the number of active links to fall below the minimum of 2.
- 2. POA-1 changes the operational priority of both its links to 3, so the links on POA 2 are now higher priority.
- **3.** POA-1 sends a LACP message to the DHD and an mLACP message to POA-2, informing both devices of the change.
- 4. The DHD tries to activate the links connected to POA-2 as these now have the highest priority.
- 5. POA-2 also ensures that its links have the highest priority and activates its links to the DHD.

At this point the switchover is complete.

Brute Force Behavior

In a brute force switchover, port priorities are not modified. Instead the failing POA sends a single *dying gasp* to the DHD over LACP, forcing it to deselect the link. It then terminates LACP communications on that link. This only leaves links between the DHD and POA-2, as links that can be selected. So, both ends select those links.

MC-LAG Topologies

This section illustrates the supported MC-LAG topologies.

Figure 24: VPWS One-way Pseudowire Redundancy in Redundancy Group





Figure 27: VPLS Pseudowires in Two Redundancy Groups

LACP Short Period Time Intervals

Load Balancing

Load balancing is a forwarding mechanism that distributes traffic over multiple links based on certain parameters. The Cisco ASR 9000 Series Router supports load balancing for all links in a bundle using Layer 2, Layer 3, and Layer 4 routing information.

This section describes load balancing support on link bundles.

For more information about other forms of load balancing on the Cisco ASR 9000 Series Router, see the following references:

- Per-flow load balancing on non-bundle interfaces using Layer 3 and 4 routing information— See the Cisco ASR 9000 Series Aggregation Services Router IP Addresses and Services Configuration Guide.
- Pseudowire (PW) Load Balancing beginning in Cisco IOS XR 4.0.1—See the Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Configuration Guide.

Layer 2 Ingress Load Balancing on Link Bundles

By default, load balancing on Layer 2 link bundles is done based on the MAC source and destination address (SA/DA) fields in the incoming packet header. Table 1 shows a summary of the parameters used for load balancing of incoming traffic at Layer 2 based on whether the default mode, EFP-based, or flow-based load balancing is in use.

Per-flow load balancing is supported on all links in the bundle. This scheme achieves load sharing by allowing the router to distribute packets over one of the links in the bundle, that is determined through a hash calculation. The hash calculation is an algorithm for link selection based on certain parameters.

The standard hash calculation is a 5-tuple hashing, using the following parameters:

- IP source address
- · IP destination address
- Router ID
- Layer 4 source port
- Layer 4 destination port

When per-flow load balancing is enabled, all packets for a certain source-destination pair will go through the same link, though there are multiple links available. Per-flow load balancing ensures that packets for a certain source-destination pair arrive in order.



Note Load balancing for multicast traffic applies only when outgoing interfaces are link bundle interfaces or subinterfaces.

Ingress Unicast, Flood, or Multicast Traffic	Parameters	Configuration
Default	Source MAC address Destination MAC address	n/a
EFP-based auto mode	XID of the xconnect	Auto mode is enabled using the bundle load-balancing hash auto command.
EFP-based with user hash	User hash	A user hash is configured using the bundl load-balancing <i>hash-value</i> command.
Flow-based with IP source and destination	Source IP addressDestination IP address	Enabled using the L2VPN load-balancing src-dst-ip command.
Flow-based with MAC source and destination	Source MAC address Destination MAC address	Enabled using the L2VPN load-balancing src-dst-mac command.

Table 13: Bundle Load Balancing for Incoming Traffic

Layer 3 Egress Load Balancing on Link Bundles

Layer 3 load balancing support began on the Cisco ASR 9000 Series Router in Cisco IOS XR 3.9.1, with changes introduced in Cisco IOS XR Release 4.0.1.

Layer 3 Load Balancing Before Cisco IOS XR Release 4.0.1

In Cisco IOS XR 3.9.1 through Cisco IOS XR 4.0, Layer 3 load balancing for link bundles is done on Ethernet Flow Points (EFPs) and is based on the IPv4 source and destination addresses in the packet. When Layer 3 service-specific load balancing is configured, all egressing bundles are load balanced based on the IPv4 source and destination addresses. When packets do not have IPv4 addresses, default load-balancing is used.

Layer 3 load balancing for link bundles is enabled globally, using the following command:

hw-module load-balance bundle l2-service l3-params

Layer 3 Load Balancing From Cisco IOS XR Release 4.0.1

Layer 3 load balancing for link bundles is done when outgoing interfaces are either bundles or bundle subinterfaces. 5-tuple hashing is used for load balancing among bundle member links, using the following parameters:

- · IP source address
- · IP destination address
- Router ID
- Layer 4 source port
- Layer 4 destination port

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.



Note In the case of in-transit IP-in-IP GRE traffic, prior to Release 5.3.1, third-generation line cards used only the outer IP header for hashing. Starting with Release 5.3.1, these line cards use both the inner and outer IP headers for hashing. In contrast, fourth and fifth-generation line cards consistently use only the outer IP header for hashing.

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.

Dynamic Load Balancing for LAG

Beginning in Cisco IOS XR Release 4.0.1, the Cisco ASR 9000 Series Router supports a method of dynamic load balancing among link aggregation (LAG) members. With dynamic load balancing, the hash algorithms for link selection include up to a maximum of 64 links, and are based on the current number of active members in the bundle.

QoS and Link Bundling

On the Cisco ASR 9000 Series Router, when QoS is applied 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 Cisco ASR 9000 Series Router, refer to the *Cisco ASR 9000 Series Aggregation Services Router Modular Quality of Service Configuration Guide* and the *Cisco ASR 9000 Series Aggregation Services Router Modular Quality of Service Command Reference*.

VLANs on an Ethernet Link Bundle

802.1Q VLAN subinterfaces can be configured on 802.3ad Ethernet link bundles. Keep the following information in mind when adding VLANs on an Ethernet link bundle:

- The maximum number of VLANs allowed per bundle is 4096.
- The maximum number of bundled VLANs allowed per router is 16384.



Note

The memory requirement for bundle VLANs is slightly higher than standard physical interfaces.

To create a VLAN subinterface on a bundle, include the VLAN subinterface instance with the **interface Bundle-Ether** command, as follows:

interface Bundle-Ether interface-bundle-id.subinterface

After you create a VLAN on an Ethernet link bundle, all VLAN subinterface configuration is supported on that link bundle.

VLAN subinterfaces can support multiple Layer 2 frame types and services, such as Ethernet Flow Points - EFPs) and Layer 3 services.

Layer 2 EFPs are configured as follows:

interface bundle-ether instance.subinterface l2transport. encapsulation dotlq xxxxx

Layer 3 VLAN subinterfaces are configured as follows:

interface bundle-ether instance.subinterface, encapsulation dot1q xxxxx



Note

The difference between the Layer 2 and Layer 3 interfaces is the l2transport keyword. Both types of interfaces use dot1q encapsulation.

Link Bundle Configuration Overview

The following steps provide a general overview of the link bundle configuration process. Keep in mind that a link must be cleared of all previous network layer configuration before it can be added to a bundle:

- 1. In global configuration mode, create a link bundle. To create an Ethernet link bundle, enter the **interface Bundle-Ether** command.
- 2. Assign an IP address and subnet mask to the virtual interface using the **ipv4 address** command.
- **3.** Add interfaces to the bundle you created in Step 1 with the **bundle id** command in the interface configuration submode.

You can add up to 64 links to a single bundle.


Note

A link is configured 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 failover between active and standby paired RSP 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 RSP fails, the standby RSP becomes operational. The configuration, node state, and checkpoint data of the failed RSP are replicated to the standby RSP. The bundled interfaces will all be present when the standby RSP becomes the active RSP.



Note

Failover is always onto the standby RSP.

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

Link Failover

When one member link in a bundle fails, traffic is redirected to the remaining operational member links and traffic flow remains uninterrupted.

Multi-Gigabit Service Control Point

Multi-Gigabit Service Control Point (MGSCP) is a deployment model that uses certain link bundling and forwarding features on the Cisco ASR 9000 Series Aggregation Services Routers to support load balancing, clustering, and redundancy for broadband subscriber traffic on Cisco Service Control Engine (SCE) devices.

The Cisco SCE platform is used to provide many services such as user authorization, reporting, and application bandwidth metering for broadband subscribers. It manages IP traffic using a stateful processing mechanism based on application and subscriber awareness. Maintaining this statefulness requires that the SCE platform captures both the upstream and downstream flows of a session to classify it and provide Layer 7 processing at the application level.

To process an application that is implemented with a bundle of flows, such as FTP or Session Initiation Protocol (SIP), the SCE platform needs to process all the flows that comprise a session of this application. In addition, when the SCE platform is configured to implement per subscriber reporting or control (sometimes referred to as *subscriber awareness*), it must process all traffic flows that a given subscriber generates.

Because of this stateful processing to the subscriber level, the SCE platform is implemented in a network with a "bump-in-the-wire" topology for Layer 2 and Layer 3 transparency. However, as the number of broadband subscribers increases along with the bandwidth that an SCE platform must support, scaling the solution presents certain challenges when inserted into a typical network environment where asymmetric routing is often implemented and the two directions of a single session, or the many flows of a specific subscriber, could be split between different links.

The MGSCP solution on the Cisco ASR 9000 Series Router satisfies these requirements by providing a topology to scale multiple SCE devices in a cluster that are connected to the router using link bundling, where

all subscriber traffic can be directed through the same bundle member link. In addition, MGSCP also provides the benefits of load balancing and redundancy.

This figure shows a basic network topology for MGSCP with a Cisco ASR 9000 Series Router connected between the subscriber and core networks, and acting as a dispatcher for the attached SCE cluster. The N+1 notation indicates one backup (or protect) link for the other active links on either side of the SCEs.

Figure 28: Basic MGSCP Network Topology

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.

YANG Data Model

You can programmatically perform the configuration using <code>openconfig-lacp.yang</code> OpenConfig data model or <code>Cisco-IOS-XR-um-lacp-cfg.yang</code> Unified data model. To get started with using data models, see the *Programmability Configuration Guide for Cisco ASR 9000 Series Routers*.

- 1. configure
- 2. interface Bundle-Ether bundle-id
- **3.** ipv4 address ipv4-address mask
- 4. bundle minimum-active bandwidth *kbps*
- 5. bundle minimum-active links links
- 6. bundle maximum-active links [hot-standby]
- 7. lacp fast-switchover
- 8. exit
- 9. interface {GigabitEthernet | TenGigE} interface-path-id
- **10.** bundle id *bundle-id* [mode {active | on | passive}]
- **11. bundle port-priority** priority
- 12. no shutdown
- 13. exit
- 14. bundle id *bundle-id* [mode {active | passive | on}] no shutdown exit
- 15. end or commit
- **16.** exit
- 17. exit

- **18.** Perform Step 1 through Step 15 on the remote end of the connection.
- **19.** show bundle Bundle-Ether bundle-id
- 20. show lacp bundle Bundle-Ether bundle-id

DETAILED STEPS

Procedure

I

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	interface Bundle-Ether <i>bundle-id</i> Example:	Creates a new Ethernet link bundle with the specified bundle-id. The range is 1 to 65535.
	RP/0/RSP0/CPU0:router#(config)# interface Bundle-Ether 3	the interface configuration submode, where you can enter interface specific configuration commands are entered. 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:	Assigns an IP address and subnet mask to the virtual interface using the ipv4 address configuration subcommand.
	<pre>RP/0/RSP0/CPU0:router(config-if)# ipv4 address 10.1.2.3 255.0.0.0</pre>	Note • On the Cisco ASR 9000 Series Router, only a Layer 3 bundle interface requires an IP address.
Step 4	bundle minimum-active bandwidth <i>kbps</i>	(Optional) Sets the minimum amount of bandwidth
	Example:	required before a user can bring up a bundle.
	RP/0/RSP0/CPU0:router(config-if)# bundle minimum-active bandwidth 580000	
Step 5	bundle minimum-active links links	(Optional) Sets the number of active links required before
	Example:	you can bring up a specific bundle.
	<pre>RP/0/RP0/CPU0:router(config-if)# bundle minimum-active links 2</pre>	
Step 6	bundle maximum-active links links [hot-standby]	(Optional) Implements 1:1 link protection for the bundle,
	Example:	which causes the highest-priority link in the bundle to become active and the second-highest-priority link to
	RP/0/RSP0/CPU0:router(config-if)# bundle maximum-active links 1 hot-standby	become the standby. Also, specifies that a switchover between active and standby LACP-enabled links is implemented per a proprietary optimization.

	Command or Action	Purpose
		 Note The priority of the active and standby links is based on the value of the bundle port-priority command.
Step 7	<pre>lacp fast-switchover Example: RP/0/RSP0/CPU0:router(config-if)# lacp fast-switchover</pre>	(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:	Exits interface configuration submode for the Ethernet link bundle.
	RP/0/RSP0/CPU0:router(config-if)# exit	
Step 9	interface {GigabitEthernet TenGigE} interface-path-id	Enters interface configuration mode for the specified interface.
	RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 1/0/0/0	Enter the GigabitEthernet or TenGigE keyword to specify the interface type. Replace the <i>interface-path-id</i> argument with the node-id in the <i>rack/slot/module</i> format.
Step 10	bundle id <i>bundle-id</i> [mode {active on passive}]	Adds the link to the specified bundle.
	Example: RP/0/RSP0/CPU0:router(config-if)# bundle-id 3	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	<pre>bundle port-priority priority Example: RP/0/RSP0/CPU0:router(config-if)# bundle port-priority 1</pre>	(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:	(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.
	<pre>RP/0/RSP0/CPU0:router(config-if)# no shutdown</pre>	

	Command or Action	Purpose
Step 13	exit	Exits interface configuration submode for the Ethernet
	Example:	interface.
	RP/0/RSP0/CPU0:router(config-if)# exit	
Step 14	bundle id <i>bundle-id</i> [mode {active passive on}] no shutdown exit	(Optional) Repeat Step 8 through Step 11 to add more links to the bundle.
	Example:	
	<pre>RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 1/0/2/1</pre>	
	RP/0/RSP0/CPU0:router(config-if)# bundle id 3	
	RP/0/RSP0/CPU0:router(config-if)# bundle port-priority 2	
	RP/0/RSP0/CPU0:router(config-if)# no shutdown	
	RP/0/RSP0/CPU0:router(config-if)# exit	
	RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 1/0/2/3	
	RP/0/RSP0/CPU0:router(config-if)# bundle id 3	
	RP/0/RSP0/CPU0:router(config-if)# no shutdown	
	<pre>RP/0/RSP0/CPU0:router(config-if)# exit</pre>	
Step 15	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	RP/0/RSP0/CPU0:router(config-if)# end	Upcommitted shares found commit them before
	or	<pre>exiting(yes/no/cancel)? [cancel]:</pre>
	RP/0/RSP0/CPU0:router(config-if)# commit	• 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.

	Command or Action	Purpose
Step 16	exit	Exits interface configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router(config-if)# exit	
Step 17	exit	Exits global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router(config)# exit	
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	(Optional) Shows information about the specified Ethernet
	Example:	link bundle.
	<pre>RP/0/RSP0/CPU0:router# show bundle Bundle-Ether 3</pre>	
Step 20 show law Example	show lacp bundle Bundle-Ether bundle-id	(Optional) Shows detailed information about LACP ports
	Example:	and their peers.
	RP/0/RSP0/CPU0:router# show lacp bundle Bundle-Ether 3	

Configuring EFP Load Balancing on an Ethernet Link Bundle

This section describes how to configure Ethernet flow point (EFP) Load Balancing on an Ethernet link bundle.

By default, Ethernet flow point (EFP) load balancing is enabled. However, the user can choose to configure all egressing traffic on the fixed members of a bundle to flow through the same physical member link. This configuration is available only on an Ethernet Bundle subinterface with Layer 2 transport (**l2transport**) enabled.



Note If the active members of the bundle change, the traffic for the bundle may get mapped to a different physical link that has a hash value that matches the configured value.

- 1. configure
- 2. hw-module load-balance bundle l2-service l3-params
- 3. interface Bundle-Ether bundle-id l2transport
- 4. bundle load-balance hash hash-value [auto]
- 5. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	hw-module load-balance bundle l2-service l3-params	(Optional) Enables Layer 3 load balancing on Layer 2 link
	Example:	bundles.
	RP/0/RSP0/CPU0:router(config)# hw-module load-balance bundle 12-service 13-params	
Step 3	interface Bundle-Ether bundle-id l2transport	Creates a new Ethernet link bundle with the specified
	Example:	<i>bundle-id</i> and with Layer 2 transport enabled.
	RP/0/RSP0/CPU0:router#(config)# interface	The range is 1 to 65535.
	Bundle-Ether 3 12transport	
Step 4	bundle load-balance hash hash-value [auto]	Configures all egressing traffic on the fixed members of a bundle to flow through the same physical member link
	Example:	• hash value Numeric value that specifies the physical
	RP/0/RSP0# bundle load-balancing hash 1	member link through which all egressing traffic in this
	or	bundle will flow. The values are 1 through 8.
	RP/0/RSP0# bundle load-balancing hash auto	• auto —The physical member link through which all egressing traffic on this bundle will flow is automatically chosen.
Step 5	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts
	RP/0/RSP0/CPU0:router(config-if)# end	you to commit changes: Uncommitted changes found, commit them before exiting (yes/no/cancel)?
	or	• Entering yes saves configuration changes to the
	RP/0/RSP0/CPU0:router(config-if)# commit	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
	• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring VLAN Bundles

This section describes how to configure a VLAN bundle. The creation of a VLAN bundle involves three main tasks:

SUMMARY STEPS

- **1.** Create an Ethernet bundle.
- 2. Create VLAN subinterfaces and assign them to the Ethernet bundle.
- **3.** Assign Ethernet links to the Ethernet bundle.

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	Create an Ethernet bundle.	
Step 2	Create VLAN subinterfaces and assign them to the Ethernet bundle.	
Step 3	Assign Ethernet links to the Ethernet bundle.	

These tasks are describe in detail in the procedure that follows.

Note

In order for a VLAN bundle to be active, you must perform the same configuration on both ends of the bundle connection.

- 1. configure
- 2. interface Bundle-Ether bundle-id
- 3. ipv4 address ipv4-address mask
- 4. bundle minimum-active bandwidth kbps
- 5. bundle minimum-active links *links*
- 6. bundle maximum-active links [hot-standby]
- 7. exit
- 8. interface Bundle-Ether bundle-id.vlan-id
- 9. encapsulation dot1q
- 10. ipv4 address ipv4-address mask

- 11. no shutdown
- **12.** exit
- **13.** Repeat Step 9 through Step 12 to add more VLANS to the bundle you created in Step 2.
- 14. end or commit
- 15. exit
- **16**. exit
- **17.** configure
- **18.** interface {GigabitEthernet | TenGigE}*interface-path-id*
- **19**. lacp fast-switchover

DETAILED STEPS

Procedure

I

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	interface Bundle-Ether bundle-id	Creates and names a new Ethernet link bundle.
	<pre>Example: RP/0/RSP0/CPU0:router#(config)# interface</pre>	This interface Bundle-Ether command enters you into the interface configuration submode, where you can enter interface-specific configuration commands. Use the exit
	Bundle-Ether 3	command to exit from the interface configuration submode back to the normal global configuration mode.
Step 3	ipv4 address ipv4-address mask	Assigns an IP address and subnet mask to the virtual
	Example:	interface using the ipv4 address configuration subcommand.
	<pre>RP/0/RSP0/CPU0:router(config-if)# ipv4 address 10.1.2.3 255.0.0.0</pre>	
Step 4	bundle minimum-active bandwidth kbpsExample:	(Optional) Sets the minimum amount of bandwidth required before a user can bring up a bundle.
	<pre>RP/0/RSP0/CPU0:router(config-if) # bundle minimum-active bandwidth 580000</pre>	
Step 5	bundle minimum-active links links	(Optional) Sets the number of active links required before
	Example:	you can bring up a specific bundle.
	<pre>RP/0/RSP0/CPU0:router(config-if)# bundle minimum-active links 2</pre>	
Step 6	bundle maximum-active links <i>links</i> [hot-standby] Example:	(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

I

	Command or Action	Purpose
	RP/0/RSP0/CPU0:router(config-if)# bundle maximum-active links 1 hot-standby	become the standby. Also, specifies that a switchover between active and standby LACP-enabled links is implemented per a proprietary optimization.
		Note The priority of the active and standby links is based on the value of the bundle port-priority command.
Step 7	exit	Exits the interface configuration submode.
	Example:	
	RP/0/RSP0/CPU0:router(config-if)# exit	
Step 8	interface Bundle-Ether bundle-id.vlan-id	Creates a new VLAN, and assigns the VLAN to the Ethernet bundle you created in Step 2.
	RP/0/RSP0/CPU0:router#(config)# interface	Replace the <i>bundle-id</i> argument with the <i>bundle-id</i> you created in Step 2.
	Bundle-Ether 3.1	Replace the <i>vlan-id</i> with a subinterface identifier. Range is from 1 to 4094 inclusive (0 and 4095 are reserved).
		Note When you include the <i>.vlan-id</i> argument with the interface Bundle-Ether <i>bundle-id</i> command, you enter subinterface configuration mode.
Step 9	encapsulation dot1q	Sets the Layer 2 encapsulation of an interface.
	<pre>Example: RP/0/RSP0/CPU0:router(config-subif)# encapsulation dotlq 100, untagged</pre>	Note The dot1q vlan command is replaced by the encapsulation dot1q command on the Cisco ASR 9000 Series Router. It is still available for backward-compatibility, but only for Layer 3 interfaces.
Step 10	ipv4 address ipv4-address mask	Assigns an IP address and subnet mask to the subinterface.
	Example:	
	<pre>RP/0/RSP0/CPU0:router#(config-subif)# ipv4 address 10.1.2.3/24</pre>	
Step 11	no shutdown	(Optional) If a link is in the down state, bring it up. The
	Example:	no shutdown command returns the link to an up or down state depending on the configuration and state of the link.
	RP/0/RSP0/CPU0:router#(config-subif)# no shutdown	
Step 12	exit	Exits subinterface configuration mode for the VLAN
	Example:	subinterface.
	RP/0/RSP0/CPU0:router(config-subif)# exit	

I

	Command or Action	Purpose
Step 13	Repeat Step 9 through Step 12 to add more VLANS to the bundle you created in Step 2.	(Optional) Adds more subinterfaces to the bundle.
Step 14	end or commit	Saves configuration changes.
	Example: BP/0/BSP0/CPU0:router(config=subif)# end	• When you issue the end command, the system prompts you to commit changes:
	or	Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:
	RP/0/RSP0/CPU0:router(config-subif)# commit	- Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		- Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		- Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.
Step 15	exit	Exits interface configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router(config-subif)# end	
Step 16	exit	Exits global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router(config)# exit	
Step 17	configure	Enters global configuration mode.
	Example:	
	RP/0/RP0/CPU0:router # configure	
Step 18	interface {GigabitEthernet TenGigE} interface-path-id	Enters interface configuration mode for the Ethernet
	Example:	interface you want to add to the Bundle.
	RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 1/0/0/0	Enter the GigabitEthernet or TenGigE keyword to specify the interface type. Replace the <i>interface-path-id</i> 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.

	Command or Action	Purpose
Step 19	lacp fast-switchover	(Optional) If you enabled 1:1 link protection (you set the
	<pre>Example: RP/0/RSP0/CPU0:router(config-if)# lacp fast-switchover</pre>	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.

Configuring POS Link Bundles

This section describes how to configure a POS link bundle.



Note

In order for a POS bundle to be active, you must perform the same configuration on both connection endpoints of the POS bundle.

- 1. configure
- 2. interface Bundle-POS bundle-id
- 3. ipv4 address ipv4-address mask
- 4. bundle minimum-active bandwidth *kbps*
- 5. bundle minimum-active links links
- 6. bundle maximum-active links [hot-standby]
- 7. exit
- 8. interface POS interface-path-id
- **9. bundle** *id bundle*-*id*
- **10. bundle port-priority** *priority*
- 11. no shutdown
- **12**. exit
- **13.** Repeat Step 19 through Step 21 to add more links to a bundle
- 14. end or commit
- 15. exit
- 16. exit
- **17.** Perform Step 1 through Step 23 on the remote end of the connection.
- 18. show bundle Bundle-POS number

DETAILED STEPS

Procedure

I

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	interface Bundle-POS bundle-id	Configures and names the new bundled POS interface.
	Example: RP/0/RSP0/CPU0:router#(config)#interface Bundle-POS 2	Enters the interface configuration submode, from where interface specific configuration commands are executed. Use the exit command to exit from the interface configuration submode, and get back to the normal global configuration mode.
Step 3	ipv4 address ipv4-address mask	Assigns an IP address and subnet mask to the virtual
	Example:	interface using the ip address configuration subcommand.
	<pre>RP/0/RSP0/CPU0:router(config-if)# ipv4 address 10.1.2.3 255.0.0.0</pre>	
Step 4	bundle minimum-active bandwidth kbps	(Optional) Sets the minimum amount of bandwidth
	Example:	required before a user can bring up a bundle.
	RP/0/RSP0/CPU0:router(config-if)# bundle minimum-active bandwidth 620000	
Step 5	bundle minimum-active links links	(Optional) Sets the number of active links required before
	Example:	you can bring up a specific bundle.
	RP/0/RSP0/CPU0:router(config-if)# bundle minimum-active links 2	
Step 6	bundle maximum-active links links [hot-standby]	(Optional) Implements 1:1 link protection for the bundle,
	Example:	become active and the second-highest-priority link in the bundle to
	RP/0/RSP0/CPU0:router(config-if)# bundle maximum-active links 1 hot-standby	become the standby. Also, specifies that a switchover between active and standby LACP-enabled links is implemented according to a proprietary optimization.
		 Note The priority of the active and standby links is based on the value of the bundle port-priority command.
Step 7	exit	Exits the interface configuration submode.

I

	Command or Action	Purpose
Step 8	interface POS interface-path-id Example:	Enters POS interface configuration mode and specifies the POS interface name and interface-path-id notation <i>rack/slot/module/port</i> .
	<pre>RP/0/RSP0/CPU0:router(config)# interface POS 0/1/0/0</pre>	
Step 9	bundle id bundle-id	Adds the link to the specified bundle.
	Example: RP/0/RSP0/CPU0:router(config-if)# bundle-id 3	To add the link to the bundle without LACP support, include the optional mode on keywords with the command string.
		 Note If you do not specify the mode keyword, the default mode is on (LACP is not run over the port).
Step 10	<pre>bundle port-priority priority Example: RP/0/RSP0/CPU0:router(config-if)# bundle</pre>	(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
	port-priority 1	the standby link to 2.
Step 11	no shutdown	Removes the shutdown configuration which forces the
	Example:	command then returns the link to an up or down state, depending on the configuration and state of the link.
0: 10	<pre>RP/U/RSPU/CPUU:router(config=11)# no shutdown</pre>	
Step 12	exit	Exits the interface configuration submode for the POS interface.
	Example:	
	RP/0/RSP0/CPU0:router# exit	
Step 13	Repeat Step 19 through Step 21 to add more links to a bundle	(Optional) Adds more links to the bundle you created in Step 2.
Step 14	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	RP/0/RSP0/CPU0:router(config-if)# end	
	or	Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:
	RP/0/RSP0/CPU0:router(config-if)# commit	• 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
		• Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.
Step 15	exit	Exits interface configuration mode.
	Example:	
	<pre>RP/0/RSP0/CPU0:router(config-if)# exit</pre>	
Step 16	exit	Exits global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router(config)# exit	
Step 17	Perform Step 1 through Step 23 on the remote end of the connection.	Brings up the other end of the link bundle.
Step 18	show bundle Bundle-POS number	(Optional) Shows information about the specified POS
	Example:	link bundle.
	RP/0/RSP0/CPU0:router# show bundle Bundle-POS 1	

Configuring Multichassis Link Aggregation

Perform these tasks to configure Multichassis Link Aggregation (MC-LAG):

Configuring Interchassis Communication Protocol

Perform this task to configure Interchassis Communication Protocol (ICCP).

- **1**. configure
- 2. redundancy iccp group group-id
- 3. member neighbor neighbor-ip-address
- 4. backbone interface interface-type-id
- **5.** isolation recovery-delay *delay*
- 6. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	redundancy iccp group group-id	Adds an ICCP redundancy group.
	Example:	
	RP/0/RSP0/CPU0:router#(config-redundancy-iccp-group)# redundancy iccp group 100	•
Step 3	member neighbor neighbor-ip-address	Configures ICCP members.
	Example:	This is the ICCP peer for this redundancy group. Only one
RP/0/RSP0/CPU0:router#(confi member neighbor 10.1.1.1	RP/0/RSP0/CPU0:router#(config-redundancy-iccp-group)# member neighbor 10.1.1.1	address is the LDP router-ID of the neighbor. This configuration is required for ICCP to function.
Step 4	backbone interface interface-type-id	Configures ICCP backbone interfaces.
	Example: RP/0/RSP0/CPU0:router#(config-redundancy-iccp-group)# backbone interface GigabitEthernet0/1/0/2	This is an optional configuration to detect isolation from the network core, and triggers switchover to the peer POA if the POA on which the failure is occurring is active. Multiple backbone interfaces can be configured for each redundancy group. When all backbone interfaces are not UP, this is an indication of core isolation. When one or more backbone interfaces are UP, then the POA is not isolated from the network core. Backbone interfaces are typically the interfaces which L2VPN pseudowires can use.
Step 5	isolation recovery-delay <i>delay</i> Example:	Configures the isolation parameters and specifies delay before clearing isolation condition after recovery from failure.
	RP/0/RSP0/CPU0:router#(config-redundancy-iccp-group)# isolation recovery-delay 30	Isolation recovery delay timer is started once the core isolation condition has cleared. When the timer expires, the POA can take over as the active POA (depending on other conditions like bundle recovery delay timer). This allows:
		• the network core to reconverge after the backbone interfaces have come up
		• ICCP state to be exchanged in order for POAs to know what state they are supposed to be in so that MCLAG bundles do not flap excessively.
		This is an optional configuration; if not configured, the delay is set to 180 seconds, by default.

	Command or Action	Purpose
Step 6	end or commit	Saves configuration changes.
	<pre>Example: RP/0/RSP0/CPU0:router(config-redundancy-iccp-group)# or RP/0/RSP0/CPU0:router(config-redundancy-iccp-group)# commit</pre>	 When you issue the end command, the system prompts you to commit changes: Uncommitted changes found, commit them before exiting (yes/no/cancel)? Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring Multichassis Link Aggregation Control Protocol Session

Perform this task to enable a Multichassis Link Aggregation Control Protocol (mLACP) session.

SUMMARY STEPS

- 1. configure
- 2. redundancy iccp group group-id
- 3. mlacp system mac mac-id
- 4. mlacp system priority priority
- 5. mlacp node node-id
- 6. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	redundancy iccp group group-id	Adds an ICCP redundancy group.
	Example:	

_	Command or Action	Purpose
	RP/0/RSP0/CPU0:router#(config-redundancy-iccp-group)# redundancy iccp group 100	
Step 3	<pre>mlacp system mac mac-id Example: RP/0/RSP0/CPU0:router#(config-redundancy-iccp-group)# mlacp system mac 1.1.1</pre>	 Configures the LACP system ID to be used in this ICCP Group. Note The <i>mac-id</i> is a user configured value for the LACP system LAG-ID to be used by the POAs. It is highly recommended that the <i>mac-ids</i> have the same value on both POAs. You can have different LAG-IDs for different groups.
Step 4	<pre>mlacp system priority priority Example: RP/0/RSP0/CPU0:router#(config-redundancy-iccp-group)# mlacp system priority 10</pre>	 Sets the LACP system priority to be used in this ICCP Group. Note It is recommended that system priority of the POAs be configured to a lower numerical value (higher priority) than the LACP LAG ID of the DHD. If the DHD has higher system priority then dynamic priority management cannot work and brute force switchover is automatically used.
Step 5	<pre>mlacp node node-id Example: RP/0/RSP0/CPU0:router#(config-redundancy-iccp-group)# mlacp node 1</pre>	 Sets the LACP system priority to be used in this ICCP Group. Note The <i>node-id</i> must be unique for each POA.
Step 6	<pre>end or commit Example: RP/0/RSP0/CPU0:router(config-if)# end or RP/0/RSP0/CPU0:router(config-if)# commit</pre>	 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.

 Command or Action	Purpose
	• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring Multichassis Link Aggregation Control Protocol Bundle

Perform this task to configure a Multichassis Link Aggregation Control Protocol (mLACP) bundle.

SUMMARY STEPS

- 1. configure
- 2. interface Bundle-Ether bundle-id
- 3. mac-address mac-id
- 4. bundle wait-while milliseconds
- 5. lacp switchover suppress-flaps milliseconds
- 6. mlacp iccp-group group-id
- 7. mlacp port-priority priority
- 8. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	interface Bundle-Ether bundle-id	Creates and names a new Ethernet link bundle.
	Example:	
	<pre>RP/0/RSP0/CPU0:router#(config)# interface Bundle-Ether 3</pre>	
Step 3	mac-address mac-id	Sets the MAC address on the interface.
	Example:	Note
	<pre>RP/0/RSP0/CPU0:router#(config-if)# mac-address 1.1.1</pre>	• Configuring the same MAC address on both POAs is highly recommended.
Step 4	bundle wait-while milliseconds	Sets the wait-while timeout for members of this bundle.
	Example:	

	Command or Action	Purpose
	RP/0/RSP0/CPU0:router#(config-if)# bundle wait-while 100	
Step 5	<pre>lacp switchover suppress-flaps milliseconds Example: RP/0/RSP0/CPU0:router#(config-if)# lacp switchover suppress-flaps 300</pre>	 Sets the time for which to suppress flaps during a LACP switchover. Note It is recommended that the value used for the <i>milliseconds</i> argument is greater than that for the wait-while timer of the local device (and DHD).
Step 6	<pre>mlacp iccp-group group-id Example: RP/0/RSP0/CPU0:router#(config-if)# mlacp iccp-group 10</pre>	Configures the ICCP redundancy group in which this bundle should operate.
Step 7	<pre>mlacp port-priority priority Example: RP/0/RSP0/CPU0:router#(config-if)# mlacp port-priority 10</pre>	 Sets the starting priority for all member links on this device when running mLACP. Note Lower value indicates higher priority. If you are using dynamic priority management the priority of the links change when switchovers occur.
Step 8	<pre>end or commit Example: RP/0/RSP0/CPU0:router(config-if)# end or RP/0/RSP0/CPU0:router(config-if)# commit</pre>	 Saves configuration changes. When you issue the end command, the system prompts you to commit changes: Uncommitted changes found, commit them before exiting (yes/no/cancel)? Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring Dual-Homed Device

Perform this task to configure the dual-homed device (DHD).

Note If an ASR 9000 Series Router is to be used as a DHD, it is recommended that you configure the **bundle maximum-active links** command where *links* is the number of links connecting the DHD to one of the POAs.

SUMMARY STEPS

- **1**. configure
- 2. interface Bundle-Ether bundle-id
- 3. bundle wait-while milliseconds
- 4. lacp switchover suppress-flaps milliseconds
- 5. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	interface Bundle-Ether bundle-id	Creates and names a new Ethernet link bundle.
	Example:	
	<pre>RP/0/RSP0/CPU0:router#(config-if)# interface Bundle-Ether 3</pre>	
Step 3	bundle wait-while milliseconds	Sets the wait-while timeout for members of this bundle.
	Example:	
	RP/0/RSP0/CPU0:router#(config-if)# bundle wait-while 100	
Step 4	lacp switchover suppress-flaps milliseconds	Sets the time for which to suppress flaps during a LACP
	Example:	switchover.
	<pre>RP/0/RSP0/CPU0:router#(config-if)# lacp switchover suppress-flaps 300</pre>	
Step 5	end or commit	Saves configuration changes.
	Example:	

Command or Action	Purpose
RP/0/RSP0/CPU0:router(config-if)# end	• When you issue the end command, the system prompts you to commit changes: Uncommitted changes found, commit them before exiting (yes/no/cancel)?
RP/0/RSP0/CPU0:router(config-if)# commit	 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.
	The members added to the bundle on one POA go <i>Active</i> , and the members on the other POA are in <i>Standby</i> state. This can be verified by using the show bundle command on either POA to display the membership information for correctly configured members on both the POAs:
	RP/0/RSP0/CPU0:router# show bundle
	Bundle-Ether1 Status: Up Local links <active configured="" standby="">: 1 / 0 / 1</active>
	Local bandwidth <effective available="">: 1000000 (1000000) kbps MAC address (source): 0000.deaf.0000</effective>
	<pre>(Configured) Minimum active links / bandwidth: 1 / 1 kbps Maximum active links: 64 Wait while timer: 100 ms LACP: Operational Flap suppression timer: 300 ms mLACP: Operational ICCP Group: 1 Role: Active Foreign links <active configured="">: 0 / 1 Switchover type: Non-revertive Recovery delay: 300 s Maximize threshold: Not configured IPv4 BFD: Not configured</active></pre>
	kbps
	Gi0/0/0/0 Local Active 0x8001, 0x9001 1000000 Link is Active

Command or Action	Purpose
	Gi0/0/0/0 5.4.3.2 Standby 0x8002, 0xa001 1000000 Link is marked as Standby by mLACP peer

Ŵ

Note To switch to an active POA, use the mlacp switchover Bundle-Ether command on the currently active router.

Configuring One-way Pseudowire Redundancy in MC-LAG

Perform this task to allow one-way pseudowire redundancy behavior when the redundancy group is configured.

SUMMARY STEPS

- 1. configure
- 2. l2vpn
- **3. pw-class** {*class-name*}
- 4. encapsulation mpls
- 5. redundancy one-way
- 6. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	l2vpn	Enters L2VPN configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router(config)# l2vpn RP/0/RSP0/CPU0:router(config-l2vpn)#	
Step 3	pw-class {class-name}	Configures the pseudowire class template name to use for
	Example:	the pseudowire.
	RP/0/RSP0/CPU0:router(config-l2vpn)# pw-class class1	
Step 4	encapsulation mpls	Configures the pseudowire encapsulation to MPLS.
	Example:	

	Command or Action	Purpose
	RP/0/RSP0/CPU0:router(config-l2vpn-pwc)# encapsulation mpls	
Step 5	<pre>redundancy one-way Example: RP/0/RSP0/CPU0:router(config=l2vpn-pwc-mpls)#</pre>	Configures one-way PW redundancy behavior. Note • The redundancy one-way command is effective only if the redundancy group is configured.
	redundancy one-way	• The redundancy one-way command is mandatory for one-way pseudowire redundancy in MC-LAG to avoid inconsistent PW states on remote PEs for the MC-LAG point of attachments.
Step 6	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac)# end	
	or	exiting(yes/no/cancel)? [cancel]:
	<pre>RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-mac)# commit</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 VPWS Cross-Connects in MC-LAG

Perform this task to configure VPWS cross-connects in MC-LAG.

- 1. configure
- **2**. l2vpn
- 3. pw-status
- 4. **xconnect group** *group*-name
- 5. p2p xconnect-name

- 6. interface type interface-path-id
- 7. neighbor A.B.C.D pw-id pseudowire-id
- 8. pw-class {class-name}
- 9. backup neighbor A.B.C.D pw-id pseudowire-id
- **10. pw-class** {*class-name*}
- **11.** end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	l2vpn	Enters L2VPN configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router(config)# l2vpn	
Step 3	pw-status	Enables pseudowire status.
	Example:	Note
	RP/0/RSP0/CPU0:router(config-l2vpn)# pw-status	• When the attachment circuit changes redundancy state to Active, Active pw-status is sent over the primary and backup pseudowires.
		When the attachment circuit changes redundancy state to Standby, Standby pw-status is sent over the primary and backup pseudowires.
Step 4	xconnect group group-name	Enters the name of the cross-connect group.
	Example:	
	RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group grp_1	
Step 5	p2p xconnect-name	Enters a name for the point-to-point cross-connect.
	Example:	
	RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p p1	
Step 6	interface type interface-path-id	Specifies the interface type ID.
	Example:	

	Command or Action	Purpose
	RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# interface Bundle-Ether 1.1	
Step 7	neighbor A.B.C.D pw-id pseudowire-id	Configures the pseudowire segment for the cross-connect.
	Example:	Optionally, you can disable the control word or set the
	RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# neighbor 10.2.2.2 pw-id 2000	transport-type to Ethernet or VLAN.
Step 8	pw-class {class-name}	Configures the pseudowire class template name to use for
	Example:	the pseudowire.
	RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# pw-class cl	
Step 9	backup neighbor A.B.C.D pw-id pseudowire-id	Adds a backup pseudowire.
	Example:	
	RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# backup neighbor 10.2.2.2 pw-id 2000	
Step 10	<pre>pw-class {class-name}</pre>	Configures the pseudowire class template name to use for
	Example:	the backup pseudowire.
	RP/0/RSP0/CPU0:router(config-12vpn-xc-p2p-pw-backup)# pw-class c2	+
Step 11	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system
	RP/0/RSP0/CPU0:router(config-12vpn-xc-p2p-pw-backup)#	#
	end or	Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:
	RP/0/RSP0/CPU0:router(config-12vpn-xc-p2p-pw-backup)# commit	 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 ICCP based Service Homing

Perform this task to configure ICCP-SM.

Before you begin

You must have configured ICCP as shown in the procedure Configuring Interchassis Communication Protocol.

SUMMARY STEPS

- 1. configure
- 2. l2vpn
- 3. redundancy iccp group group-id
- 4. multi-homing node-id node-id
- 5. mac-flush *type*
- **6. interface** *type interface-path-id*
- 7. primary vlan {vlan range}
- 8. secondary vlan {vlan range}
- 9. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	l2vpn	Enters L2VPN configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router(config)# 12vpn RP/0/RSP0/CPU0:router(config-12vpn)#	
Step 3	redundancy iccp group group-id	Enables L2VPN redundancy mode and enters redundance
	Example:	configuration submode. Adds an ICCP redundancy group.
	RP/0/RSP0/CPU0:router#(config-l2vpn)# redundancy iccp group 100	
Step 4	multi-homing node-id node-id	Enter the pseudo MLACP node ID. Enables the ICCP based
	Example:	multi-homing service. The node-ID is used for ICCP signaling arbitration.
	RP/0/RSP0/CPU0:router(config-l2vpn-red-grp)# multi-homing node-id 1	

Step 5 mac-flush rype Example: RP/0/RSP0/CFU0:router(config-12vpn-red-grp)# Step 6 interface type interface-path-id Example: Specifies the interface type ID. It can be a physical p name or the main bundle name (sub-port is not allow can be any physical Ethernet or bundle Ethernet interconnecting to a dual homed CE device. NP/0/RSP0/CFU0:router(config-12vpn-red-grp)# Specifies the interface type ID. It can be a physical p name or the main bundle name (sub-port is not allow can be any physical Ethernet or bundle Ethernet interconnecting to a dual homed CE device. NP/0/RSP0/CFU0:router(config-12vpn-red-grp)# Configures the list of VLANs under the main port, w default to active (forwarding) when there are no fault detected. Step 7 primary vlan (vlan range) Configures the list of VLANs under the main port, w default to active (forwarding) when there are no fault detected. Step 8 secondary vlan (vlan range) Configures the list of VLANs under the main port, w default to standby (blocked) when there are no faults detected. NP/0/RSP0/CFU0:router(config-12vpn-reg-grp)# Saves configuration changes. Sweet of comma separated VLAN ranges individual VLANs. Step 8 end or commit Example: Saves configuration changes. Sweet onfiguration changes for commit the performance sestion and regeriting (yee/no/cance) N/0/RSP0/CPU0:router(config-12vpn-bg-bd=vfi-pw)# Saves configuration changes. Shee forman sest		Command or Action	Purpose
BY/0/RSP0/CPU0:router(config-12xpn-red-grp)# mac=flush_stp-tem Specifies the interface type ID. It can be a physical p name or the main bundle name (sub-port is not allow can be any physical Ethernet or bundle Ethernet inter connecting to a dual homed CE device. Step 6 primary vlan (vlan range) Specifies the interface path-id interface mundle-kther 1 Step 7 primary vlan (vlan range) Configures the list of VLANs under the main port, w default to active (forwarding) when there are no fault detected. Step 8 secondary vlan (vlan range) Configures the list of VLANs under the main port, w default to active (forwarding) when there are no fault detected. Step 8 secondary vlan (vlan range) Specify the list of VLANs under the main port, w default to standby (blocked) when there are no faults detected. Step 9 end or commit Saves configuration changes. Saves configuration changes. Fx/0/RSP0/CPU0:router(config-12xpn-bg-bd-vfi-pv)# end or Saves configuration changes. Saves configuration changes to the running configuration changes to the running configuration changes to the running configuration changes. Entering no exits the configuration session and r the router to EXEC mode without committing the configuration changes. Entering no exits the configuration session without exiting or commit the configuration changes. Entering no exits the configuration changes.	Step 5	mac-flush type Example:	Specifies the type of MAC flush, either stp tcn or mvrp (default).
Step 6 interface type interface-path-id Specifies the interface type ID. It can be a physical p name or the main hundle name (sub-port is not allow or and e any physical Ethernet or bundle Ethernet interface bundle-Ether 1 RP/0/RSP0/CPU0:router(config-12vpn-reg-grp)* Specifies the interface type ID. It can be a physical p name or the main hundle name (sub-port is not allow or an baing physical Ethernet or bundle Ethernet interconceting to a dual homed CE device. Step 7 primary vlan (vlan range) Configures the list of VLANs under the main port, w default to active (forwarding) when there are no fault detected. Step 8 secondary vlan (vlan range) Configures the list of VLANs under the main port, w default to standby (blocked) when there are no faults detected. Step 9 end or commit Example: RP/0/RSP0/CPU0:router(config-12vpn-reg-grp)* Saves configuration changes. Step 9 end or commit Example: RP/0/RSP0/CPU0:router(config-12vpn-bg-bd-vfi-pw)* Saves configuration changes. RP/0/RSP0/CPU0:router(config-12vpn-bg-bd-vfi-pw)* Saves configuration changes. • When you issue the end command, the system privou to commit thene force exiting (yes/no/cance end) RP/0/RSP0/CPU0:router(config-12vpn-bg-bd-vfi-pw)* entering yes saves configuration changes to comfiguration changes. P/0/RSP0/CPU0:router(config-12vpn-bg-bd-vfi-pw)* entering yes aves configuration changes. P/0/RSP0/CPU0:router(config-12vpn-bg-bd-vfi-pw)* entering yes aves conf		RP/0/RSP0/CPU0:router(config-l2vpn-red-grp)# mac-flush stp-tcn	
Step 7 primary vlan (vlan range) Configures the list of VLANs under the main port, widefault to active (forwarding) when there are no fault detected. RF/0/RSP0/CPU0:router(config-12vpn-reg-grp)# Specify the list of of comma separated VLAN ranges individual VLANs. Step 8 secondary vlan (vlan range) Example: RF/0/RSP0/CPU0:router(config-12vpn-reg-grp)# Configures the list of VLANs under the main port, widefault to active (forwarding) when there are no fault detected. Step 8 secondary vlan (vlan range) Example: RF/0/RSP0/CPU0:router(config-12vpn-reg-grp)# Specify the list of VLANs under the main port, widefault to standby (blocked) when there are no faults detected. Step 9 end or commit Specify the list of VLANs under the main port, widefault to standby (blocked) when there are no faults detected. Step 9 end or commit Specify the list of VLANs under the main port, widefault to standby (blocked) when there are no faults detected. Step 9 end or commit Saves configuration changes. RP/0/RSP0/CPU0:router(config-12vpn-bg-bd-vfi-pw)# Saves configuration changes. or running configuration changes to the running configuration changes to the running configuration changes. RP/0/RSP0/CPU0:router(config-12vpn-bg-bd-vfi-pw)# Entering no exits the configuration session and returns the router to EXEC mode. entering no exits the configu	Step 6	<pre>interface type interface-path-id Example: RP/0/RSP0/CPU0:router(config-l2vpn-reg-grp)# interface Bundle-Ether 1</pre>	Specifies the interface type ID. It can be a physical port name or the main bundle name (sub-port is not allowed). It can be any physical Ethernet or bundle Ethernet interface connecting to a dual homed CE device. Only bundle-ethernet main ports are allowed for ICCP-SM. If you want to use this feature on a single ethernet link, then you must configure a bundle with that link.
RF/0/RSP0/CPU0:router(config-l2vpn-reg-grp)# Specify the list of of comma separated VLAN ranges individual VLANs. Step 8 secondary vlan (vlan range) Configures the list of VLANs under the main port, we default to standby (blocked) when there are no faults detected. RP/0/RSP0/CPU0:router(config-l2vpn-reg-grp)# Specify the list of of comma separated VLAN ranges individual VLANs. Step 9 end or commit Saves configuration changes. Example: • When you issue the end command, the system proyou to commit changes: Uncommit the mean of the system proyou to commit changes: Uncommit the mean of the system proyou to commit them before exiting (yes/no/cance) or • RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# • Entering nexis the configuration session and re the router to EXEC mode. RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# • Entering nexis the configuration session and re the router to EXEC mode. RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# • Entering nexis the configuration session and re the router to EXEC mode. Picture of the commit • Entering nexis the configuration session and re the router to EXEC mode. • Entering nexis the configuration session without exiting or commit the configuration changes. • Entering nexis the configuration session and re the router to EXEC mode. • Entering cancel leaves the router in the current configuration session without exiting or comm	Step 7	primary vlan {vlan range} Example:	Configures the list of VLANs under the main port, which default to active (forwarding) when there are no faults detected.
Step 8 secondary vlan (vlan range) Example: Rp/0/RSP0/CPU0:router(config-l2vpn-reg-grp)# Step 9 end or commit Example: Saves configuration changes. P/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# Saves configuration changes. P/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# Saves configuration changes. P/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# Saves configuration changes. RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# Saves configuration changes. RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# Saves configuration changes. RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# Saves configuration changes. Bref 9 Entering ves saves configuration session and returns the router to EXEC mode. RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# Saves configuration session and returns the configuration session and returns the router to EXEC mode. Entering no exits the configuration session and return the configuration session without exiting or commit the configuration session without exiting or commit the configuration session without exiting or commit the configuration session.		RP/0/RSP0/CPU0:router(config-l2vpn-reg-grp)# primary vlan 1-10	Specify the list of of comma separated VLAN ranges or individual VLANs.
RP/0/RSP0/CPU0:router(config-12vpn-reg-grp)# secondary vlan 11-20 Specify the list of of comma separated VLAN ranges individual VLANs. Step 9 end or commit Saves configuration changes. Example: ned RP/0/RSP0/CPU0:router(config-12vpn-bg-bd-vfi-pw)# end Saves configuration changes. or RP/0/RSP0/CPU0:router(config-12vpn-bg-bd-vfi-pw)# commit • When you issue the end command, the system proyou to commit them before exiting (yes/no/cance) RP/0/RSP0/CPU0:router(config-12vpn-bg-bd-vfi-pw)# commit • When you issue the 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 return to EXEC mode without committing the configuration changes. • Entering cancel leaves the router in the current configuration changes. • Use the commit command to save the configuration file and re within the configuration session.	Step 8	secondary vlan {vlan range} Example:	Configures the list of VLANs under the main port, which default to standby (blocked) when there are no faults detected.
Step 9 end or commit Example: RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# or • When you issue the end command, the system proyou to commit changes: Uncommitted changes from commit them before exiting (yes/no/cance) or • Entering yes saves configuration changes to the running configuration file, exits the configuration session and returns the router to EXEC mode. RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# • Entering no exits the configuration session and returns the router to EXEC mode. RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# • Entering no 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 commit the configuration session. • Use the commit command to save the configuration file and rewithin the configuration session.		RP/0/RSP0/CPU0:router(config-l2vpn-reg-grp)# secondary vlan 11-20	Specify the list of of comma separated VLAN ranges or individual VLANs.
Example: RP/0/RSP0/CPU0:router(config-12vpn-bg-bd-vfi-pw)# end• When you issue the end command, the system pro- you to commit changes: Uncommitted changes for commit them before exiting (yes/no/cance) • Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.RP/0/RSP0/CPU0:router(config-12vpn-bg-bd-vfi-pw)# commit• Entering no exits the configuration session and returns the router to EXEC mode.Entering no exits the configuration session and returns the router to EXEC mode.• Entering no exits the configuration session and returns the router to EXEC mode.Entering no exits the configuration changes.• Entering cancel leaves the router in the current configuration changes.• Use the commit command to save the configuration file and re within the configuration session.	Step 9	end or commit	Saves configuration changes.
Note		Example: RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# end Or RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# commit	 When you issue the end command, the system prompts you to commit changes: Uncommitted changes found, commit them before exiting (yes/no/cancel)? Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. Use the commit command to save the configuration changes to the running configuration session.

 Command or Action	Purpose
	You can use the show iccp group , show l2vpn iccp-sm and show lacp bundle-ether commands to monitor ICCP-SM.

Configuring VPLS in MC-LAG

Perform this task to configure VPLS in MC-LAG.

SUMMARY STEPS

- 1. configure
- **2**. l2vpn
- **3**. pw-status
- 4. bridge group bridge-group-name
- 5. bridge-domain bridge-domain-name
- 6. interface type interface-path-id
- **7. vfi** {*vfi-name*}
- 8. neighbor A.B.C.D pw-id pseudowire-id
- **9. pw-class** {*class-name*}
- 10. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	l2vpn	Enters L2VPN configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router(config)# l2vpn	
Step 3	pw-status	(Optional) Enables pseudowire status.
	Example:	All the pseudowires in the VFI are always active, independent of the attachment circuit redundancy state.
	<pre>RP/0/RSP0/CPU0:router(config-l2vpn)# pw-status</pre>	

I

	Command or Action	Purpose
Step 4	bridge group bridge-group-name Example:	Creates a bridge group so that it can contain bridge domains and then assigns network interfaces to the bridge domain.
	<pre>RP/0/RSP0/CPU0:router(config-l2vpn)# bridge group csco RP/0/RSP0/CPU0:router(config-l2vpn-bg)#</pre>	
Step 5	bridge-domain bridge-domain-name Example:	Establishes a bridge domain and enters L2VPN bridge group bridge domain configuration mode.
	<pre>RP/0/RSP0/CPU0:router(config-l2vpn-bg)# bridge-domain abc RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)#</pre>	
Step 6	<pre>interface type interface-path-id Example: RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd)# interface Bundle-Ether 1.1</pre>	Specifies the interface type ID.
Step 7	<pre>vfi {vfi-name} Example: RP/0/RSP0/CPU0:router(config=l2vpn-bg-bd-ac) # vfi vfi-east</pre>	Enters virtual forwarding instance (VFI) configuration mode.
Step 8	neighbor A.B.C.D pw-id pseudowire-id	Configures the pseudowire segment for the cross-connect.
	Example: RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi)# neighbor 10.2.2.2 pw-id 2000	Optionally, you can disable the control word or set the transport-type to Ethernet or VLAN.
Step 9	<pre>pw-class {class-name} Example: RP/0/RSP0/CPU0:router(config=l2vpn-bg-bd-vfi-pw)# pw-class canada</pre>	Configures the pseudowire class template name to use for the pseudowire.
Step 10	end or commit	Saves configuration changes.
	Example: RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# end Or RP/0/RSP0/CPU0:router(config-l2vpn-bg-bd-vfi-pw)# 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

Command or Action	Purpose
	- 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.

How to Configure MGSCP

Perform these tasks to configure MGSCP.

Prerequisites for Configuring MGSCP

Before configuring MGSCP, be sure that the following prerequisites are met:

- You have Gigabit Ethernet or 10-Gigabit Ethernet line cards installed in the Cisco ASR 9000 Series Router.
- You understand how to configure your cluster of Service Control Engine (SCE) devices and configure them according to the desired requirements of your network, including the following requirements for MGSCP support:
 - When you connect the SCE devices to the Cisco ASR 9000 Series Router, be sure that each SCE device has two separate physical links connecting to two different bundle interfaces on the Cisco ASR 9000 Series Router as follows:

— One link from each SCE device is connected to a link on the bundle interface that is routed to the access (or subscriber) side of the network.

— The second link from each SCE device is connected to a link on another bundle interface that is routed to the core side of the network.

• On the SCE device, you configure the SCE ports for link failure reflection (using the **link failure-reflection** command) to ensure that if a link on one side of the SCE goes down, then the link on the other side is automatically shut down. For more information, see the "Configuring the Connection" chapter in the Cisco SCE software configuration guide for your device and release at:

http://www.cisco.com/en/US/products/ps6134/products_installation_and_configuration_guides_list.html

- For your bundle configuration on the Cisco ASR 9000 Series Router, determine the following information:
 - The maximum number of active links that you will support.
 - The bundle links that will be protect (backup) links. You can configure a maximum of 4 protect links.

 To maintain the statefulness of the connected SCEs, all subscriber flows must pass through the same SCE. Therefore, before you configure MGSCP, you need to determine how you want to configure the router to redirect subscriber traffic to ensure that it passes through the appropriate bundle interfaces connected to that SCE.

You can use one of the following methods:

- ACL-Based Forwarding (ABF)—Supports only IP addresses for the next hop, and can be complex to configure. For more information about ABF, see the "Implementing Access Lists and Prefix Lists" chapter of the Cisco ASR 9000 Series Aggregation Services Router IP Addresses and Services Configuration Guide.
- Virtual Routing and Forwarding (VRF)—Recommended. Uses VRF instances for the access and network bundles, which can then be routed using static or dynamic routing with OSPF and BGP.

Restrictions for Configuring MGSCP

Before configuring MGSCP, consider these restrictions:

- You can configure up to a maximum of 4 protect links on a bundle.
- You can configure up to a maximum of 8 member links on a bundle.
- Packets received on the ingress interface must not be tagged with MPLS for MGSCP to effectively do load balancing at the egress interface.

Configuring the Access Bundle for the Subscriber-Facing Side

The configuration of the access bundle facing the subscriber side of the network is similar to the core bundle configuration, with the following guidelines:

- If using VRFs to route subscriber traffic on the same SCE to the bundle (recommended), then a separate VRF is used for the subscriber-facing side.
- Link-order signaling is required to enable LACP processing of link ordering numbers (LONs) for load balancing tables.
- Bundle load balancing is configured based on source IP address.
- The maximum number of active links must be configured to match the maximum number of active links on the core bundle.

- 1. configure
- 2. interface Bundle-Ether bundle-id
- 3. vrf vrf-name
- 4. ipv4 address ipv4-address mask
- 5. lacp cisco enable link-order signaled
- 6. bundle load-balancing hash src-ip
- 7. bundle maximum-active links [hot-standby]
- 8. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	interface Bundle-Ether bundle-id	Specifies or creates an Ethernet bundle interface for the
	Example:	subscriber-facing side of the network, where <i>bundle-id</i> is a number from 1 to 65535, and enters interface
	RP/0/RSP0/CPU0:router(config)# interface Bundle-Ether 100	configuration mode.
Step 3	vrf vrf-name	(Optional-Recommended) Specifies the VRF instance for
	Example:	the subscriber-facing side of the network in which this Ethernet bundle participates.
	<pre>RP/0/RSP0/CPU0:router(config-if)# vrf access</pre>	
Step 4	ipv4 address ipv4-address mask	Specifies an IPv4 address and mask that is part of the
	Example:	32-bit IP address with corresponding mask in dotted-decimal
	RP/0/RSP0/CPU0:router(config-if)# ipv4 address 10.1.1.1 255.255.255.0	format (A.B.C.D).
		 Note This command must be specified after the vrf command to be sure that the IP address is part of the VRF instance.
Step 5	lacp cisco enable link-order signaled	Enables the use of Cisco TLVs to include link order
	Example:	numbering as part of the LACP processing on this bundle.
	RP/0/RSP0/CPU0:router(config-if)# lacp cisco enable link-order signaled	
Step 6	bundle load-balancing hash src-ip	Specifies that the hash used for load balancing on the
	Example:	subscriber bundle interface is based on source IP address.
	RP/0/RSP0/CPU0:router(config-if)# bundle load-balancing hash src-ip	
Step 7	bundle maximum-active links <i>links</i> [hot-standby]	Specifies the maximum number of active links allowed for
	Example:	numbers in use for load balancing tables.
	<pre>RP/0/RSP0/CPU0:router(config-if)# bundle maximum-active links 2</pre>	Note To support MGSCP, this command must also be configured with the same value on the core bundle.

	Command or Action	Purpose
Step 8	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	RP/0/RSP0/CPU0:router (config-bfd-if) # end	
	or	Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:
	RP/0/RSP0/CPU0:router(config-bfd-if)# commit	- Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		- Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		- Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring the Network Bundle for the Core-Facing Side

The configuration of the bundle facing the core side of the network is similar to the access bundle configuration, with the following guidelines:

- If using VRFs to route subscriber traffic on the same SCE to the bundle (recommended), then a separate VRF is used for the core-facing side.
- Link-order signaling is required to enable LACP processing of LONs for load balancing tables.
- Bundle load balancing is configured based on destination IP address.
- The maximum number of active links must be configured to match the maximum number of active links on the access bundle.

- 1. configure
- 2. interface Bundle-Ether bundle-id
- 3. vrf vrf-name
- 4. ipv4 address ipv4-address mask
- 5. lacp cisco enable link-order signaled
- 6. bundle load-balancing hash dst-ip
- 7. bundle maximum-active links links [hot-standby]
- 8. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	interface Bundle-Ether bundle-id	Specifies or creates an Ethernet bundle interface for the subscriber-facing side of the network, where <i>bundle-id</i> is a number from 1 to 65535, and enters interface configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router(config)# interface Bundle-Ether 100	
Step 3	vrf vrf-name	(Optional—Recommended) Specifies the VRF instance for
	Example:	the core-facing side of the network in which this Ethernet bundle participates.
	RP/0/RSP0/CPU0:router(config-if)# vrf access	
Step 4	ipv4 address ipv4-address mask	Specifies an IPv4 address and mask that is part of the
	Example:	specified VRF for this interface, where <i>ipv4-address</i> is the 32-bit IP address with corresponding mask in dotted-decimal format (A.B.C.D).
	RP/0/RSP0/CPU0:router(config-if)# ipv4 address 10.1.1.1 255.255.255.0	
		 Note This command must be specified after the vrf command to be sure that the IP address is part of the VRF instance.
Step 5	lacp cisco enable link-order signaled	Enables the use of Cisco TLVs to include link order
	Example:	numbering as part of the LACP processing on this bundle.
	RP/0/RSP0/CPU0:router(config-if)# lacp cisco enable link-order signaled	
Step 6	bundle load-balancing hash dst-ip	Specifies that the hash used for load balancing on the subscriber bundle interface is based on destination IP address.
	Example:	
	RP/0/RSP0/CPU0:router(config-if)# bundle load-balancing hash dst-ip	
Step 7	bundle maximum-active links links [hot-standby]	Specifies the maximum number of active links allowed for
	Example:	numbers in use for load balancing tables.
	RP/0/RSP0/CPU0:router(config-if)# bundle maximum-active links 2	 Note To support MGSCP, this command must also be configured with the same value on the access bundle.

	Command or Action	Purpose
Step 8	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	RP/0/RSP0/CPU0:router (config-bfd-if) # end	
	or	Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:
	RP/0/RSP0/CPU0:router(config-bfd-if)# commit	- Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		- Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		- Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring the Bundle Member Interfaces

When the access and core bundles have been configured, bundle interfaces must be configured as the active and protect links on those bundles, with the following guidelines:

- A link becomes a bundle member using the **bundle id** command and specifying the ID of the corresponding bundle interface. For MGSCP, there are two different bundles: one for the access side traffic, and one for the core side traffic. These bundles each have a link connecting to either side of an SCE. Be sure to carefully map your interfaces to the appropriate bundle.
- LACP is required for MGSCP, so the link must be configured with mode active on the bundle.
- Active and backup (protect) links are configured using the **bundle port-priority** command:
 - To configure a working (active) link, use a priority of 1. The maximum number of active links that you can configure is determined by the value of the **bundle maximum-active links** command on the bundle.
 - Any priority other than 1 designates the link as a protect link. You can configure a maximum of 4
 protect links.

- 1. configure
- 2. interface [GigabitEthernet | TenGigE] interface-path-id
- 3. bundle id *bundle-id* mode active
- 4. bundle port-priority *priority*
- 5. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	<pre>interface [GigabitEthernet TenGigE] interface-path-id Example: RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/0/0/0</pre>	Specifies or creates a Gigabit Ethernet or 10-Gigabit Ethernet interface, where <i>interface-path-id</i> is the physical location of the interface using <i>rack/slot/module/port</i> notation, and enters interface configuration mode.
Step 3	<pre>bundle id bundle-id mode active Example: RP/0/RSP0/CPU0:router(config-if)# bundle id 100 mode active</pre>	Adds the interface as a member of the specified bundle, and runs LACP in active mode on the interface to exchange LACP packets for MGSCP.
Step 4	<pre>bundle port-priority priority Example: RP/0/RSP0/CPU0:router(config-if)# bundle port-priority 1</pre>	 Specifies the LACP priority for the interface and determines if a bundle interface is an active or protect link for MGSCP: Value of 1—Specifies the link is an active interface. Value other than 1—Specfies the link is a protect interface. The default is 32768.
Step 5	<pre>end or commit Example: RP/0/RSP0/CPU0:router (config-bfd-if)# end or RP/0/RSP0/CPU0:router(config-bfd-if)# commit</pre>	 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.

Command or Action	Purpose
	- 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 VRFs to Route Traffic to the Bundles

VRFs are the recommended way to route subscriber traffic to the bundles to be sure that all subscriber traffic remains with the same SCE device for statefulness. To configure VRFs for MGSCP, complete one of the following tasks:

Configuring VRFs with Static Routing

These steps summarize the tasks required to configure VRFs using static routing:

- 1. Configure two VRFs in global configuration—one each for the access and core sides of the network. Be sure to specify the IPv4 unicast address family.
- 2. Configure IPv4 addresses at each of the bundle interfaces and associate those addresses with the corresponding VRF that you configured in global configuration for the access and core side of the network.
- **3.** Configure IPv4 addresses at the Gigabit Ethernet physical interfaces and associate those addresses with the corresponding VRF that you configured in global configuration for the access and core side of the network.
- 4. Configure static routing using the **router static** command to map the access and core VRFs to their corresponding bundle interfaces.

For a sample configuration, see the "Example: Configuring VRFs with Static Routing" section.

Configuring VRFs with Dynamic Routing

VRFs for MGSCP are supported for both OSPF and BGP routing protocols. The general configuration of the VRFs in global configuration and at the bundle and physical interfaces is the same as for static routing.

These steps summarize the tasks required to configure VRFs using OSPF routing:

- 1. Configure two VRFs in global configuration—one each for the access and core sides of the network. Be sure to specify the IPv4 unicast address family.
- 2. Configure IPv4 addresses at each of the bundle interfaces and associate those addresses with the corresponding VRF that you configured in global configuration for the access and core side of the network.
- **3.** Configure IPv4 addresses at the Gigabit Ethernet physical interfaces and associate those addresses with the corresponding VRF that you configured in global configuration for the access and core side of the network.
- 4. Configure a dynamic routing protocol, such as OSPF, using the **router ospf** command to define the VRFs and associate the bundle and physical interfaces to the OSPF areas.

For a sample configuration, see the "Example: Configuring VRFs with OSPF Routing" section on page 310.

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 running LACP:

```
RP/0/RSP0/CPU0:Router(config)# config
```

RP/0/RSP0/CPU0:Router(config) # config

```
RP/0/RSP0/CPU0:Router(config-if)# interface Bundle-Ether 3
RP/0/RSP0/CPU0:Router(config-if)# ipv4 address 1.2.3.4/24
RP/0/RSP0/CPU0:Router(config-if)# bundle minimum-active bandwidth 620000
RP/0/RSP0/CPU0:Router(config-if)# bundle minimum-active links 1
RP/0/RSP0/CPU0:Router(config-if)# exit
RP/0/RSP0/CPU0:Router(config-if)# interface TenGigE 0/3/0/0
RP/0/RSP0/CPU0:Router(config-if)# bundle id 3 mode active
RP/0/RSP0/CPU0:Router(config-if)# no shutdown
RP/0/RSP0/CPU0:Router(config-if)# interface TenGigE 0/3/0/1
RP/0/RSP0/CPU0:Router(config-if)# interface TenGigE 0/3/0/1
RP/0/RSP0/CPU0:Router(config-if)# interface TenGigE 0/3/0/1
RP/0/RSP0/CPU0:Router(config-if)# bundle id 3 mode active
RP/0/RSP0/CPU0:Router(config-if)# no shutdown
RP/0/RSP0/CPU0:Router(config-if)# interface TenGigE 0/3/0/1
```

This example shows the configuration in the case of a mixed speed bundle:

```
RP/0/RSP0/CPU0:Router(config-if)# interface bundle-ether 50
RP/0/RSP0/CPU0:Router(config-if) # root
RP/0/RSP0/CPU0:Router(config-if) # interface TenGigE 0/0/0/11
RP/0/RSP0/CPU0:Router(config-if)# bundle id 50 mode active
RP/0/RSP0/CPU0:Router(config-if) # no shutdown
RP/0/RSP0/CPU0:Router(config-if) # interface TenGigE 0/0/0/16
RP/0/RSP0/CPU0:Router(config-if) # bundle id 50 mode active
RP/0/RSP0/CPU0:Router(config-if) # no shutdown
RP/0/RSP0/CPU0:Router(config-if) # interface TenGigE 0/0/0/27
RP/0/RSP0/CPU0:Router(config-if) # bundleid 50 mode active
RP/0/RSP0/CPU0:Router(config-if) # no shutdown
RP/0/RSP0/CPU0:Router(config-if) # interface HundredGigE 0/6/0/1
RP/0/RSP0/CPU0:Router(config-if)# bundleid 50 mode active
RP/0/RSP0/CPU0:Router(config-if) # no shutdown
RP/0/RSP0/CPU0:Router(config-if) # root
RP/0/RSP0/CPU0:Router(config)# commit
RP/0/RSP0/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 (13000000) kbps
MAC address (source): 0011.2233.4458 (Chassis pool)
Inter-chassis link:
                         No
Minimum active links / bandwidth:
                               1 / 1 kbps
Maximum active links: 64
Maximum acci
Wait while timer: 2000 m
Default
                        2000 ms
Load balancing: Deraur
Operational
 Flap suppression timer: Off
Cisco extensions: Disabled
 Cisco extensions:
mLACP:
                   Not configured
IPv4 BFD:
                    Not configured
          Device
                   State Port ID
                                     B/W, kbps
Port
 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 10000000
  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
```

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	slo	t 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

Example: Configuring a VLAN Link Bundle

The following example shows how to create and bring up two VLANs on an Ethernet bundle:

```
RP/0/RSP0/CPU0:Router(config-subif)# config
RP/0/RSP0/CPU0:Router(config-subif) # interface Bundle-Ether 1
RP/0/RSP0/CPU0:Router(config-ifsubif)# ipv4 address 1.2.3.4/24
RP/0/RSP0/CPU0:Router(config-ifsubif) # bundle minimum-active bandwidth 620000
RP/0/RSP0/CPU0:Router(config-if) # bundle minimum-active links
RP/0/RSP0/CPU0:Router(config-ifsubif)# exit
RP/0/RP0/CPU0:Router(config-subif) # ip addr 20.2.3.4/24
RP/0/RSP0/CPU0:Router(config-subif)# interface Bundle-Ether 1.1
RP/0/RSP0/CPU0:Router(config-subif)# dot1q vlan 10
RP/0/RSP0/CPU0:Router(config-subif) # ip addr 10.2.3.4/24
RP/0/RSP0/CPU0:Router(config-subifif)# no shutdown
RP/0/RSP0/CPU0:Router(config-subifif)# exit
RP/0/RSP0/CPU0:Router(config-if) # interface Bundle-Ether 1.2
RP/0/RSP0/CPU0:Router(config-subif) # dot1q vlan 10
RP/0/RSP0/CPU0:Router(config-subif)Router # ip addr20.2.3.4/24
RP/0/RSP0/CPU0:Router(config-subifif)# no shutdown
RP/0/RSP0/CPU0:Router(config-subifif) # exit
RP/0/RSP0/CPU0:Router(config) # interface gig 0/1/5/7
RP/0/RSP0/CPU0:Router(config-if)# bundle-id 1 mode act
RP/0/RSP0/CPU0:Router(config-if)# commit
RP/0/RSP0/CPU0:Router(config-if) # exit
```

Example: Configuring a POS Link Bundle

The following example shows how to join two ports to form a Packet-over-SONET (POS) link bundle:

```
RP/0/RSP0/CPU0:Router# config
RP/0/RSP0/CPU0:Router(config)# interface Bundle-POS 5
RP/0/RSP0/CPU0:Router(config-if)#ipv4 address 1.2.3.4/24
RP/0/RSP0/CPU0:Router(config-if)# bundle minimum-active bandwidth 620000
RP/0/RSP0/CPU0:Router(config-if)# bundle minimum-active bandwidth 620000
RP/0/RSP0/CPU0:Router(config-if)# exit
RP/0/RSP0/CPU0:Router(config-if)# interface POS 0/0/1/1
RP/0/RSP0/CPU0:Router(config-if)# bundle id 5
RP/0/RSP0/CPU0:Router(config-if)# no shutdown
RP/0/RSP0/CPU0:Router(config-if)# exit
```

Example: Configuring EFP Load Balancing on an Ethernet Link Bundle

The following example shows how to configure all egressing traffic on the fixed members of a bundle to flow through the same physical member link automatically.

```
RP/0/RP0/CPU0:router# configuration terminal
RP/0/RP0/CPU0:router(config)# interface bundle-ether 1.1 l2transport
RP/0/RP0/CPU0:router(config-subif)#bundle load-balancing hash auto
RP/0/RP0/CPU0:router(config-subif)#
```

The following example shows how to configure all egressing traffic on the fixed members of a bundle to flow through a specified physical member link.

```
RP/0/RP0/CPU0:router# configuration terminal
RP/0/RP0/CPU0:router(config)# interface bundle-ether 1.1 l2transport
RP/0/RP0/CPU0:router(config-subif)#bundle load-balancing hash 1
RP/0/RP0/CPU0:router(config-subif)#
```

Example: Configuring Multichassis Link Aggregation

This example shows how to configure POAs:

Active POA

```
interface Bundle-Ether10
mlacp iccp-group 1
mlacp port-priority 10
```

Standby POA

interface Bundle-Ether10
 mlacp iccp-group 1
 mlacp port-priority 20

This example shows how to configure ICCP:

```
redundancy iccp group
 member neighbor 1.2.3.4
 backbone interface GigabitEthernet0/0/0/0
 isolation recovery-delay 30
```

This example shows how to configure mLACP:

```
configure
redundancy iccp group 100
mlacp system mac 1.1.1
mlacp system priority 10
mlacp node 1
interface Bundle-Ether 3
mac-address 1.1.1
bundle wait-while 100
lacp switchover suppress-flaps 300
mlacp iccp-group 100
```

This example illustrates a switchover:

```
RP/0/0/CPU0:router# show bundle
```

```
Bundle-Ether1
Status:
                     αU
Local links <active/standby/configured>: 1 / 0 / 1
Local bandwidth <effective/available>: 1000000 (1000000) kbps
MAC address (source): 0000.deaf.0000 (Configured)
Minimum active links / bandwidth: 1 / 1 kbps
Maximum acc...
Wait while timer: Operational
Maximum active links: 64
                         100 ms
 Flap suppression timer: 300 ms
mLACP:
         Operational
 ICCP Group:
                      1
 Role:
                    Active
 Foreign links <active/configured>: 0 / 1
 Switchover type: Non-revertive
                       300 s
 Recovery delay:
 Maximize threshold:
                          Not configured
                     Not configured
 IPv4 BFD:
          Device State Port ID B/W, kbps
Port
 _____ ____
Gi0/0/0/0 Local Active 0x8001, 0x9001 1000000
  Link is Active
 Gi0/0/0/0 5.4.3.2 Standby 0x8002, 0xa001 1000000
  Link is marked as Standby by mLACP peer
RP/0/0/CPU0:router#mlacp switchover Bundle-Ether 1
This will trigger the peer device (Node 5.4.3.2 in IG 1) to become active for Bundle-Ether1.
This may result in packet loss on the specified bundle.
Proceed with switch over? [confirm]
RP/0/0/CPU0:Jan 31 23:46:44.666 : BM-DISTRIB[282]: %L2-BM-5-MLACP BUNDLE ACTIVE : This
device is no longer the active device for Bundle-Ether1
RP/0/0/CPU0:Jan 31 23:46:44.668 : BM-DISTRIB[282]: %L2-BM-6-ACTIVE : GigabitEthernet0/0/0/0
is no longer Active as part of Bundle-Ether1 (Not enough links available to meet
minimum-active threshold)
RP/0/0/CPU0:router#show bundle
Mon Jun 7 06:04:17.778 PDT
Bundle-Ether1
Status:
                     mLACP hot standby
Local links <active/standby/configured>: 0 / 1 / 1
Local bandwidth <effective/available>: 0 (0) kbps
MAC address (source): 0000.deaf.0000 (Configured)
Minimum active links / bandwidth: 1 / 1 kbps
Maximum active links:
64
                          100 ms
 Flap suppression timer: 300 ms
mLACP:
                    Operational
 ICCP Group:
                       1
 Role:
                    Standby
 Foreign links <active/configured>: 1 / 1
 Switchover type: Non-revertive
                       300 s
 Recovery delay:
 Maximize threshold:
                         Not configured
 IPv4 BFD:
                     Not configured
Port
          Device
                     State Port ID B/W, kbps
 _____ ____
                                               -----
```

Gi0/0/0/0 Local Standby 0x8003, 0x9001 1000000 mLACP peer is active Gi0/0/0/0 5.4.3.2 Active 0x8002, 0xa001 1000000 Link is Active RP/0/0/CPU0:router#

This example shows how to add a backup pseudowire to a VPLS access pseudowire:

```
l2vpn bridge group bg1
bridge-domain bd1
neighbor 101.101.101.01 pw-id 5000
    pw-class class1
        backup neighbor 102.102.102.102 pw-id 3000
        pw-class class1
        !
    !
    !
    !
    !
    !
    !
    !
    !
    !
```

This example shows how to configure one-way pseudowire redundancy behavior when redundancy group is configured:

```
l2vpn pw-class class_mpls
  encapsulation mpls
   redundancy one-way
 !
!
```

This example illustrates an overall MC-LAG configuration:

Topology:

```
DHD POA 1 POA 2

Gi0/0/0/0 ------ Gi0/0/0/0

Gi0/0/0/1 ----- Gi0/0/0/1

Gi0/0/0/2

Gi0/0/0/3 ----- Gi0/0/0/2

Gi0/0/0/2 Gi0/0/0/2

Gi0/0/0/3 ----- Gi0/0/0/3

Gi0/0/0/4 ----- Gi0/0/0/3
```

On POA 1:

```
redundancy
iccp
group 1
mlacp node 1
mlacp system mac 000d.000e.000f
mlacp system priority 1
member
neighbor 5.4.3.2
!
!
!
!
interface Bundle-Ether1
lacp switchover suppress-flaps 300
```

L

```
mlacp iccp-group 1
mac-address 0.deaf.0
bundle wait-while 100
1
interface Loopback0
ipv4 address 5.4.3.1 255.255.255.255
Т
interface GigabitEthernet0/0/0/0
description Connected to DHD Gi0/0/0/0
bundle id 1 mode active
lacp period short
no shutdown
interface GigabitEthernet0/0/0/3
description Connected to POA2 Gi0/0/0/3
ipv4 address 1.2.3.1 255.255.255.0
proxy-arp
no shutdown
!
router static
address-family ipv4 unicast
 5.4.3.2/32 1.2.3.2
 1
1
mpls ldp
router-id 5.4.3.1
discovery targeted-hello accept
log
neighbor
 1
 interface GigabitEthernet0/0/0/3
 1
!
```

On POA 2:

```
redundancy
iccp
group 1
 mlacp node 2
 mlacp system mac 000d.000e.000f
 mlacp system priority 1
 member
 neighbor 5.4.3.1
 1
 1
1
!
interface Bundle-Ether1
lacp switchover suppress-flaps 300
mlacp iccp-group 1
mac-address 0.deaf.0
bundle wait-while 100
interface Loopback0
ipv4 address 5.4.3.2 255.255.255.255
!
interface GigabitEthernet0/0/0/0
description Connected to DHD Gi0/0/0/3
bundle id 1 mode active
lacp period short
no shutdown
!
```

```
interface GigabitEthernet0/0/0/3
description Connected to POA1 Gi0/0/0/3
ipv4 address 1.2.3.2 255.255.255.0
proxy-arp
no shutdown
!
router static
address-family ipv4 unicast
 5.4.3.1/32 1.2.3.1
1
!
mpls ldp
router-id 5.4.3.2
discovery targeted-hello accept
log
neighbor
interface GigabitEthernet0/0/0/3
 1
!
```

On the DHD:

```
interface Bundle-Ether1
lacp switchover suppress-flaps 300
bundle wait-while 100
T.
interface GigabitEthernet0/0/0/0
description Connected to POA1 Gi0/0/0/0
bundle id 1 mode active
lacp period short
no shutdown
1
interface GigabitEthernet0/0/0/3
description Connected to POA2 Gi0/0/0/0
bundle id 1 mode active
lacp period short
no shutdown
1
```

Configuration Examples for MGSCP

This figure illustrates a sample network with a single Cisco ASR 9000 Series Router as a dispatcher for a cluster of SCE devices that is used as an example for the sample configurations.



Figure 29: Cisco ASR 9000 Series Router as Dispatcher for SCE Cluster

This section includes the following examples:

Example: Configuring Bundle Interfaces and Member Links

This example shows how to configure the two bundles on the Cisco ASR 9000 Series Router shown in the figure above. Each bundle supports a maximum of two active links (configurations for both bundles must match), with one backup protect link.

The bundle interface members in Ethernet bundle 100 connect the SCE device links for the subscriber side of the network using load balancing based on source IP address. The bundle interface members in Ethernet bundle 200 connect the SCE device links for the core side of the network using load balancing based on destination IP address.

Subscriber-Facing Access Bundle Configuration

```
interface Bundle-Ether 100
description Faces-SCE-Subscriber-Side
vrf access
 ipv4 address 10.0.0.1 255.255.255.0 <<-- Same subnet as Bundle-Ether 200
lacp cisco enable link-order signaled <<-- Enables Cisco LACP extensions, required for
MGSCP
bundle load-balancing hash src-ip <<-- Hashes traffic based on source (Subscriber) IP
bundle maximum-active links 2
!
interface GigabitEthernet 0/0/0/0
description to SCE1
bundle id 100 mode active
bundle port-priority 1
ļ
interface GigabitEthernet 0/0/0/1
description to SCE2
bundle id 100 mode active
bundle port-priority 1
1
interface GigabitEthernet 0/0/0/3
```

description to SCE3 (backup) bundle id 100 mode active

Core-Facing Bundle Configuration

```
interface Bundle-Ether 200
description Faces-SCE-Network-Side
vrf core
 ipv4 address 10.0.0.2 255.255.255.0 <<-- Same subnet as Bundle-Ether100
 lacp cisco enable link-order signaled <<-- Enables Cisco LACP extensions, required for
MGSCP
bundle load-balancing hash dst-ip <<-- Hashes traffic based on destination (Subscriber)
ΤP
bundle maximum active links 2
interface GigabitEthernet 0/0/1/0
description from SCE1
bundle id 200 mode active
bundle port-priority 1
interface GigabitEthernet 0/0/1/1
description from SCE2
bundle id 200 mode active
bundle port-priority 1
1
interface GigabitEthernet 0/0/1/2
description from SCE3 (standby)
bundle id 200 mode active
```

Examples: Configuring VRFs to Route Traffic to the Bundles

To ensure that the traffic to and from the same subscriber is going through the same port of the SCE, VRFs are recommended. You need to configure two VRFs for MGSCP: One for the access traffic, and one for the core traffic.

The examples in this section also show two different ways that you can route using VRFs with either static or dynamic (OSPF) routing for the bundle interface at the VRF:

Example: Configuring VRFs with Static Routing

In the following configuration examples, VRFs are established for the core and access sides of the network using IPv4. From there, the bundle interface addresses for each side are each configured as part of the VRF, as well as two physical interfaces. The final piece of the configuration shows how to configure a static route to each VRF using the bundle interfaces.

VRF Global Configuration

```
vrf core
address-family ipv4 unicast
import route-target
1:1
!
export route-target
1:1
!
vrf access
address-family ipv4 unicast
```

L

```
import route-target
1:1
!
export route-target
1:1
!
```

VRF Configuration on Bundle Interfaces

```
interface Bundle-Ether100
vrf access
ipv4 address 10.0.0.1 255.255.255.0
!
interface Bundle-Ether200
vrf core
ipv4 address 10.0.0.2 255.255.255.0
```

VRF Configuration on Physical Interfaces

```
interface GigabitEthernet0/2/0/1
desc Subscriber-facing
vrf access
ipv4 address 10.10.1.2 255.255.255.0
interface GigabitEthernet0/2/0/9
desc Network-facing
vrf core
ipv4 address 10.20.1.2 255.255.255.0
negotiation auto
```

Static Routing Configuration for the VRFs to the Bundle Interfaces

```
router static
vrf core
address-family ipv4 unicast
0.0.0.0/0 10.0.0.1
!
!
vrf access
address-family ipv4 unicast
0.0.0.0/0 10.0.0.2
!
!
```

Example: Configuring VRFs with OSPF Routing

In the following configuration examples, VRFs are established for the core and access sides of the network using IPv4. From there, you configure an OSPF routing instance and area to include the VRFs and associate the bundle and physical interfaces.

Global VRF Configuration

```
vrf core
address-family ipv4 unicast
import route-target
1:1
export route-target
```

```
1:1
vrf access
address-family ipv4 unicast
import route-target
1:1
export route-target
1:1
```

VRF Configuration on Physical Interfaces

OSPF Routing Configuration for the VRFs and the Bundle and Physical Interfaces

```
router ospf 100
vrf core
router-id 10.20.1.2
area 0
interface Bundle-Ether200
interface GigabitEthernet0/2/0/9
vrf access
router-id 10.10.1.2
area 0
interface Bundle-Ether100
interface GigabitEthernet0/2/0/1
```

Example: Configuring MGSCP with ABF to Route Traffic to the Bundles

The following example routes traffic to the bundles using access lists to forward the traffic.

```
ipv4 access-list inbound
!
! Set the nexthop address to be a virtual IP address on the same network
! as the access bundle.
10 permit ipv4 any any nexthop 10.10.1.5
!
ipv4 access-list outbound
! Set the nexthop address to be a virtual IP address on the same network
! as the core bundle.
10 permit ipv4 any any nexthop 10.20.1.5
1
! Configure static ARP for the virtual IP addresses
arp vrf default 10.10.1.5 0024.98eb.bf8a ARPA
arp vrf default 10.20.1.5 0024.98eb.bf8b ARPA
interface Bundle-Ether100
ipv4 address 10.10.1.2 255.255.255.0
```

```
!
interface Bundle-Ether200
ipv4 address 10.20.1.2 255.255.255.0
!
interface GigabitEthernet0/2/0/1
ipv4 address 10.10.1.3 255.255.255.0
ipv4 access-group inbound
!
interface GigabitEthernet0/2/0/9
ipv4 address 10.20.1.3 255.255.0
ipv4 access-group outbound
!
```

Example: Configuring MGSCP with ABF to Route Traffic to the Bundles



Configuring Traffic Mirroring

This module describes the configuration of the traffic mirroring feature. Traffic mirroring is sometimes called port mirroring, or switched port analyzer (SPAN).

Feature	History	for	Traffic	Mirr	oring
---------	---------	-----	---------	------	-------

Release 3.9.1	This feature was introduced.
Release 4.0.1	 These traffic mirroring features were added: Traffic mirroring over a pseudowire Flow or ACL-based traffic mirroring Layer 3 interface support Partial packet mirroring
Release 5.1.0	The Sampled Traffic Mirroring feature was introduced.
Release 7.1.2	The SPAN to File feature was introduced.
Release 7.1.2	The File Mirroring feature was introduced.

- Introduction to Traffic Mirroring, on page 319
- Restrictions for Traffic Mirroring, on page 325
- Configuring Traffic Mirroring, on page 327
- Traffic Mirroring Configuration Examples, on page 340
- Troubleshooting Traffic Mirroring, on page 345
- SPAN to File, on page 348
- Introduction to File Mirroring, on page 352

Introduction to Traffic Mirroring

Traffic mirroring, which is sometimes called port mirroring, or Switched Port Analyzer (SPAN) is a Cisco proprietary feature that enables you to monitor Layer 2 or Layer 3 network traffic passing in, or out of, a set of Ethernet interfaces. You can then pass this traffic to a network analyzer for analysis.

Traffic mirroring copies traffic from one or more Layer 3 or Layer 2 interfaces or sub-interfaces, including Layer 2 link bundle interfaces or sub-interfaces, and sends the copied traffic to one or more destinations for analysis by a network analyzer or other monitoring device. Traffic mirroring does not affect the switching of traffic on the source interfaces or sub-interfaces, and allows the mirrored traffic to be sent to a destination interfaceor sub-interface.

Traffic mirroring was introduced on switches because of a fundamental difference between switches and hubs. When a hub receives a packet on one port, the hub sends out a copy of that packet from all ports except from the one at which the hub received the packet. In the case of switches, after a switch boots, it starts to build up a Layer 2 forwarding table on the basis of the source MAC address of the different packets that the switch receives. After this forwarding table is built, the switch forwards traffic that is destined for a MAC address directly to the corresponding port.

For example, if you want to capture Ethernet traffic that is sent by host A to host B, and both are connected to a hub, just attach a traffic analyzer to this hub. All other ports see the traffic between hosts A and B.



Figure 30: Traffic Mirroring Operation on a Hub

On a switch or router, after the host B MAC address is learned, unicast traffic from A to B is only forwarded to the B port. Therefore, the traffic analyzer does not see this traffic.



In this configuration, the traffic analyzer captures only traffic that is flooded to all ports, such as:

- · Broadcast traffic
- Multicast traffic with CGMP or Internet Group Management Protocol (IGMP) snooping disabled
- Unknown unicast traffic on a switch

An extra feature is necessary that artificially copies unicast packets that host A sends. This extra feature is traffic mirroring. When traffic mirroring is enabled, the traffic analyzer is attached to a port that is configured to receive a copy of every packet that host A sends. This port is called a traffic mirroring port. The other sections of this document describe how you can fine tune this feature.

Sampled Traffic Mirroring

SPAN is supported on all types of forwarding interfaces of the main interface (port level) such as, L2, L3 interfaces, sub-interface, bundle interface, and BNG interface. But Sampled SPAN is supported only at port level. Sampled SPAN is configurable in ingress direction only. SPAN and Sampled SPAN cannot be configured at the same time on main interface (at port level). When Sampled SPAN is enabled on main interface, SPAN is still configurable on rest of the forwarding interfaces on the port.

When Sampled SPAN is enabled on the underlying physical port and SPAN is configured on a forwarding interface, the packets are mirrored as follows:

- Sampled packet on the physical port is mirrored just to the destination port of the Sampled SPAN session.
- Non-sampled packet is mirrored for each of the regular SPAN session on the associated forwarding interface.

Sampled Traffic Mirroring allows you to:

- 1. Sample the packets based on a configured interval.
- 2. Apply Sampled SPAN on a physical port in order to include all forwarding interfaces on that port.
- **3.** Configure the Sampling rate of monitoring that is configured for each source port. You can choose to configure one of these sampling rates; 1K, 2K, 4K, 8K, and 16K. For example, when 4K is configured as the sampling rate, for every 4K packets on the source port one packet is sampled and mirrored to the destination port.
- 4. Use Sampled SPAN along with Traffic Mirroring.
- 5. Enable Sampled SPAN on every bundle member, if the physical port is part of a link bundle.
- 6. Use all destination ports that were supported for SPAN.
- 7. Enable statistics support on each monitoring session.
- **8.** Truncate and mirror a fixed portion of each mirrored packet (for example, the first 64 bytes of every packet received from the source port is mirrored to the destination port). You can configure the offset or the amount of fixed portion.

You can configure these source to destination combinations in sampled SPAN:

- Physical Port mirrored to Physical Port
- · Physical Port mirrored to Pseudo-wire
- Bundle member port mirrored to Physical Port
- · Bundle member port mirrored to Pseudo-wire

Implementing Traffic Mirroring on the Cisco ASR 9000 Series Router

Traffic Mirroring Terminology

- Ingress traffic—Traffic that enters the switch.
- Egress traffic—Traffic that leaves the switch.

- Source port—A port that is monitored with the use of traffic mirroring. It is also called a monitored port.
- Destination port—A port that monitors source ports, usually where a network analyzer is connected. It is also called a monitoring port.
- Monitor session—A designation for a collection of traffic mirroring configurations consisting of a single destination and, potentially, many source interfaces.

Characteristics of the Source Port

A source port, also called a monitored port, is a switched or routed port that you monitor for network traffic analysis. In a single local or remote traffic mirroring session, you can monitor source port traffic, such as received (Rx) for ingress traffic, transmitted (Tx) for egress traffic, or bidirectional (for both ingress and egress traffic). Your router can support any number of source ports (up to a maximum number of 800).

A source port has these characteristics:

• It can be any port type, such as Bundle Interface, Gigabit Ethernet, 10-Gigabit Ethernet, or EFPs.

Note

Bridge group virtual interfaces (BVIs) are not supported.

- Each source port can be monitored in only one traffic mirroring session.
- It cannot be a destination port.
- Partial Packet Mirroring. The first 64 to 256 bytes of the packet can be mirrored.
- Each source port can be configured with a direction (ingress, egress, or both) to monitor. For bundles, the monitored direction applies to all physical ports in the group.

Figure 31: Network Analysis on a Cisco ASR 9000 RouterWith Traffic Mirroring



In the figure above, the network analyzer is attached to a port that is configured to receive a copy of every packet that host A sends. This port is called a traffic mirroring port.

Characteristics of the Monitor Session

A monitor session is a collection of traffic mirroring configurations consisting of a single destination and, potentially, many source interfaces. For any given monitor session, the traffic from the source interfaces (called *source ports*) is sent to the monitoring port (called the *destination port*). Some optional operations such as VLAN tag imposition and ACL filtering can be performed on the mirrored traffic streams. If there is

more than one source port in a monitoring session, the traffic from the several mirrored traffic streams is combined at the destination port. The result is that the traffic that comes out of the destination port is a combination of the traffic from one or more source ports, and the traffic from each source port may or may not have VLAN push operations or ACLs applied to it.

Monitor sessions have these characteristics:

- A single Cisco ASR 9000 Router can have a maximum of eight monitor sessions.
- A single monitor session can have only one destination port.
- A single destination port can belong to only one monitor session.
- A single Cisco ASR 9000 Router can have a maximum of 800 source ports.
- A monitor session can have a maximum of 800 source ports, as long as the maximum number of source ports from all monitoring sessions does not exceed 800.

Characteristics of the Destination Port

Table 14: Feature History Table

Feature Name	Release	Description
SPAN support for bundle as a destination interface	Release 7.5.1	This feature adds the ability to configure a bundle destination for SPAN. Bundle interfaces increase the bandwidth by allowing forwarding across all available members of the bundle. Additionally, they provide redundancy in the case of link failure.

Each local session or remote destination session must have a destination port (also called a monitoring port) that receives a copy of the traffic from the source ports.

A destination port has these characteristics:

- A destination port must reside on the same router as the source port.
- A destination port can be any Ethernet physical port, EFP, pseudowire, or a bundle interface.
- A destination port can only be a Layer 2 transport interface. An L3 interface as a SPAN destination cannot be configured on the Cisco ASR 9000 Series Router.
- A destination port can be a trunk (main) interface or a subinterface.
- At any one time, a destination port can participate in only one traffic mirroring session. A destination port in one traffic mirroring session cannot be a destination port for a second traffic mirroring session. In other words, no two monitor sessions can have the same destination port.
- A destination port cannot also be a source port.
- The destination interface MTU must be configured to a value greater than or equal to the source interface MTU for the 4th and 5th generation line cards.

- 1 Source traffic mirroring ports (can be traffic lngress traffic Router Network analyzer
- Figure 32: Network Analysis on a Cisco ASR 9000 Router With Traffic Mirroring

Supported Traffic Mirroring Types

These traffic mirroring types are supported:

- Local traffic mirroring. This is the most basic form of traffic mirroring. The network analyzer or sniffer is directly attached to the destination interface. In other words, all monitored ports are all located on the same switch as the destination port.
- Remote traffic mirroring (known as R-SPAN). In this case, the network analyzer is not attached directly to the destination interface, but is on a VLAN accessible to the switch. For example, the destination interface is a sub-interface with a VLAN encapsulation.

A restricted form of remote traffic mirroring can be implemented by sending traffic to a single destination port that pushes a VLAN tag, instead of switching through a bridge domain. Remote traffic mirroring:

- Allows decoupling of the network analyzer and destination, but there is no on-the-box redundancy.
- Allows multiple remote network analyzers as long as they can attach to the traffic mirroring VLAN.

This is supported on Cisco IOS XR software because the destination port is an EFP that can push a VLAN tag.

- Pseudowire traffic mirroring (known as PW-SPAN in Cisco IOS Software). Instead of using a standard
 destination interface, traffic is mirrored to a remote site through an MPLS pseudowire.
- ACL-based traffic mirroring. Traffic is mirrored based on the configuration of the global interface ACL.
- Partial Packet Mirroring. The first 64 to 256 bytes of the packet can be mirrored.
- Layer 2 or Layer 3 traffic mirroring is supported. Both Layer 2 and Layer 3 source ports can be mirrored.

Pseudowire Traffic Mirroring

You can configure the traffic mirroring destination port to be a pseudowire rather than a physical port. In this case, the system mirrors the designated traffic on the source port over the pseudowire to a central location. This allows the centralization of expensive network traffic analysis tools.

Because the pseudowire carries only mirrored traffic, this traffic is unidirectional. There must not be any traffic coming from the remote provider edge.

In such a pseudowire traffic mirroring scenario, though the system mirrors traffic successfully, the statistics for sent pseudowire packet statistics remains zero.

To protect the pseudowire traffic mirroring path against network failures, it is possible to configure a traffic engineering tunnel as the preferred path and enable fast reroute protection for the pseudowire.

Figure 33: Pseudowire Traffic Mirroring



ACL-Based Traffic Mirroring

You can mirror traffic based on the definition of a global interface access list (ACL). If you are mirroring Layer 2 traffic, the ACL is configured using the **ethernet-services access-list** command with the **capture** keyword. When you are mirroring Layer 3 traffic, the ACL is configured using the **ipv4 access-list** or **ipv6 access-list** command with the **capture** keyword. The **permit** and **deny** commands determine the behavior of regular traffic. The **capture** keyword designates that the packet is to be mirrored to the destination port.

Restrictions for Traffic Mirroring

A maximum of eight monitoring sessions are supported. You can configure 800 source ports on a single monitoring session or an aggregate of 800 source ports over eight monitoring sessions.

These forms of traffic mirroring are not supported:

- ACL-Based SPAN on Layer 3 interface configured with a dot1q encapsulation.
- Mirroring traffic to a GRE tunnel (also known as Encapsulated Remote Switched Port Analyzer [ER-SPAN] in Cisco IOS Software).
- Mirroring traffic from a full bridge domain (also known as VLAN-based SPAN in Cisco IOS Software).

- Mirroring traffic on an individual bundle member interface is not supported. SPAN must be configured only on a bundle interface and it is applied to all members.
- If the destination of traffic mirroring is an nV satellite port and ICL is configured with a bundle interface, then replicated packets are not forwarded to the destination.
- The system does not support MAP-T inline and SPAN on the same NPU.
- To avoid traffic loss, disable SPAN, if enabled on MAP-E/T service-inline interfaces.
- SPAN is not supported on those line card ports that are carrying traffic bound for a VSM. This behaviour is observed only on the Cisco ASR 9000 High Density 100GE Ethernet line cards and Cisco ASR 9000 Series 24-Port and 48-Port Dual-Rate 10GE/1GE line cards.
- When you configure ingress port SPAN on an interface, BFD sessions such as BFD-over-BVI, may
 encounter flaps during traffic congestion. This happens because the BFD-over-BVI traffic is handled via
 Priority Normal Traffic Manager Loopback queue in spite of prioritising the BFD-over-BVI packets.
 Except for the Cisco ASR 9000 Series 5th Generation High-Density Line Cards and Cisco ASR 9000
 Series 4th Generation QSFP28 based dense 100GE Line Cards, this limitation is observed in all the other
 line cards.
- On Cisco ASR 9903 routers, the Online Insertion and Removal (OIR) of a Port Expansion Card (PEC) with BFD sessions, support 300ms asynchronous timers and 150ms echo timers. BFD sessions with less than the supported timer values may encounter flaps during the PEC OIR.

Restrictions of Sampled Traffic Mirroring

These are the restrictions of Sampled Traffic Mirroring:

- Sampled SPAN can be applied to ingress traffic only.
- The source for sampled SPAN must be on Cisco ASR 9000 Enhanced Ethernet Line Cards.
- · Sampled SPAN works only on physical interfaces.
- The source port cannot be on bundles; however it can be applied to bundle member links.
- Sampled SPAN does not work on sub-interfaces, however it can be applied to a physical port with sub-interfaces(main port).
- Only these intervals are accepted: 512, 1K, 2K, 4K, 8K, and 16K. The default interval is 16K.
- Sampled SPAN is configurable at physical port level only.
- Sampled SPAN rate is ingress port specific and not session specific. This means that a destination port
 can take multiple ingress sampled ports at different sampling rates.
- In the case of a bundle interface, you must configure Sampled SPAN on all the physical ports that are members of the bundle.
- ACL filtering is not supported for Sampled Mirrored Traffic.

Performance Impact with Traffic Mirroring

It is recommended that you do not mirror more than 15% of your total transit traffic. On the Cisco ASR 9000 Ethernet Line Card, that uses Ten Gigabit Ethernet interfaces or bundle interfaces there is a limit of 1.5G of data on each of the ingress and egress traffic that can be mirrored. This limitation is not applicable on the Cisco ASR 9000 Enhanced Ethernet Line Card.

Configuring Traffic Mirroring

These tasks describe how to configure traffic mirroring:

How to Configure Local Traffic Mirroring

SUMMARY STEPS

- 1. configure
- **2.** monitor-session session-name
- **3.** destination interface dest-interface
- 4. exit
- 5. interface source-interface
- 6. l2transport
- 7. monitor-session session-name [direction {rx-only | tx-only]
- 8. end or commit
- 9. show monitor-session [session-name] status [detail] [error]

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	monitor-session session-name	Defines a monitor session and enters monitor session
	Example:	configuration mode.
	<pre>RP/0/RSP0/CPU0:router(config) # monitor-session mon1 RP/0/RSP0/CPU0:router(config-mon) #</pre>	
Step 3	destination interface <i>dest-interface</i> Example:	Specifies the destination interface to which traffic is replicated.
	<pre>RP/0/RSP0/CPU0:router(config-mon)# destination interface gigabitethernet0/0/0/15</pre>	

	Command or Action	Purpose
Step 4	exit Example:	Exits monitor session configuration mode and returns to global configuration mode.
	<pre>RP/0/RSP0/CPU0:router(config-mon)# exit RP/0/RSP0/CPU0:router(config)#</pre>	
Step 5	<pre>interface source-interface Example: RP/0/RSP0/CPU0:router(config)# interface gigabitethernet0/0/0/11</pre>	Enters interface configuration mode for the specified interface. The interface number is entered in <i>rack/slot/module/port</i> notation. For more information about the syntax for the router, use the question mark (?) online help function.
Step 6	<pre>12transport Example: RP/0/RSP0/CPU0:router(config-if)# l2transport</pre>	 (Optional) Enables Layer 2 transport mode on the interface and enters Layer 2 transport configuration mode. Note Use the l2transport command to mirror all traffic types.
Step 7	<pre>monitor-session session-name [direction {rx-only tx-only] Example: RP/0/RSP0/CPU0:router(config-if-l2)# monitor-session mon1</pre>	Specifies the monitor session to be used on this interface. Use the direction keyword to specify that only ingress or only egress traffic is mirrored.
Step 8	end or commit	Saves configuration changes.
	Example: RP/0/RSP0/CPU0:router(config-if-l2)# end or	• When you issue the end command, the system prompts you to commit changes: Uncommitted changes found, commit them before exiting (yes/no/cancel)?
	RP/0/RSP0/CPU0:router(config-if-l2)# commit	 [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.

	Command or Action	Purpose
Step 9	show monitor-session [session-name] status [detail] [error]	Displays information about the monitor session.
	Example:	
	RP/0/RSP0/CPU0:router# show monitor-session	

How to Configure Remote Traffic Mirroring

SUMMARY STEPS

- 1. configure
- 2. monitor-session session-name
- **3. destination interface** *dest-subinterface*
- 4. exit
- 5. interface *dest-subinterface* l2transport
- 6. encapsulation dot1q vlan
- 7. rewrite ingress tag pop *tag-to-remove*
- 8. interface source-subinterface [l2transport]
- 9. monitor-session session-name [direction {rx-only | tx-only]
- 10. end or commit
- **11.** show monitor-session [session-name] status [detail] [error]

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	monitor-session session-name	Defines a monitor session and enters monitor session
	Example:	configuration mode.
	<pre>RP/0/RSP0/CPU0:router(config)# monitor-session mon1 RP/0/RSP0/CPU0:router(config-mon)#</pre>	
Step 3	destination interface <i>dest-subinterface</i> Example:	Specifies the destination subinterface to which traffic is replicated.
	<pre>RP/0/RSP0/CPU0:router(config-mon)# destination interface gigabitethernet0/0/0/15</pre>	

	Command or Action	Purpose
Step 4	exit Example:	Exits monitor session configuration mode and returns to global configuration mode.
	<pre>RP/0/RSP0/CPU0:router(config-mon)# exit RP/0/RSP0/CPU0:router(config)#</pre>	
Step 5	<pre>interface dest-subinterface l2transport Example: RP/0/RSP0/CPU0:router(config)# interface gigabitethernet0/0/0/11.10 l2transport</pre>	Enters interface configuration mode for the specified sub-interface. The interface number is entered in <i>rack/slot/module/port</i> notation. For more information about the syntax for the router, use the question mark (?) online help function.
		The l2transport keyword is used to enable Layer 2 transport mode on the destination subinterface.
Step 6	encapsulation dot1q vlan Example: RP/0/RSP0/CPU0:router(config-if)# encapsulation dot1g 1	Specifies 802.1Q encapsulation and the VLAN number that is used.
Step 7	<pre>rewrite ingress tag pop tag-to-remove Example: RP/0/RSP0/CPU0:router(config-if)# rewrite ingress tag pop 1</pre>	Specifies to remove the outer tag only for the EFP.
Step 8	<pre>interface source-subinterface [l2transport] Example: RP/0/RSP0/CPU0:router(config)# interface gigabitethernet0/0/0/11.10 l2transport</pre>	Enters interface configuration mode for the specified subinterface. The interface number is entered in <i>rack/slot/module/port</i> notation. For more information about the syntax for the router, use the question mark (?) online help function. To configure a Layer 2 subinterface to be the source interface, use the l2transport keyword to enable Layer 2
		transport mode on the subinterface.
Step 9	monitor-session session-name [direction {rx-only tx-only]	Specifies the monitor session to be used on this interface. Use the direction keyword to specify that only ingress or
	Example:	egress traffic is mirrored.
	<pre>RP/0/RSP0/CPU0:router(config-if-l2)# monitor-session mon1</pre>	
Step 10	end or commit	Saves configuration changes.
	Example: RP/0/RSP0/CPU0:router(config-if)# end	• When you issue the end command, the system prompts you to commit changes:
	or	Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:

	Command or Action	Purpose
	RP/0/RSP0/CPU0:router(config-if)# commit	- Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		- Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		- Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.
		Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.
Step 11	show monitor-session [session-name] status [detail] [error]	Displays information about the traffic mirroring session.
	Example:	
	RP/0/RSP0/CPU0:router# show monitor-session	

How to Configure Traffic Mirroring over Pseudowire

SUMMARY STEPS

- 1. configure
- 2. monitor-session session-name
- **3**. destination psuedowire
- 4. exit
- **5. interface** *source-interface*
- 6. l2transport
- 7. monitor-session *session-name* [direction {rx-only | tx-only}]
- 8. exit
- 9. exit
- **10.** exit
- 11. l2vpn
- 12. pw-class class-name
- **13**. encapsulation mpls
- 14. exit
- 15. exit
- **16. xconnect group** *group-name*
- **17. p2p** *xconnect-name*

- **18.** monitor-session session-name
- 19. neighbor peer-ip pw-id pseudowire-id
- **20. pw-class** *class-name*
- 21. end or commit
- 22. show monitor-session [session-name] status [detail] [error]

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	monitor-session session-name	Defines a monitor session and enters monitor session
	Example:	configuration mode.
	RP/0/RSP0/CPU0:router(config)# monitor-session mon1 RP/0/RSP0/CPU0:router(config-mon)#	
Step 3	destination psuedowire	Specifies that the traffic is replicated to a pseudowire.
	Example:	
	<pre>RP/0/RSP0/CPU0:router(config-mon)# destination pseudowire</pre>	
Step 4	exit	Exits monitor session configuration mode and returns to
	Example:	global configuration mode.
	<pre>RP/0/RSP0/CPU0:router(config-mon)# exit RP/0/RSP0/CPU0:router(config)#</pre>	
Step 5	interface source-interface	Enters interface configuration mode for the specified
	Example:	<i>rack/slot/module/port</i> notation. For more information about
	<pre>RP/0/RSP0/CPU0:router(config)# interface gigabitethernet0/0/0/11.10</pre>	the syntax for the router, use the question mark (?) online help function.
Step 6	12transport	(Optional) Enables Layer 2 transport mode on the
	Example:	subinterface and enters Layer 2 transport configuration mode.
	<pre>RP/0/RSP0/CPU0:router(config-if)# l2transport</pre>	 Note Use the l2transport command to mirror all traffic types.
		1

I

	Command or Action	Purpose
Step 7	monitor-session session-name [direction {rx-only tx-only}]	Specifies the monitor session to be used on this interface. Use the direction keyword to specify that only ingress or
	Example:	egress traffic is mirrored.
	RP/0/RSP0/CPU0:router(config-if-12)# monitor-session mon1	
Step 8	exit	Exits monitor session configuration mode and returns to 12transport configuration mode.
	Example:	
	<pre>RP/0/RSP0/CPU0:router(config-if-mon)# exit RP/0/RSP0/CPU0:router(config-if-l2)#</pre>	
Step 9	exit	Exits l2transport configuration mode and returns to
	Example:	interface configuration mode.
	<pre>RP/0/RSP0/CPU0:router(config-if-l2)# exit RP/0/RSP0/CPU0:router(config-if)#</pre>	
Step 10	exit	Exits interface configuration mode and returns to global
	Example:	configuration mode.
	<pre>RP/0/RSP0/CPU0:router(config-if)# exit RP/0/RSP0/CPU0:router(config)#</pre>	
Step 11	l2vpn	Enters Layer 2 VPN configuration mode.
	Example:	
	<pre>RP/0/RSP0/CPU0:router(config)# l2vpn RP/0/RSP0/CPU0:router(config-l2vpn)#</pre>	
Step 12	pw-class class-name	Configures a pseudowire class template and enters
	Example:	pseudowire class template configuration mode.
	RP/0/RSP0/CPU0:router(config-l2vpn)# pw-class pw1	
Step 13	encapsulation mpls	Configures the pseudowire encapsulation to MPLS.
	Example:	
	<pre>RP/0/RSP0/CPU0:router(config-l2vpn-pwc)# encapsulation mpls</pre>	
Step 14	exit	Exits pseudowire encapsulation configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router(config-l2vpn-pwc-mpls)# exit RP/0/RSP0/CPU0:router(config-l2vpn-pwc)	

	Command or Action	Purpose
Step 15	exit	Exits pseudowire class template configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router(config-l2vpn-pwc)# exit RP/0/RSP0/CPU0:router(config-l2vpn)	
Step 16	xconnect group group-name	Configures a group cross connect.
	Example:	
	RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group gl	
Step 17	p2p xconnect-name	Configures a point-to-point cross connect.
	Example:	
	RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p xc1	
Step 18	monitor-session session-name	Attaches a traffic mirroring session to the point-to-point
	Example:	cross connect.
	<pre>RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# monitor-session mon1</pre>	
Step 19	neighbor peer-ip pw-id pseudowire-id	Configures the point-to-point cross connect.
	Example:	
	RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# neighbor 192.168.2.2 pw-id 3	
Step 20	pw-class class-name	Specifies the pseudowire class template to use for the cross
	Example:	connect.
	RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# pw-class pw1	
Step 21	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	<pre>RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# end</pre>	Uncommitted changes found, commit them be exiting (yes/no/cancel)? [cancel]:
	ог	
	RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw)# commit	- Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		- Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.

	Command or Action	Purpose
		- Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.
		Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.
Step 22	show monitor-session [session-name] status [detail] [error]	Displays information about the traffic mirroring session.
	Example:	
	RP/0/RSP0/CPU0:router# show monitor-session	

How to Configure ACL-Based Traffic Mirroring

Before you begin

The global interface ACL should be configured using one of these commands with the capture keyword:

- ipv4 access-list
- ipv6 access-list
- ethernet-services access-list

For more information, refer to the *Cisco ASR 9000 Series Aggregation Services Router IP Addresses* and Services Command Reference or the ASR 9000 Series Aggregation Services Router L2 VPN and Ethernet Services Command Reference.

SUMMARY STEPS

- 1. configure
- 2. monitor-session session-name
- **3. destination interface** *dest-interface*
- 4. exit
- 5. interface source-interface
- 6. l2transport
- 7. exit
- 8. ethernet-services access-group access-list-name [ingress | egress]
- 9. monitor-session session-name [ipv4|ipv6] [direction {rx-only|tx-only}]
- 10. acl
- 11. end or commit
- 12. show monitor-session [session-name] status [detail] [error]

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	monitor-session session-name	Defines a monitor session and enters monitor session
	Example:	configuration mode.
	RP/0/RSP0/CPU0:router(config)# monitor-session	
	<pre>mon1 RP/0/RSP0/CPU0:router(config-mon)#</pre>	
Step 3	destination interface dest-interface	Specifies the destination interface to which traffic should
•	Example:	be replicated.
	PR/0/PRP0/CDU0.router(config men) # destination	
	interface gigabitethernet0/0/0/15	
Step 4	exit	Exits monitor session configuration mode and returns to
	Example:	global configuration mode.
	<pre>RP/0/RSP0/CPU0:router(config-mon)# exit RP/0/RSP0/CPU0:router(config)#</pre>	
Step 5	interface source-interface	Enters interface configuration mode for the specified
	Example:	<i>interface.</i> The interface number is entered in <i>rack/slot/module/port</i> notation. For more information about
	<pre>RP/0/RSP0/CPU0:router(config)# interface gigabitethernet0/0/0/11</pre>	the syntax for the router, use the question mark (?) online help function.
Step 6	l2transport	(Optional) Enables Layer 2 transport mode on the
	Example:	subinterface and enters Layer 2 transport configuration mode.
	RP/0/RSP0/CPU0:router(config-if)# l2transport	Note • Use the l2transport command to mirror all traffic types.
Step 7	exit	Exits Layer 2 transport configuration mode and returns to interface configuration mode.
	Example:	
	<pre>RP/0/RSP0/CPU0:router(config-if-l2)# exit RP/0/RSP0/CPU0:router(config-if)#</pre>	

I

Step 8 ethernet-services access-group access-list-name [ingress] Associates the access list definition with the interface being mirrored. Step 9 monitor-session session-name [ipv4ipv6] [direction [rx-only[tz-only]] Specifies the monitor session to be used on this interface. Step 9 monitor-session session-name [ipv4ipv6] [direction [rx-only[tz-only]] Specifies the monitor session to be used on this interface. Step 10 act Specifies that the traffic mirrored is according to the defined global interface ACL. RP/0/RSP0/CPD0:router(config-if-mon) # acl Saves configuration changes. Step 11 end or commit Saves configuration changes. RP/0/RSP0/CPD0:router(config-if) # end or router(config-if) # commit Saves configuration changes. RP/0/RSP0/CPD0:router(config-if) # commit Saves configuration changes found, connuit them before existing router (config-if) # commit RP/0/RSP0/CPD0:router(config-if) # commit Saves configuration changes. RP/0/RSP0/CPD0:router(config-if) # commit - Intering row sits the configuration session and returns the router to EXEC mode. - Intering cancel leaves the configuration session and returns the router to EXEC mode. - Entering cancel leaves the configuration session and returns the router to EXEC mode. - Intering cancel leaves the configuration session and returns the router to EXEC mode. - Entering cancel leaves the configuration session.		Command or Action	Purpose
Example: PP/0/RSP0/CSU0:router(config-1f)# step 9 monitor-session session-name [ipv4]ipv6] [direction Specifies the monitor session to be used on this interface. Step 9 monitor-session session-name [ipv4]ipv6] [direction Specifies the monitor session to be used on this interface. Step 10 acl Specifies that the traffic mirrored is according to the defined global interface ACL. RP/0/RSP0/CD0:router(config-if)# monitor-session Specifies that the traffic mirrored is according to the defined global interface ACL. Step 11 end or commit Example: Saves configuration changes. xu/0/NSP0/CD0:router(config-if)# end or Saves configuration changes. PF/0/RSP0/CD0:router(config-if)# commit Saves configuration changes. •When you issue the end command, the system prompts you to commit changes found, commit them before exiting (yes/no/cancel)? RP/0/RSP0/CD0:router(config-if)# commit Uncommit changes found, commit them before exiting (yes/no/cancel)? RP/0/RSP0/CD0:router(config-1f)# commit -Entering yes saves configuration figuration figuration session and returns the router to EXEC mode. RP/0/RSP0/CD0:router(config-1f)# commit -Entering reacel leaves the router in the current configuration figuration session and returns the router to EXEC mode without committing the configuration changes.	Step 8	ethernet-services access-group access-list-name [ingress egress]	Associates the access list definition with the interface being mirrored.
BP/0/RBP0/CPU0:router(config-if)# ethernet-services.access-group.acl1 ingress Specifies the monitor session to be used on this interface. Step 9 monitor-session session-name [ipv4[ipv6] [direction [tr-only]] Specifies the monitor session to be used on this interface. Step 10 acl Example: RP/0/RBP0/CPU0:router(config-if)# monitor-session Specifies that the traffic mirrored is according to the defined global interface ACL. Step 11 end or commit Example: RP/0/RBP0/CPU0:router(config-if)# end or RP/0/RBP0/CPU0:router(config-if)# commit Saves configuration changes. RP/0/RBP0/CPU0:router(config-if)# commit Saves configuration changes. • When you issue the end command, the system prompts you to commit changes: numit changes found, commit them before exiting (ups/no/cancel)? [cancel]: RP/0/RBP0/CPU0:router(config-if)# commit Saves configuration figuration figuration session and returns the router to EXEC mode. - Entering no exits the configuration session and returns the router to EXEC mode. - Entering mode without committing the configuration changes. Step 12 show monitor-session [session-name] status [detail] [error] Displays information about the monitor session.		Example:	
Step 9 monitor-session session-name [ipv4 ipv6] [direction [rx-only[tx-only]] Specifies the monitor session to be used on this interface. RP/0/RSP0/CPU0:router(config-if) # monitor-session mon1 direction rx-only Specifies that the traffic mirrored is according to the defined global interface ACL. Step 10 acl Example: RP/0/RSP0/CPU0:router(config-if-mon) # acl Specifies that the traffic mirrored is according to the defined global interface ACL. Step 11 end or commit Example: RP/0/RSP0/CPU0:router(config-if) # end or RP/0/RSP0/CPU0:router(config-if) # commit Saves configuration changes. Nuccessful changes found, comfit them before exiting (yes/no/cancel)? (cancel]: • When you issue the end command, the system prompts you to commit changes: • Uncommitted changes found, comfit them before exiting (yes/no/cancel)? (cancel]: RD/0/RSPD/CPU0:router(config-if) # commit • 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. • Entering cancel leaves the router in the current configuration session without exiting or committing the configuration session. Step 12 show monitor-session [session-name] status [detail] [error] Displays information about the monitor session.		<pre>RP/0/RSP0/CPU0:router(config-if)# ethernet-services access-group acl1 ingress</pre>	
Example: RP/0/RSPD/CDU0:router(config-if)# monitor-session mon1 direction rx-only Step 10 acl Example: RP/0/RSPD/CPU0:router(config-if)# monitor-session Specifies that the traffic mirrored is according to the defined global interface ACL. Step 11 end or commit Example: RP/0/RSP0/CPU0:router(config-if)# end or Saves configuration changes. RP/0/RSP0/CPU0:router(config-if)# end or Uncommited changes found, commit them before exiting (yes/no/cancel)? [cancel]? RP/0/RSP0/CPU0:router(config-if)# commit Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. - Entering yes saves configuration changes. - Entering cancel leaves the router in the current configuration changes. Step 12 show monitor-session [session-name] status [detail] [error] Displays information about the monitor session.	Step 9	monitor-session session-name [ipv4 ipv6] [direction {rx-only tx-only}]	Specifies the monitor session to be used on this interface.
BP/0/RSP0/CPU0:router(config-if)# monitor-session mon1 direction rx-only Specifies that the traffic mirrored is according to the defined global interface ACL. Step 10 acl Example: RP/0/RSP0/CPU0:router(config-if-mon)# acl Specifies that the traffic mirrored is according to the defined global interface ACL. Step 11 end or commit Example: RP/0/RSP0/CPU0:router(config-if)# end or RP/0/RSP0/CPU0:router(config-if)# commit Saves configuration changes. * When you issue the end command, the system prompts you to commit changes: RP/0/RSP0/CPU0:router(config-if)# commit Uncommitted changes found, commit them before exiting (yes/no/cancel)? (cancel): * Fintering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. - Entering nexits the configuration session and returns the router to EXEC mode. * Entering cancel leaves the router in the current configuration changes. - Entering cancel leaves the router in the current configuration changes. * Use the commit command to save the configuration session. • Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session. Step 12 show monitor-session [session-name] status [detail] [error] Displays information about the monitor session.		Example:	
Step 10 acl Specifies that the traffic mirrored is according to the defined global interface ACL. Step 11 end or commit Saves configuration changes. P/0/RSPD/CPU0:router(config-if) # end or Saves configuration changes. • When you issue the end command, the system prompts you to commit them before exiting (yes/no/cancel)? [cancel]: RP/0/RSPD/CPU0:router(config-if) # commit Uncommitted changes found, commit them before exiting (yes/no/cancel)? RP/0/RSPD/CPU0:router(config-if) # commit - 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. - Entering cancel leaves the router in the current configuration changes. Step 12 show monitor-session [session-name] status [detail] [error] Displays information about the monitor session.		<pre>RP/0/RSP0/CPU0:router(config-if)# monitor-session mon1 direction rx-only</pre>	
Example: defined global interface ACL. RP/0/RSP0/CPU0:router(config-if-mon) # acl Saves configuration changes. Step 11 end or commit Example: Saves configuration changes. RP/0/RSP0/CPU0:router(config-if) # end or Or RP/0/RSP0/CPU0:router(config-if) # commit Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: RP/0/RSP0/CPU0:router(config-if) # commit - 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. - Entering cancel leaves the router in the current configuration changes. - Entering cancel leaves the router in the current configuration changes. - Entering cancel leaves the router in the current configuration file and remain within the configuration file and remain within the configuration file and remain within the configuration session. Step 12 show monitor-session [session-name] status [detail] [error] Displays information about the monitor session.	Step 10	acl	Specifies that the traffic mirrored is according to the
RP/0/RSP0/CPU0:router(config-if-mon)# acl Step 11 end or commit Example: Saves configuration changes. RP/0/RSP0/CPU0:router(config-if)# end or or Uncommited changes found, commit them before exiting (yes/no/cancel)? [cancel]: RP/0/RSP0/CPU0:router(config-if)# commit 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. - Entering cancel leaves the router in the current configuration changes. Step 12 show monitor-session [session-name] status [detail] [error] Displays information about the monitor session.		Example:	defined global interface ACL.
Step 11 end or commit Example: Saves configuration changes. RP/0/RSPD/CPU0:router(config-if) # end • When you issue the end command, the system prompts you to commit changes: or P/0/RSP0/CPU0:router(config-if) # commit RP/0/RSP0/CPU0:router(config-if) # commit Uncommitted changes found, commit them before exiting (yes/no/cancel)? RP/0/RSP0/CPU0:router(config-if) # commit - 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. - Entering no exits the configuration session and returns the router to EXEC mode. - Entering cancel leaves the router in the current configuration changes. - Entering cancel leaves the router in the current configuration session without exiting or committing the configuration session. Step 12 show monitor-session [session-name] status [detail] [error] Example: RP/0/RSP0/CPU0:router # show monitor-session		RP/0/RSP0/CPU0:router(config-if-mon)# acl	
Example: • When you issue the end command, the system prompts you to commit changes: RP/0/RSP0/CPU0:router(config-if)# end • Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: RP/0/RSP0/CPU0:router(config-if)# commit • 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. • Entering no exits the configuration session and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode. • Entering no exits the configuration session and returns the router to EXEC mode. • Entering no 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 session without exiting or committing the configuration session without exiting or committing the configuration session. Step 12 show monitor-session [session-name] status [detail] RP/0/RSP0/CPU0:router# show monitor-session Displays information about the monitor session.	Step 11	end or commit	Saves configuration changes.
RP/0/RSP0/CPU0:router(config-if) # end Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: RP/0/RSP0/CPU0:router(config-if) # commit - 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 changes. - Use the commit command to save the configuration file and remain within the configuration session. Step 12 show monitor-session [session-name] status [detail] [error] Example: Displays information about the monitor session.		Example:	• When you issue the end command, the system prompts you to commit changes:
or Or RP/0/RSP0/CPU0:router(config-if)# commit Second provide the configuration changes found, commit the Deformation (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. - Entering no exits the configuration session and returns the router to EXEC mode. - Entering no exits the configuration session and returns the router to EXEC mode. - Entering cancel leaves the router in the current configuration changes. - Entering cancel leaves the router in the current configuration changes. - Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session. Step 12 show monitor-session [session-name] status [detail] [error] Example: RP/0/RSP0/CPU0:router# show monitor-session Displays information about the monitor session.		<pre>RP/0/RSP0/CPU0:router(config-if) # end</pre>	
RP/0/RSP0/CPU0:router(config-if)# commit - 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. Step 12 show monitor-session [session-name] status [detail] [error] Displays information about the monitor session.		or	<pre>Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:</pre>
Step 12 show monitor-session [session-name] status [detail] [error] Example: RP/0/RSP0/CPU0:router# show monitor-session		RP/0/RSP0/CPU0:router(config-if)# commit	- Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
Step 12 show monitor-session [session-name] status [detail] [error] Displays information about the monitor session. RP/0/RSP0/CPU0:router# show monitor-session - Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.			- Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
Step 12 show monitor-session [session-name] status [detail] Displays information about the monitor session. Step 12 show monitor-session [session-name] status [detail] Displays information about the monitor session. RP/0/RSP0/CPU0:router# show monitor-session Displays information about the monitor session.			- Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
Step 12 show monitor-session [session-name] status [detail] Displays information about the monitor session. [error] Example: RP/0/RSP0/CPU0:router# show monitor-session			• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.
Example: RP/0/RSP0/CPU0:router# show monitor-session	Step 12	show monitor-session [session-name] status [detail] [error]	Displays information about the monitor session.
RP/0/RSP0/CPU0:router# show monitor-session		Example:	
		RP/0/RSP0/CPU0:router# show monitor-session	

Troubleshooting ACL-Based Traffic Mirroring

Take note of these configuration issues:

- If an ACL-Based SPAN is configured on a Layer 3 interface configured with a dot1q encapsulation, the configuration will not commit.
- Even when the **acl** command is configured on the source mirroring port, if the ACL configuration command does not use the **capture** keyword, no traffic gets mirrored.
- If the ACL configuration uses the **capture** keyword, but the **acl** command is not configured on the source port, traffic is mirrored, but no access list configuration is applied.
- All ingress traffic is mirrored, regardless of the ACL definition; only egress traffic permitted in the ACL definition is mirrored.

This example shows both the **capture** keyword in the ACL definition and the **acl** command configured on the interface:

```
monitor-session tm_example
!
ethernet-services access-list tm_filter
10 deny 0000.1234.5678 0000.abcd.abcd any capture
!
interface GigabitEthernet0/2/0/0
monitor-session tm_example direction rx-only
acl
!
l2transport
!
ethernet-services access-group tm_filter ingress
end
```

How to Configure Partial Packet Mirroring

SUMMARY STEPS

- 1. configure
- 2. monitor-session session-name
- 3. destination interface dest-interface
- 4. exit
- 5. interface source-interface
- 6. monitor-session session-name[direction {rx-only | tx-only]
- 7. mirror first bytes
- 8. end or commit
- 9. show monitor-session [session-name] status
DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	monitor-session session-name	Defines a monitor session and enters monitor session
	Example:	configuration mode.
	RP/0/RSP0/CPU0:router(config)# monitor-session mon1 RP/0/RSP0/CPU0:router(config-mon)#	
Step 3	destination interface dest-interface	Specifies the destination interface to which traffic should
	Example:	be replicated.
	<pre>RP/0/RSP0/CPU0:router(config-mon)# destination interface gigabitethernet0/0/0/15</pre>	
Step 4	exit	Exits monitor session configuration mode and returns to
	Example:	global configuration mode.
	<pre>RP/0/RSP0/CPU0:router(config-mon)# exit RP/0/RSP0/CPU0:router(config)#</pre>	
Step 5	interface source-interface	Enters interface configuration mode for the specified
	Example:	<i>rack/slot/module/port</i> notation. For more information about
	<pre>RP/0/RSP0/CPU0:router(config)# interface gigabitethernet0/0/0/11.10</pre>	the syntax for the router, use the question mark (?) online help function.
Step 6	<pre>monitor-session session-name[direction {rx-only tx-only]</pre>	Specifies the monitor session to be used on this interface. Use the direction keyword to specify that only ingress or
	Example:	egress traffic is mirrored.
	<pre>RP/0/RSP0/CPU0:router(config-if-l2)# monitor-session mon1</pre>	
Step 7	mirror first bytes	Specifies the number of bytes of the packet to mirror. Values
	Example:	can range from 64 to 256.
	<pre>RP/0/RSP0/CPU0:router(config-if-mon)# mirror first bytes</pre>	
Step 8	end or commit	Saves configuration changes.
	Example:	

	Command or Action	Purpose
	RP/0/RSP0/CPU0:router(config-if)# end	• When you issue the end command, the system prompts you to commit changes:
	or	
	RP/0/RSP0/CPU0:router(config-if)# commit	Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:
		- Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		- Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		- Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.
Step 9	show monitor-session [session-name] status	Displays information about the traffic mirroring session.
	Example:	
	RP/0/RSP0/CPU0:router# show monitor-session	

Traffic Mirroring Configuration Examples

This section contains examples of how to configure traffic mirroring:

Traffic Mirroring with Physical Interfaces (Local): Example

This example shows a basic configuration for traffic mirroring with physical interfaces. When traffic flows over the point-to-point cross connect between gig0/2/0/19 and gig0/2/0/11, packets received and transmitted on gig0/2/0/19 are also mirrored to gig0/2/0/15.

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# monitor-session ms1
RP/0/RSP0/CPU0:router(config-mon)# destination interface gig0/2/0/15
RP/0/RSP0/CPU0:router(config-mon)# commit
RP/0/RSP0/CPU0:router(config)# interface gig0/2/0/11
RP/0/RSP0/CPU0:router(config-subif)# 12transport
RP/0/RSP0/CPU0:router(config-if-12)# commit
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface gig0/2/0/15
RP/0/RSP0/CPU0:router(config-subif)# 12transport
```

```
RP/0/RSP0/CPU0:router(config-if-l2) # commit
```

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface gig0/2/0/19
RP/0/RSP0/CPU0:router(config-subif)# l2transport
RP/0/RSP0/CPU0:router(config-subif-l2)# monitor-session msl
RP/0/RSP0/CPU0:router(config-if-l2)# commit
```

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# 12vpn
RP/0/RSP0/CPU0:router(config-12vpn)# xconnect group xg1
RP/0/RSP0/CPU0:router(config-12vpn-xc)# p2p xg1_p1
RP/0/RSP0/CPU0:router(config-12vpn-xc-p2p)# interface gig0/2/0/11
RP/0/RSP0/CPU0:router(config-12vpn-xc-p2p)# interface gig0/2/0/19
RP/0/RSP0/CPU0:router(config-if-12)# commit
```

Traffic Mirroring with EFPs (Remote): Example

This example shows a basic configuration for remote traffic mirroring with EFP interfaces. When traffic flows over the point-to-point cross connect between gig0/2/0/19.10 and gig0/2/0/11.10, packets received and transmitted on gig0/2/0/19.10 are also mirrored to gig0/2/0/10.1.

```
RP/0/RSP0/CPU0:router#monitor-session ms1
RP/0/RSP0/CPU0:router(config)# destination interface gig0/2/0/10.1
RP/0/RSP0/CPU0:router(config)# interface gig0/2/0/10.1 l2transport
RP/0/RSP0/CPU0:router(config-if-l2)# encapsulation dotlq 1
RP/0/RSP0/CPU0:router(config-if-l2)# rewrite ingress tag pop 1
RP/0/RSP0/CPU0:router(config)# interface gig0/2/0/11.10 l2transport
RP/0/RSP0/CPU0:router(config)# interface gig0/2/0/11.10 l2transport
RP/0/RSP0/CPU0:router(config-if-l2)# encapsulation dotlq 10
```

RP/0/RSP0/CPU0:router(config)# interface gig0/2/0/19.10 l2transport RP/0/RSP0/CPU0:router(config-if-l2)# encapsulation dotlq 10 RP/0/RSP0/CPU0:router(config-if-l2)# monitor-session ms1

```
RP/0/RSP0/CPU0:router(config)# 12vpn
RP/0/RSP0/CPU0:router(config-12vpn)# xconnect group xg1
RP/0/RSP0/CPU0:router(config-12vpn-xc)# p2p xg1_p1
RP/0/RSP0/CPU0:router(config-12vpn-xc-p2p)# interface gig0/2/0/11.10
RP/0/RSP0/CPU0:router(config-12vpn-xc-p2p)# interface gig0/2/0/19.10
```

Traffic Mirroring with Bundle Interfaces: Example

This example shows a basic configuration for traffic monitoring with bundle interfaces.

```
Router# configure
Router(config)# monitor-session ms1
Router(config-mon)# destination interface bundle-ether1
Router(config-mon)# commit
```

```
Router# configure
Router(config)# interface bundle-ether1
Router(config-if)# l2transport
Router(config-if-l2)# commit
```

Viewing Monitor Session Status: Example

This example shows sample output of the show monitor-session command with the status keyword:

RP/0/RSP0/CPU0:router# show monitor-session status

Router# show monitor-session status

RP/0/RSP0/CPU0:router# show monitor-session status detail

RP/0/RSP0/CPU0:router# show monitor-session status error

Monitor Session Statistics: Example

Use the **show monitor-session** command with the **counters** keyword to show the statistics/counters (received/transmitted/dropped) of different source ports. For each monitor session, this command displays a list of all source interfaces and the replicated packet statistics for that interface.

The full set of statistics displayed for each interface is:

- · RX replicated packets and octets
- · TX replicated packets and octets
- Non-replicated packet and octets

RP/0/RSP0/CPU0:router# show monitor-session counters

```
Monitor-session ms1
GigabitEthernet0/2/0/19.10
Rx replicated: 1000 packets, 68000 octets
Tx replicated: 1000 packets, 68000 octets
Non-replicated: 0 packets, 0 octets
```

Use the **clear monitor-session counters** command to clear any collected statistics. By default this command clears all stored statistics; however, an optional interface filter can be supplied.

RP/0/RSP0/CPU0:router# clear monitor-session counters

Traffic Mirroring over Pseudowire: Example

This example shows how to configure traffic mirroring over a pseudowire:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet0/11/0/1
RP/0/RSP0/CPU0:router(config-if)# l2transport
RP/0/RSP0/CPU0:router(config)# monitor-session pw-span-test
RP/0/RSP0/CPU0:router(config)# destination pseudowire
RP/0/RSP0/CPU0:router(config)# l2vpn
RP/0/RSP0/CPU0:router(config-l2vpn)# pw-class class1
RP/0/RSP0/CPU0:router(config-l2vpn)# encapsulation mpls
RP/0/RSP0/CPU0:router(config-l2vpn)# xconnect group g1
RP/0/RSP0/CPU0:router(config-l2vpn-xc)# p2p x1
RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# monitor-session pw-span-test
RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p)# neighbor 2.2.2.2 pw-id 1
RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-w)# pw-class class1
```

RP/0/RSP0/CPU0:router(config-l2vpn-xc-p2p-pw) # commit

Layer 3 ACL-Based Traffic Mirroring: Example

This example shows how to configure Layer 3 ACL-based traffic mirroring:

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

```
RP/0/RSP0/CPU0:router(config)# monitor-session ms1
RP/0/RSP0/CPU0:router(config-mon)# destination
RP/0/RSP0/CPU0:router(config-mon)# commit
RP/0/RSP0/CPU0:router(config)# interface gig0/2/0/11
RP/0/RSP0/CPU0:router(config-if)# ipv4 access-group span ingress
RP/0/RSP0/CPU0:router(config-if)# monitor-session ms1
RP/0/RSP0/CPU0:router(config-if-mon)# commit
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# ipv4 access-list span
RP/0/RSP0/CPU0:router(config)# ipv4 access-list span
RP/0/RSP0/CPU0:router(config-ipv4-acl)# 5 permit ipv4 any any dscp 5 capture
```

RP/0/RSP0/CPU0:router(config-ipv4-acl)# 10 permit ipv4 any any RP/0/RSP0/CPU0:router(config-ipv4-acl)# commit

Layer 2 ACL-Based Traffic Mirroring: Example

This example shows how to configure Layer 2 ACL-based traffic mirroring:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# monitor-session ms1
RP/0/RSP0/CPU0:router(config-mon)# destination interface gig0/2/0/15
RP/0/RSP0/CPU0:router(config-mon)# commit
```

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface gig0/2/0/11
RP/0/RSP0/CPU0:router(config-if)# l2transport
RP/0/RSP0/CPU0:router(config-if-12)# exit
RP/0/RSP0/CPU0:router(config-if)# ethernet-services access-group acl_mirror ingress
RP/0/RSP0/CPU0:router(config-if)# acl
RP/0/RSP0/CPU0:router(config-if)# monitor-session ms1
RP/0/RSP0/CPU0:router(config-if-mon)# commit
```

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# ipv4 access-list acl_mirror
RP/0/RSP0/CPU0:router(config-ipv4-acl)# 5 permit ipv4 any any dscp 5 capture
RP/0/RSP0/CPU0:router(config-ipv4-acl)# 10 permit ipv4 any any
RP/0/RSP0/CPU0:router(config-ipv4-acl)# commit
```

Partial Packet Mirroring: Example

This example shows how to configure mirroring of the first 100 bytes of the packet:

```
RP/0/RP0/CPU0:router(config)# interface gigabitethernet0/0/0/11
RP/0/RP0/CPU0:router(config-if-l2)# monitor-session mon1
RP/0/RSP0/CPU0:router(config-if-mon)# mirror first 100
```

Sampled Traffic Mirroring: Example

This example shows how to configure Sampled Traffic Mirroring:

Destination Port

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:(config)# interface TenGigE 0/3/1/3
```

```
RP/0/RSP0/CPU0:(config-if)# l2transport
RP/0/RSP0/CPU0:(config-if-l2)# commit
RP/0/RSP0/CPU0:(config)# monitor-session sampled-span1
RP/0/RSP0/CPU0:(config-mon)# destination interface TenGigE 0/3/1/3
RP/0/RSP0/CPU0:(config-mon)# commit
```

Source Port

```
RP/0/RSP0/CPU0:(config)# interface TenGigE 0/3/0/0
RP/0/RSP0/CPU0:(config-if)# l2transport
RP/0/RSP0/CPU0:(config-if-l2)# monitor-session sampled-span1 direction rx-only port-level
RP/0/RSP0/CPU0:(config-if-mon)# mirror interval 512
RP/0/RSP0/CPU0:(config-if-mon)# commit
```

In order to display the session status with the Sampled SPAN information, use the **show monitor-session status detail** command.

```
RP/0/RSP0/CPU0 # show monitor-session status detail
Monitor-session sampled-span1
Destination interface TenGigE0/3/1/3
Source Interfaces
------
TenGigE0/3/0/0
Direction: Rx-only
Port level: True
ACL match: Disabled
Portion: Full packet
Interval: 512
Status: Operational
```

In order to display the session statistics, use the **show monitor-session counters** command.

```
RP/0/RSP0/CPU0:router# show monitor-session counters
Monitor-session sampled-span1
TenGigE0/3/0/0
Rx replicated: 1952 packets, 390400 octets
Tx replicated: 0 packets, 0 octets
Non-replicated: 0 packets, 0 octets
```

Troubleshooting Traffic Mirroring

When you encounter any issue with traffic mirroring, begin troubleshooting by checking the output of the **show monitor-session status** command. This command displays the recorded state of all sessions and source interfaces:

```
Monitor-session sess1

<Session status>

Source Interface Dir Status

Gi0/0/0/0 Both <Source interface status>

Gi0/0/0/2 Both <Source interface status>
```

I

Session Status	Explanation
Session is not configured globally	The session does not exist in global configuration. Check show run command output to ensure that a session with a correct name has been configured.
Destination interface <intf> is not configured</intf>	The interface that has been configured as the destination does not exist. For example, the destination interface may be configured to be a VLAN subinterface, but the VLAN subinterface may not have been yet created.
Destination interface <intf> (<down-state>)</down-state></intf>	The destination interface is not in Up state in the Interface Manager. You can verify the state using the show interfaces command. Check the configuration to see what might be keeping the interface from coming up (for example, a sub-interface needs to have an appropriate encapsulation configured).
Destination pseudowire is not configured	The L2VPN configuration that is to set up the pseudowire is missing. Configure the traffic mirroring session name as one segment of the xconnect p2p.
Destination pseudowire <name> (down)</name>	The pseudowire is configured, but is down. Check the L2VPN configuration to identify why the pseudowire is not coming up.

In the preceding example, the line marked as <session status> can indicate one of these configuration errors:

The <Source interface status> can report these messages:

Source Interface Status	Explanation
Operational	Everything appears to be working correctly in traffic mirroring PI. Please follow up with the platform teams in the first instance, if mirroring is not operating as expected.
Not operational (Session is not configured globally)	The session does not exist in global configuration. Check the show run command output to ensure that a session with the right name has been configured.
Not operational (destination interface not known)	The session exists, but it either does not have a destination interface specified, or the destination interface named for the session does not exist (for example, if the destination is a sub-interface that has not been created).
Not operational (source same as destination)	The session exists, but the destination and source are the same interface, so traffic mirroring does not work.
Not operational (destination not active)	The destination interface or pseudowire is not in the Up state. See the corresponding <i>Session status</i> error messages for suggested resolution.

Source Interface Status	Explanation
Not operational (source state <down-state>)</down-state>	The source interface is not in the Up state. You can verify the state using the show interfaces command. Check the configuration to see what might be keeping the interface from coming up (for example, a sub-interface needs to have an appropriate encapsulation configured).
Error: see detailed output for explanation	Traffic mirroring has encountered an error. Run the show monitor-session status detail command to display more information.

The **show monitor-session status detail** command displays full details of the configuration parameters, and of any errors encountered. For example:

RP/0/RSP0 show monitor-session status detail

```
Monitor-session sess1
 Destination interface is not configured
 Source Interfaces
 _____
 GigabitEthernet0/0/0/0
 Direction: Both
 ACL match: Enabled
 Portion: Full packet
 Status: Not operational (destination interface not known)
GigabitEthernet0/0/0/2
 Direction: Both
 ACL match: Disabled
 Portion: First 100 bytes
 Status: Not operational (destination interface not known). Error: 'Viking SPAN PD' detected
the 'warning' condition 'PRM connection creation failure'.
Monitor-session foo
Destination next-hop GigabitEthernet 0/0/0/0
 Source Interfaces
  _____
 GigabitEthernet 0/1/0/0.100:
 Direction: Both
 Status: Operating
GigabitEthernet 0/2/0/0.200:
 Direction: Tx
 Status: Error: <blah>
Monitor session bar
No destination configured
Source Interfaces
  _____
 GigabitEthernet 0/3/0/0.100:
 Direction: Rx
 Status: Not operational (no destination)
```

Here are additional trace and debug commands:

```
RP/0/RSP0/CPU0:router# show monitor-session platform trace ?
all Turn on all the trace
errors Display errors
events Display interesting events
```

```
RP/0/RSP0/CPU0:router# show monitor-session trace ?
process Filter debug by process
RP/0/RSP0/CPU0:router# debug monitor-session platform ?
 all
      Turn on all the debugs
errors VKG SPAN EA errors
 event VKG SPAN EA event
 info VKG SPAN EA info
RP/0/RSP0/CPU0:router# debug monitor-session platform all
RP/0/RSP0/CPU0:router# debug monitor-session platform event
RP/0/RSP0/CPU0:router# debug monitor-session platform info
RP/0/RSP0/CPU0:router# show monitor-session status ?
detail Display detailed output
errors Display only attachments which have errors
 internal Display internal monitor-session information
      Output Modifiers
 RP/0/RSP0/CPU0:router# show monitor-session status
RP/0/RSP0/CPU0:router# show monitor-session status errors
```

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 on the Cisco ASR 9000 Series Router module 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.

SPAN to File

Starting with Cisco IOS XR Software Release 7.1.2, the SPAN to File is an extension of the pre-existing SPAN feature to allow network packets to be mirrored to a file instead of an interface, so that they can be analyzed later. The file format is pcap, so that it can be easily used with tools such as tcpdump or wireshark.



Note SPAN to File supports 100 local SPAN sessions, each supporting 1 destination file and 100 source interfaces. A maximum of 1000 source ports are supported across the system. Individual platforms may support lower numbers. The SPAN session may be any of these currently supported classes: Ethernet, IPv4, IPv6, MPLS-IPv4, and MPLS-IPv6. When a file is configured as a destination for a SPAN session, a buffer is created on each node to which the network packets are logged. The buffer is for all packets on the node regardless of which interface they are from, that is, multiple interfaces may be providing packets for the same buffer. The buffers are deleted when the session configuration is removed. The file is written by each node to a location on the active RP which contains the node ID of the node on which the buffer was located.

If multiple interfaces are attached to a session, then interfaces on the same node are expected to have their packets sent to the same file. Bundle interfaces can be attached to a session with a file destination, which is similar to attaching individual interfaces. For example, this would result in a file on each node on which the bundle has members which contains packets from all members on that node.



Note

The maximum traffic rate that can be mirrored for any SPAN (ingress or egress) is limited to 10% of the line rate. Exceeding this threshold may result in the loss of transit traffic.

Guidelines and Restrictions for SPAN to File

- SPAN to File feature is designed for low traffic rate (Maximum: 500 Mbps). SPAN to File may not capture all the frames when SPAN is applied on higher traffic rate.
- If the SPAN traffic exceeds 10% of the line rate, it may lead to a loss of transit traffic.
- To limit the SPAN traffic, use SPAN along with ACL.
- Immediately after the SPAN to File configuration is applied, the network processor begins the SPAN processing. This processing continues regardless of the status of the action commands explained in the following section.

Action Commands for SPAN to File

Action commands are added to start and stop network packet collection. The commands may only be run on sessions where the destination is a file. The action command auto-completes names of globally configured SPAN to File sessions. See the table below for more information on action commands.

Action	Command	Description
Start	monitor-session <name> packet-collection start</name>	Issue this command to start writing packets for the specified session to the configured buffer.

Table 15: Action Commands for SPAN to File

Action	Command	Description
Stop	<pre>monitor-session <name> packet-collection stop [discard-data write directory <dir> filename <filename>]</filename></dir></name></pre>	Issue this command to stop writing packets to the configured buffer. If the discard-data option is specified, the buffer is simply cleared, whereas if the write option is specified, the buffer is written to disk before clearing. If the buffer is to be written, it is done so in .pcap format to this location: / <directory>/<node_id>/<filename>.pcap. If the user adds a .pcap extension when specifying the filename, this is removed so that the extension is not added twice.</filename></node_id></directory>

Configuring SPAN to File

Use the following command to configure SPAN to File:

```
monitor-session <name> [ethernet|ipv4|ipv6|mpls-ipv4|mpls-ipv6]
    destination file [size <kbytes>] [buffer-type linear]
```

The monitor-session <name> [ethernet|ipv4|ipv6|mpls-ipv4|mpls-ipv6] part of the command creates a monitor-session with the specified name and class and is a pre-existing chain point from the current SPAN feature. The destination file [size <kbytes>] [buffer-type linear] part of the command adds a new "file" option to the existing "destination".

destination file has the following configuration options:

- Buffer size.
- · Two types of buffer:
 - Circular: Once the buffer is full, the start is overwritten.
 - Linear: Once the buffer is full, no further packets are logged.



Note The default buffer-type is circular. Only linear buffer is explicitly configurable. Changing any of the parameters (buffer size or type) recreates the session, and clears any buffers of packets.

All configuration options which are applied to an attachment currently supported for other SPAN types should also be supported by SPAN to file. This may include:

- ACLs
- Write only first X bytes of packet.
- Mirror interval from 512 to 16k.

Note These options are implemented by the platform when punting the packet.

Once a session has been created, then interfaces may be attached to it using the following configuration:

```
interface GigabitEthernet 0/0/0/0
    monitor-session <name> [ethernet|ipv4|ipv6|mpls-ipv4|mpls-ipv6]
```

The attachment configuration is unchanged by SPAN to File feature.

Configuration Examples

To configure a mon1 monitor session, use the following commands:

```
monitor-session mon1 ethernet
destination file size 230000
```

In the above example, omitting the buffer-type option results in default circular buffer.

To configure a mon2 monitor session, use the following commands:

```
monitor-session mon2 ethernet
    destination file size 1000 buffer-type linear
!
```

To attach monitor session to a physical or bundle interface, use the following commands:

```
RP/0/RSP0/CPU0:router#show run interface Bundle-Ether 1
Fri Apr 24 12:12:59.348 EDT
interface Bundle-Ether1
monitor-session ms7 ethernet
'
```

Running Configuration

```
!! IOS XR Configuration 7.1.1.124I
!! Last configuration change at Tue Nov 26 19:29:05 2019 by root
!
hostname OC
logging console informational
!
monitor-session mon1 ethernet
destination file size 230000 buffer-type circular
!
monitor-session mon2 ethernet
destination file size 1000 buffer-type linear
!
interface Bundle-Ether1
monitor-session ms7 ethernet
end
```

Verification

To verify monitor session counters:

```
RP/0/RP0/CPU0:router#show monitor-session counters
Monitor-session mon2
Bundle-Ether2.1
Rx replicated: 9463555 packets, 1287043296 octets
Tx replicated: 0 packets, 0 octets
Non-replicated: 0 packets, 0 octets
```

```
Monitor-session mon4
```

```
Bundle-Ether1.1
Rx replicated: 0 packets, 0 octets
Tx replicated: 9732869 packets, 1284738317 octets
Non-replicated: 0 packets, 0 octets
```

To verify packet collection status:

RP/0/RP0/CPU0:router#show monitor-session status Monitor-session mon1 Destination File - Packet collecting _____ Source Interface Dir Status _____ ____ Rx Operational Hu0/9/0/2 Monitor-session mon2 Destination File - Packet collecting _____ Source Interface Dir Status _____ _ _____ BE2.1 Rx Operational

If packet collection is not active, the following line is displayed:

Monitor-session mon2 Destination File - Not collecting

Introduction to File Mirroring

Prior to Cisco IOS XR Software Release 7.1.2, the router did not support file mirroring from active RP to standby RP. Administrators had to manually perform the task or use EEM scripts to sync files across active RP and standby RP. Starting with Cisco IOS XR Software Release 7.1.2, file mirroring feature enables the router to copy files or directories automatically from /harddisk:/mirror location in active RP to /harddisk:/mirror location in standby RP or RSP without user intervention or EEM scripts.

Two new CLIs have been introduced for the file mirroring feature:

• mirror enable

The /harddisk:/mirror directory is created by default, but file mirroring functionality is only enabled by executing the mirror enable command from configuration terminal. Status of the mirrored files can be viewed with show mirror status command.

• mirror enable checksum

The mirror enable checksum command enables MD5 checksum across active to standby RP to check integrity of the files. This command is optional.

Limitations

The following limitations apply to file mirroring:

- Supported only on Dual RP systems.
- Supports syncing only from active to standby RP. If files are copied into standby /harddisk:/mirror location, it won't be synced to active RP.

- A slight delay is observed in show mirror command output when mirror checksum configuration is enabled.
- Not supported on multichassis systems.

Configure File Mirroring

File mirroring has to be enabled explicitly on the router. It is not enabled by default.

RP/0/RSP0/CPU0:router#**show run mirror**

Thu Jun 25 10:12:17.303 UTC mirror enable mirror checksum

Following is an example of copying running configuration to harddisk:/mirror location:

```
RP/0/RSP0/CPU0:router#copy running-config harddisk:/mirror/run_config
Wed Jul 8 10:25:51.064 PDT
Destination file name (control-c to abort): [/mirror/run_config]?
Building configuration..
32691 lines built in 2 seconds (16345)lines/sec
[OK]
```

Verification

To verify the syncing of file copied to mirror directory, use the show mirror command.

If checksum is disabled, show mirror command displays the following output:

If there is a mismatch during the syncing process, use show mirror mismatch command to verify.

```
RP/0/RP0/CPU0:router# show mirror mismatch
Wed Jul 8 10:31:21.644 PDT
MIRROR DIR: /harddisk:/mirror/
% Last sync of this dir ended at Wed Jul 8 10:31:11 2020
Location |Mismatch Reason |Action Needed
```

test.txt |newly created item. |send to standby



Configuring Virtual Loopback and Null Interfaces

This module describes the configuration of loopback and null interfaces. Loopback and null interfaces are considered virtual interfaces.

A virtual interface represents a logical packet switching entity within the router. Virtual interfaces have a global scope and do not have an associated location. Virtual interfaces have instead a globally unique numerical ID after their names. Examples are Loopback 0, Loopback 1, and Loopback 99999. The ID is unique per virtual interface type to make the entire name string unique such that you can have both Loopback 0 and Null 0.

Loopback and null interfaces have their control plane presence on the active route switch processor (RSP). The configuration and control plane are mirrored onto the standby RSP and, in the event of a failover, the virtual interfaces move to the ex-standby, which then becomes the newly active RSP.

Feature History for Configuring Loopback and Null Interfaces on Cisco IOS XR Software

Release	Modification
Release 3.7.2	This feature was introduced.

- Prerequisites for Configuring Virtual Interfaces, on page 355
- Information About Configuring Virtual Interfaces, on page 355
- How to Configure Virtual Interfaces, on page 357
- Configuration Examples for Virtual Interfaces, on page 361

Prerequisites for Configuring Virtual Interfaces

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

Information About Configuring Virtual Interfaces

To configure virtual interfaces, you must understand the following concepts:

Virtual Loopback Interface Overview

A virtual loopback interface is a virtual interface with a single endpoint that is always up. Any packet transmitted over a virtual loopback interface is immediately received by the selfsame interface. Loopback interfaces emulate a physical interface.

In Cisco IOS XR software virtual loopback interfaces perform the following 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 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 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 unreachables** command. With the **ipv4 unreachables** 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.

The Null 0 interface is created by default on the RSP during boot and cannot be removed. The **ipv4 unreachables** command can be configured for this interface, but most configuration is unnecessary because this interface just discards all the packets sent to it.

The Null 0 interface can be displayed with the show interfaces null0 command.

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 RSP is active. An IPv4 virtual address persists across route switch processor (RSP) failover situations. For this to happen, the virtual IPv4 address must share a common IPv4 subnet with a management Ethernet interface on both RPs.

On a Cisco ASR 9000 Series Router where each RSP has multiple management Ethernet interfaces, the virtual IPv4 address maps to the management Ethernet interface on the active RSP that shares the same IP subnet.

Active and Standby RPs and Virtual Interface Configuration

The standby RSP is available and in a state in which it can take over the work from the active RSP should that prove necessary. Conditions that necessitate the standby RSP to become the active RSP and assume the active RSP's duties include:

- · Failure detection by a watchdog
- Administrative command to take over
- · Removal of the active RSP from the chassis

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

After failover, the virtual interfaces all are present on the standby (now active) RSP. 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 RSP.



Note

P 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. It must not be used by another interface on the router, and it must not be used by an interface on any other router on the network.

SUMMARY STEPS

- 1. configure
- 2. interface loopback instance
- 3. ipv4 address ip-address
- 4. end or commit
- 5. show interfaces type instance

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	interface loopback instance	Enters interface configuration mode and names the new
	Example:	loopback interface.
	RP/0/RSP0/CPU0:router#(config)# interface Loopback 3	
Step 3	ipv4 address ip-address	Assigns an IP address and subnet mask to the virtual
	Example:	loopback interface using the ipv4 address configuration command.
	RP/0/RSP0/CPU0:router(config-if)# ipv4 address 172.18.189.38/32	
Step 4	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts
	RP/0/RSP0/CPU0:router(config-if)# end	you to commit changes:
	or	Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:
	RP/0/RSP0/CPU0:router(config-if)# commit	• 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 type instance	(Optional) Displays the configuration of the loopback
	Example:	interface.
	RP/0/RSP0/CPU0:router# show interfaces Loopback 3	

Configuring Null Interfaces

This task explains how to configure a basic null interface.

SUMMARY STEPS

- **1**. configure
- 2. interface null 0
- 3. end or commit
- 4. show interfaces null 0

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	interface null 0	Enters the null 0 interface configuration mode.
	Example:	
	<pre>RP/0/RSP0/CPU0:router#(config)# interface null 0</pre>	
Step 3	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	RP/0/RSP0/CPU0:router(config-null0)# end	
	or	Uncommitted changes found, commit them before exiting(yes/no/cancel)?
	<pre>RP/0/RSP0/CPU0:router(config-null0)# commit</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.
		1

I

	Command or Action	Purpose
Step 4	show interfaces null 0	Verifies the configuration of the null interface.
	Example:	
	RP/0/RSP0/CPU0:router# show interfaces null 0	

Configuring Virtual IPv4 Interfaces

This task explains how to configure an IPv4 virtual interface.

SUMMARY STEPS

- 1. configure
- 2. ipv4 address virtual address ipv4-
- 3. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	ipv4 address virtual address <i>ipv4-</i>	Defines an IPv4 virtual address for the management Ethernet
	Example:	interface.
	<pre>RP/0/RSP0/CPU0:router(config)# ipv4 virtual address 10.3.32.154/8</pre>	
Step 3	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	RP/0/RSP0/CPU0:router(config-null0)# end	
	or	Uncommitted changes found, commit them be exiting(yes/no/cancel)? [cancel]:
	RP/0/RSP0/CPU0:router(config-null0)# commit	• Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.

 Command or Action	Purpose
	• 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 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/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface Loopback 3
RP/0/RSP0/CPU0:router(config-if)# ipv4 address 172.18.189.38/32
RP/0/RSP0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes
RP/0/RSP0/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
6 minute output rate 0 bits/sec, 0 packets/sec
7 minute output rate 0 bits/sec, 0 packets/sec
8 minute output rate 0 bits/sec, 0 packets/sec
9 mackets input, 0 bytes, 0 total input drops
9 drops for unrecognized upper-level protocol
8 Received 0 broadcast packets, 0 multicast packets
9 mackets output, 0 bytes, 0 total output drops
9 Output 0 broadcast packets, 0 multicast packets
```

Configuring a Null Interface: Example

The following example indicates how to configure a null interface:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface Null 0
RP/0/RSP0/CPU0:router(config-null0)# ipv4 unreachables
RP/0/RSP0/CPU0:router(config-null0)# end
Uncommitted changes found, commit them? [yes]: yes
RP/0/RSP0/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 0 drops for upt, 0 bytes, 0 total output drops

Configuring a Virtual IPv4 Interface: Example

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



Configuring Dense Wavelength Division Multiplexing Controllers

This module describes the configuration of dense wavelength division multiplexing (DWDM) controllers.

DWDM is an optical technology that is used to increase bandwidth over existing fiber-optic backbones. DWDM can be configured on supported 10-Gigabit Ethernet (GE) line cards. After you configure the DWDM controller, you can configure an associated 10-Gigabit Ethernet interface.

Release	Modification
Release 3.9.0	This feature was introduced on the Cisco ASR 9000 Series Router on the following cards:
	• Cisco 8-Port 10 Gigabit Ethernet Line Card (A9K-8T-L and -E)
	• Cisco 2-port 10 Gigabit Ethernet + 20-port Gigabit Ethernet Combination Line Card (A9K-2T20GE-L)
Release 3.9.1	Support for the following cards was added:
	• Cisco 8-Port 10 Gigabit Ethernet Line Card (A9K-8T-B)
	• Cisco 2-port 10 Gigabit Ethernet + 20-port Gigabit Ethernet Combination Line Card (A9K-2T20GE-B and -E)
Release 4.0.0	Support for IPoDWDM Proactive Protection was added on these cards:
	• Cisco 8-Port 10 Gigabit Ethernet Line Card (A9K-8T-L, -B, and -E)
	• Cisco 2-port 10 Gigabit Ethernet + 20-port Gigabit Ethernet Combination Line Card (A9K-2T20GE-L, -B, and -E)
Release 4.2.1	Support for IPoDWDM Proactive Protection was added on these Module Port Adaptors:
	• A9K-MPA-4x10GE
	• A9K-MPA-2X10GE

Feature History for Configuring DWDM Controller Interfaces

Release 4.2.3	Support for IPoDWDM Proactive Protection was added on these Module Port Adaptors:
	• A9K-MPA-2X40GE
	• A9K-MPA-1X40GE

- Configuring Dense Wavelength Division Multiplexing Controllers, on page 364
- Prerequisites for Configuring DWDM Controller Interfaces, on page 365
- Information About the DWDM Controllers, on page 365
- CFP2 DCO Optics Version Compatibility, on page 366
- Information about IPoDWDM, on page 367
- How to Configure DWDM Controllers, on page 368
- How to Perform Performance Monitoring on DWDM Controllers, on page 373
- Configuring IPoDWDM, on page 378
- Configuration Examples, on page 383

Configuring Dense Wavelength Division Multiplexing Controllers

This module describes the configuration of dense wavelength division multiplexing (DWDM) controllers.

DWDM is an optical technology that is used to increase bandwidth over existing fiber-optic backbones. DWDM can be configured on supported 10-Gigabit Ethernet (GE) line cards. After you configure the DWDM controller, you can configure an associated 10-Gigabit Ethernet interface.

Release	Modification
Release 3.9.0	This feature was introduced on the Cisco ASR 9000 Series Router on the following cards:
	• Cisco 8-Port 10 Gigabit Ethernet Line Card (A9K-8T-L and -E)
	• Cisco 2-port 10 Gigabit Ethernet + 20-port Gigabit Ethernet Combination Line Card (A9K-2T20GE-L)
Release 3.9.1	Support for the following cards was added:
	• Cisco 8-Port 10 Gigabit Ethernet Line Card (A9K-8T-B)
	• Cisco 2-port 10 Gigabit Ethernet + 20-port Gigabit Ethernet Combination Line Card (A9K-2T20GE-B and -E)
Release 4.0.0	Support for IPoDWDM Proactive Protection was added on these cards:
	• Cisco 8-Port 10 Gigabit Ethernet Line Card (A9K-8T-L, -B, and -E)
	• Cisco 2-port 10 Gigabit Ethernet + 20-port Gigabit Ethernet Combination Line Card (A9K-2T20GE-L, -B, and -E)

Feature History for Configuring DWDM Controller Interfaces

Release 4.2.1	Support for IPoDWDM Proactive Protection was added on these Module Port Adaptors:
	• A9K-MPA-4x10GE
	• A9K-MPA-2X10GE
Release 4.2.3	Support for IPoDWDM Proactive Protection was added on these Module Port Adaptors:
	• A9K-MPA-2X40GE
	• A9K-MPA-1X40GE
1	

Prerequisites for Configuring DWDM Controller 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 a DWDM controller, be sure that you have installed one of these cards that support DWDM:

- Cisco 8-Port 10 Gigabit Ethernet Line Card
- Cisco 2-port 10 Gigabit Ethernet + 20-port Gigabit Ethernet Combination Line Card

Information About the DWDM Controllers

DWDM support in Cisco IOS XR software is based on the Optical Transport Network (OTN) protocol that is specified in ITU-T G.709. This standard combines the benefits of SONET/SDH technology with the multiwavelength networks of DWDM. It also provides for forward error correction (FEC) that can allow a reduction in network costs by reducing the number of regenerators used.



Note Configuring two ends of an OTN link with different FEC modes is not supported. Even if different FEC modes are configured, the FEC mismatch alarm will not be raised. Interface may experience continuous port flap in addition to continuous bit interleaved parity (BIP) errors at both OTN and LAN level.

To enable multiservice transport, OTN uses the concept of a wrapped overhead (OH). To illustrate this structure:

- Optical channel payload unit (OPU) OH information is added to the information payload to form the OPU. The OPU OH includes information to support the adaptation of client signals.
- Optical channel data unit (ODU) OH is added to the OPU to create the ODU. The ODU OH includes information for maintenance and operational functions to support optical channels.
- Optical channel transport unit (OTU) OH together with the FEC is added to form the OTU. The OTU OH includes information for operational functions to support the transport by way of one or more optical channel connections.

• Optical channel (OCh) OH is added to form the OCh. The OCh provides the OTN management functionality and contains four subparts: the OPU, ODU, OTU, and frame alignment signal (FAS). See figure below.

Figure 34: OTN Optical Channel Structure

		0	OTU	FAS
FEC	Payload	OPU	วบ	O
FEC	Payload	В	JU	O

CFP2 DCO Optics Version Compatibility

There are two hardware versions of the CFP2 DCO optics (A0 and B0). The following table shows hardware version compatibility.

You can identify the version using show inventory command and looking at the version ID (VID):

V01 = A0

V02 = B0

Table 16: Hardware Version Compatibility

FEC Mode	CFP2 DCO Versions	Supported
15sdfec	A0 -> A0 / B0 -> B0	Yes
	A0 -> B0 / B0 -> A0	Yes
15sdfecde	A0 -> A0 / B0 -> B0	Yes
	A0 -> B0 / B0 -> A0	Yes
Staircase FEC	A0 -> A0 / B0 -> B0	Yes
	A0 -> B0 / B0 -> A0	No
	B0 -> A9K-400G-DWDM-TR	Yes
	A0 -> A9K-400G-DWDM-TR	No

Note

- A9K-400G-DWDM-TR (2x100G (CFP2) + 20x10GE (SFP+) Combination IPoDWDM) supports 100G ACO optics (ONS-CFP2-WDM).
 - A9K-MPA-2X100GE (2x100G CFP2) supports CFP2 DCO, but only at 100G.
 - CFP2 DCO A0 version optics do not support 100G, 7% Staircase FEC mode as per the required standard and therefore is non-compatible with CFP2 DCO B0 version and A9K-400G-DWDM-TR (2x100G (CFP2) + 20x10GE (SFP+) Combination IPoDWDM) ACO Optics.

Information about IPoDWDM

Cisco IOS XR software includes the IP over Dense Wavelength Division Multiplexing (IPoDWDM) feature.

IPoDWDM is supported on these hardware devices:

- Cisco 8-Port 10 Gigabit Ethernet Line Card
- Cisco 2-port 10 Gigabit Ethernet + 20-port Gigabit Ethernet Combination Line Card



```
Note
```

The **ONS-C2-WDM-DE-1HL** Line Card supports only the OTN mode. To make **ONS-C2-WDM-DE-1HL** work in the OTN mode, use the following configuration:

```
Set port to OTN mode
(config)#controller dwdm 0/2/0/0
(config-dwdm)#g709 enable
(config-dwdm)#admin-state in-service
(config-dwdm)#commit
```

IPoDWDM currently provides these software features:

Proactive Maintenance

Proactive Maintenance

Proactive maintenance automatically triggers Forward Error Correction-Fast Re-Route (FEC-FRR). Proactive maintenance requires coordinated maintenance between Layer 0 (L0) and Layer 3 (L3). L0 is the DWDM optical layer. FEC-FRR is an L3 protection mechanism. FEC-FRR detects failures before they happen and corrects errors introduced during transmission or that are due to a degrading signal.

Shared Risk Link Group (SRLG)

System administrators can configure the following IPoDWDM features:

- Optical Layer DWDM port, see Configuring the Optical Layer DWDM Ports.
- Administrative state of DWDM optical ports, see Configuring the Administrative State of DWDM Optical Ports.
- FEC-FRR trigger threshold, window size, revert threshold, and revert window size, see Configuring Proactive FE-FRR Triggering.

FEC-FRR Triggering

FEC-FRR can be configure to be triggered by the following alarms:

- ais Alarm Indication Signal (AIS)
- bdi Backward Defect Indication (BDI)
- *bdiO Backward Defect Indication Overhead (BDI-O)
- *bdiP Backward Defect Indication Payload (BDI-P)

- *deg Degraded (DEG)
- lck Locked (LCK)
- lof Loss of Frame (LOF)
- lom Loss of Multi Frame
- los Loss of Signal (LOS)
- *losO Loss of Signal Overhead (LOS-O)
- *losP Loss of Signal Payload (LOS-P)
- oci Open Connection Indication (OCI)
- plm Payload Mismatch (PLM)
- *ssf Server Signal Failure (SSF)
- *ssfO Server Signal Failure Overhead (SSF-O)
- *ssfP Server Signal Failure Payload (SSF-P)
- tim Trace Identifier Mismatch (TIM)

Signal Logging

DWDM statistic data, such as EC, UC and alarms, are collected and stored in the log file on the DWDM line card.

How to Configure DWDM Controllers

The DWDM controllers are configured in the physical layer control element of the Cisco IOS XR software configuration space. This configuration is done using the **controller dwdm** command, and is described in the following task:



All interface configuration tasks for Gigabit Ethernet interfaces still must be performed in interface configuration mode.

Configuring the Optical Parameters

This task describes how to configure the receive power threshold and the wavelength parameters for the DWDM controller. You should verify that the optical parameters are configured correctly for your DWDM installation and if necessary, perform this task.

Before you begin

The **rx-los-threshold**, **wavelength** and **transmit-power** commands can be used only when the controller is in the shutdown state. Use the **shutdown** command.

Restrictions

The transmit power level and receive LOS threshold are configurable only on the Cisco Cisco 1-Port OC-768c/STM-256c DWDM PLIM.

SUMMARY STEPS

- 1. configure
- **2. controller dwdm** *interface-path-id*
- **3**. admin-state {maintenance | out-of-service}
- 4. commit
- 5. rx-los-threshold power-level
- **6.** wavelength *frequency-grid channel-number*
- 7. transmit-power power-level
- 8. end or commit
- 9. admin-state in-service
- **10.** show controllers dwdm *interface-path-id* [optics | wavelength-map]

DETAILED STEPS

Procedure

	Command or Action	Purpose	
Step 1	configure	Enters global configuration mode.	
	Example:		
	RP/0/RP0/CPU0:Router# configure		
Step 2	controller dwdm interface-path-id	Specifies the DWDM controller name in the notation	
	Example:	<i>rack/slot/module/port</i> and enters DWDM configuration mode.	
	RP/0/RP0/CPU0:Router(config)# controller dwdm 0/1/0/0		
Step 3	admin-state {maintenance out-of-service}	Specifies the DWDM interface administrative state. You	
	Example:	must put the controller in maintenance or out-of-service state before you can use the DWDM configuration	
	RP/0/RP0/CPU0:Router(config-dwdm)# admin-state maintenance	commands.	
Step 4	commit	Saves configuration changes. This performs the shutdown	
	Example:	from the previous step. When the controller has been sh down, you can proceed with the configuration.	
	<pre>RP/0/RP0/CPU0:Router(config-dwdm) # commit</pre>		
Step 5	rx-los-threshold power-level	Configures the transponder receive power threshold.	
	Example:	Values are in units of 0.1 dBm and can range from -350 to 50. This corresponds to a range of -35 dBm to 5 dBm.	
	RP/0/RP0/CPU0:Router(config-dwdm)# rx-los-threshold -10		

	Command or Action	Purpose
Step 6	<pre>wavelength frequency-grid channel-number Example: RP/0/RP0/CPU0:Router(config-dwdm)# wavelength 50GHz-Grid 1</pre>	 Configures the channel number corresponding to the first wavelength. Values can range from 1 to 185, but not all channels are supported on all PLIMs. Use the show controller dwdm command with the wavelength-map keyword to determine which channels and wavelengths are supported on a specific controller. Note There is no cross-checking to determine if the chosen wavelength is being used on another port on the same PLIM or on another PLIM in the system.
Step 7	<pre>transmit-power power-level Example: RP/0/RP0/CPU0:Router(config-dwdm)# transmit-power 10</pre>	Configures the transponder transmit power. Values are in units of 0.1 dBm and can range from -190 to +10. This corresponds to a range of -19 dBm to +1 dBm.
Step 8	<pre>end or commit Example: RP/0/RP0/CPU0:Router(config-dwdm)# end or RP/0/RP0/CPU0:Router(config-dwdm)# commit</pre>	 Saves configuration changes. When you issue the end command, the system prompts you to commit changes: Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.
Step 9	<pre>admin-state in-service Example: RP/0/RP0/CPU0:Router(config-dwdm)# admin-state in-service</pre>	Places the DWDM port in In Service (IS) state, to support all normal operation.

	Command or Action	Purpose
Step 10	<pre>show controllers dwdm interface-path-id [optics wavelength-map]</pre>	Displays the output power level, input power level, wavelength, and laser bias current monitoring information.
	Example:	
	RP/0/RP0/CPU0:Router# show controller dwdm 0/1/0/0 optics	

Troubleshooting Tips

L

You will get an error message if you try to commit configuration changes to the controller when it is in the up state. You must use the **admin-states maintenance** or **admin-states out-of-service** command before you can use the DWDM configuration commands.

Configuring G.709 Parameters

Before you begin

The , and **g709 fec** commands can be used only when the controller is in the shutdown state. Use the **admin-state** command.

SUMMARY STEPS

- 1. configure
- 2. controller dwdm interface-path-id
- 3. admin-state maintenance or admin-state out-of-service
- 4. commit
- 5. g709 disable
- 6. loopback {internal | line}
- 7. g709 fec {disable | standard}
- 8. g709 {odu | otu} report *alarm* disable
- 9. g709 otu overhead tti {expected | sent} {ascii | hex} tti-string
- 10. end or commit
- **11.** admin-state in-service
- 12. show controllers dwdm interface-path-id g709

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:Router# configure	

	Command or Action	Purpose
Step 2	controller dwdm interface-path-id Example:	Specifies the DWDM controller name in the notation <i>rack/slot/module/port</i> and enters DWDM configuration mode.
	<pre>RP/0/RSP0/CPU0:Router(config)# controller dwdm 0/1/0/0</pre>	
Step 3	admin-state maintenance or admin-state out-of-service	Disables the DWDM controller. You must disable the
	Example:	controller before you can use the DWDM configuration commands.
	<pre>RP/0/RSP0/CPU0:Router(config-dwdm)# admin-state out-of-service</pre>	
Step 4	commit	Saves configuration changes. This performs the shutdown
	Example:	down, you can proceed with the configuration.
	RP/0/RSP0/CPU0:Router(config-dwdm)# commit	
Step 5	g709 disable	(Optional) Disables the G.709 wrapper. The wrapper is
	Example:	Note
	<pre>RP/0/RP0/CPU0:Router(config-dwdm)# g709 disable</pre>	• The g709 disable command is available on the Cisco 4-Port 10-Gigabit Ethernet DWDM PLIM only.
Step 6	loopback {internal line}	(Optional) Configures the DWDM controller for loopback
	Example:	mode.
	<pre>RP/0/RSP0/CPU0:Router(config-dwdm)# loopback internal</pre>	
Step 7	g709 fec {disable standard}	(Optional) Configures the forward error correction mode
	Example:	(FEC) for the DWDM controller. By default, enhanced FEC is enabled.
	<pre>RP/0/RSP0/CPU0:Router(config-dwdm)# g709 fec disable</pre>	
Step 8	g709 {odu otu} report alarm disable	(Optional) Disables the logging of selected optical channel
	Example:	data unit (ODU) alarms or optical channel transport unit (OTU) alarms to the console for a DWDM controller. By
	<pre>RP/0/RSP0/CPU0:Router(config-dwdm)# g709 odu bdi disable</pre>	default, all alarms are logged to the console.
Step 9	g709 otu overhead tti {expected sent} {ascii hex}	Configures a transmit or expected Trail Trace Identifier (TTI) that is displayed in the show controller dwdm
	Example:	command.
	_	
	<pre>RP/0/RSP0/CPU0:Router(config-dwdm)# g709 otu overhead tti expected ascii test OTU 5678</pre>	

	Command or Action	Purpose
Step 10	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	RP/0/RSP0/CPU0:Router(config-dwdm)# end	
	or	Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:
	RP/0/RSP0/CPU0:Router(config-dwdm)# commit	• 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 11	admin-state in-service	Places the DWDM port in In Service (IS) state, to support
	Example:	all normal operation.
	<pre>RP/0/RSP0/CPU0:Router(config-dwdm)# admin-state in-service</pre>	
Step 12	show controllers dwdm interface-path-id g709	Displays the G.709 Optical Transport Network (OTN)
	Example:	protocol alarms and counters for Bit Errors, along with the FEC statistics and threshold-based alerts.
	RP/0/RSP0/CPU0:Router# show controller dwdm 0/1/0/0 optics	

What to do next

All interface configuration tasks for the POS or Gigabit Ethernet interfaces still must be performed in interface configuration mode. Refer to the corresponding modules in this book for more information.

This task describes how to customize the alarm display and the thresholds for alerts and forward error correction (FEC). You need to use this task only if the default values are not correct for your installation.

How to Perform Performance Monitoring on DWDM Controllers

Performance monitoring parameters are used to gather, store, set thresholds for, and report performance data for early detection of problems. Thresholds are used to set error levels for each performance monitoring parameter. During the accumulation cycle, if the current value of a performance monitoring parameter reaches

or exceeds its corresponding threshold value, a threshold crossing alert (TCA) can be generated. The TCAs provide early detection of performance degradation.

Performance monitoring statistics are accumulated on a 15-minute basis, synchronized to the start of each quarter-hour. They are also accumulated on a daily basis starting at midnight. Historical counts are maintained for thirty-three 15-minute intervals and two daily intervals.

Performance monitoring is described in the following task:

Configuring DWDM Controller Performance Monitoring

This task describes how to configure performance monitoring on DWDM controllers and how to display the performance parameters.

SUMMARY STEPS

- 1. configure
- 2. controller dwdm interface-path-id
- 3. pm {15-min | 24-hour} fec threshold {ec-bits | uc-words} threshold
- 4. pm {15-min | 24-hour} optics threshold {lbc | opr | opt} {max | min} threshold
- **5. pm** {**15-min** | **24-hour**} **otn threshold** *otn-parameter threshold*
- 6. pm {15-min | 24-hour} fec report {ec-bits | uc-words} enable
- 7. pm {15-min | 24-hour} optics report {lbc | opr | opt} {max-tca | min-tca} enable
- 8. pm {15-min | 24-hour} otn report *otn-parameter* enable
- 9. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:Router# configure	
Step 2	controller dwdm interface-path-id	Specifies the DWDM controller name in the notation
	Example:	<i>rack/slot/module/port</i> and enters DWDM configuration mode.
	RP/0/RSP0/CPU0:Router(config)# controller dwdm 0/1/0/0	
Step 3	<pre>pm {15-min 24-hour} fec threshold {ec-bits uc-words} threshold</pre>	Configures a performance monitoring threshold for specific parameters on the FEC layer.
	Example:	
	<pre>RP/0/RSP0/CPU0:Router(config-dwdm)# pm 15-min fec threshold ec-bits 49000000</pre>	
I

	Command or Action	Purpose
	RP/0/RSP0/CPU0:Router(config-dwdm)# pm 15-min fec threshold uc-words xxxxxx	
Step 4	pm {15-min 24-hour} optics threshold {lbc opr opt} {max min} threshold	Configures a performance monitoring threshold for specific parameters on the optics layer.
	Example:	
	RP/0/RSP0/CPU0:Router(config-dwdm)# pm 15-min optics threshold opt max xxx RP/0/RSP0/CPU0:Router(config-dwdm)# pm 15-min optics threshold lbc min xxx	
Step 5	pm { 15-min 24-hour } otn threshold <i>otn-parameter threshold</i>	Configures a performance monitoring threshold for specific parameters on the optical transport network (OTN) layer.
	Example: RP/0/RSP0/CPU0:Router(config-dwdm)# pm 15-min otr	• bbe-pm-fe —Far-end path monitoring background block errors (BBE-PM)
	<pre>threshold bbe-pm-ne xxx RP/0/RSP0/CPU0:Router(config-dwdm) # pm 15-min otn threshold es-sm-fe xxx</pre>	• bbe-pm-ne —Near-end path monitoring background block errors (BBE-PM)
		• bbe-sm-fe —Far-end section monitoring background block errors (BBE-SM)
		• bbe-sm-ne —Near-end section monitoring background block errors (BBE-SM)
		• bber-pm-fe —Far-end path monitoring background block errors ratio (BBER-PM)
		• bber-pm-ne —Near-end path monitoring background block errors ratio (BBER-PM)
		• bber-sm-fe —Far-end section monitoring background block errors ratio (BBER-SM)
		• bber-sm-ne —Near-end section monitoring background block errors ratio (BBER-SM)
		• es-pm-fe —Far-end path monitoring errored seconds (ES-PM)
		• es-pm-ne—Near-end path monitoring errored seconds (ES-PM)
		• es-sm-fe—Far-end section monitoring errored seconds (ES-SM)
		• es-sm-ne—Near-end section monitoring errored seconds (ES-SM)
		• esr-pm-fe —Far-end path monitoring errored seconds ratio (ESR-PM)

Command or Action	Purpose
	• esr-pm-ne—Near-end path monitoring errored seconds ratio (ESR-PM)
	 esr-sm-fe—Far-end section monitoring errored seconds ratio (ESR-SM)
	 esr-sm-ne—Near-end section monitoring errored seconds ratio (ESR-SM)
	• fc-pm-fe —Far-end path monitoring failure counts (FC-PM)
	• fc-pm-ne —Near-end path monitoring failure counts (FC-PM)
	• fc-sm-fe —Far-end section monitoring failure counts (FC-SM)
	• fc-sm-ne —Near-end section monitoring failure counts (FC-SM)
	• ses-pm-fe —Far-end path monitoring severely errored seconds (SES-PM)
	• ses-pm-ne —Near-end path monitoring severely errored seconds (SES-PM)
	• ses-sm-fe —Far-end section monitoring severely errored seconds (SES-SM)
	• ses-sm-ne —Near-end section monitoring severely errored seconds (SES-SM)
	• sesr-pm-fe —Far-end path monitoring severely errored seconds ratio (SESR-PM)
	 sesr-pm-ne—Near-end path monitoring severely errored seconds ratio (SESR-PM)
	• sesr-sm-fe —Far-end section monitoring severely errored seconds ratio (SESR-SM)
	• sesr-sm-ne —Near-end section monitoring severely errored seconds ratio (SESR-SM)
	• uas-pm-fe —Far-end path monitoring unavailable seconds (UAS-PM)
	• uas-pm-ne —Near-end path monitoring unavailable seconds (UAS-PM)
	• uas-sm-fe —Far-end section monitoring unavailable seconds (UAS-SM)
	• uas-sm-ne —Near-end section monitoring unavailable seconds (UAS-SM)
	1

I

	Command or Action	Purpose
Step 6	pm {15-min 24-hour} fec report {ec-bits uc-words} enable	Configures threshold crossing alert (TCA) generation for specific parameters on the FEC layer.
	Example:	
	RP/0/RSP0/CPU0:Router(config-dwdm)# pm 15-min fec report ec-bits enable RP/0/RSP0/CPU0:Router(config-dwdm)# pm 15-min fec report uc-words enable	
Step 7	pm {15-min 24-hour} optics report {lbc opr opt} {max-tca min-tca} enable	Configures TCA generation for specific parameters on the optics layer.
	Example:	
	RP/0/RSP0/CPU0:Router(config-dwdm)# pm 15-min optics report opt enable RP/0/RSP0/CPU0:Router(config-dwdm)# pm 15-min optics report lbc enable	
Step 8	<pre>pm {15-min 24-hour} otn report otn-parameter enable Example:</pre>	Configures TCA generation for specific parameters on the optical transport network (OTN) layer. OTN parameters are shown in Step 5.
	<pre>RP/0/RSP0/CPU0:Router(config-dwdm)# pm 15-min otn report bbe-pm-ne enable RP/0/RSP0/CPU0:Router(config-dwdm)# pm 15-min otn report es-sm-fe enable</pre>	
Step 9	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	RP/0/RSP0/CPU0:Router(config-dwdm)# end	Incommitted changes found commit them before
	or	exiting(yes/no/cancel)? [cancel]:
	RP/0/RSP0/CPU0:Router(config-dwdm)# commit	• 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 IPoDWDM

This section provides the following configuration procedures:

Configuring the Optical Layer DWDM Ports

Use this procedure to configure the Optical Layer DWDM ports.

SUMMARY STEPS

- 1. configure
- 2. controller dwdm interface-path-id
- **3.** network srlg value1 value2 value3
- 4. network port id *id-number*
- 5. network connection id *id-number*
- 6. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:Router# config	
Step 2	controller dwdm interface-path-id	Specifies the DWDM controller and enters DWDM controller mode.
	Example:	
	RP/0/RSP0/CPU0:Router(config)# controller dwdm 0/1/0/1	
Step 3	network srlg value1 value2 value3	Configures the Shared Risk Link Group (SRLG).
	Example:	
	RP/0//CPU0:Router(config-dwdm)# network srlg value1 value2 value3	L
Step 4	network port id id-number	Assigns an identifier number to a port for the Multi Service
	Example:	Transport Protocol (MSTP).
	RP/0/RSP0/CPU0:Router(config-dwdm)# network port id 1/0/1/1	

	Command or Action	Purpose
Step 5	network connection id id-number	Configures a connection identifier for the Multi Service
	Example:	Transport Protocol (MSTP).
	RP/0/RSP0/CPU0:Router(config-dwdm)# network connection id 1/1/1/1	
Step 6	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	RP/0/RSP0/CPU0:Router(config-dwdm)# end	
	or	Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:
	RP/0/RSP0/CPU0:Router(config-dwdm)# commit	• Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring the Administrative State of DWDM Optical Ports

Use this procedure to configure the administrative state and optionally set the maintenance embargo flag.

SUMMARY STEPS

- 1. configure
- 2. controller dwdm interface-path-id
- **3.** admin-state {in-service | maintenance | out-of-service}
- 4. exit
- 5. interface pos interface-path-id
- **6.** or
- 7. interface tengige interface-path-id
- 8. maintenance disable
- 9. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:Router# config	
Step 2	controller dwdm interface-path-id	Specifies the DWDM controller and enters DWDM
	Example:	controller mode.
	RP/0/RSP0/CPU0:Routerconfig)# controller dwdm 0/1/0/1	
Step 3	admin-state {in-service maintenance out-of-service}	Specifies the transport administration state.
	Example:	
	RP/0/RSP0/CPU0:Router(config-dwdm)# admin-state maintenance	
Step 4	exit	Exits to the previous mode.
	Example:	
	RP/0/RSP0/CPU0:Router(config-dwdm)# exit	
Step 5	interface pos interface-path-id	
Step 6	or	
Step 7	interface tengige interface-path-id	Specifies the interface and enters interface configuration
	Example:	mode.
	<pre>RP/0/RSP0/CPU0:Router(config)# interface pos</pre>	
	1/0/1/1 or	
	<pre>RP/0/RSP0/CPU0:Router(config)# interface tengige 1/0/1/1</pre>	
Step 8	maintenance disable	Provisions the maintenance embargo flag, which preven
	Example:	maintenance activities from being performed on an interface.
	RP/0/RSP0/CPU0:Router(config-if)# maintenance disable	
Step 9	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts
	RP/0/RSP0/CPU0:Router(config-dwdm)# end	you to commit changes.

Command or Action	Purpose
Or RP/0/RSP0/CFU0:Router(config-dwdm)# commit	 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes. Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes. Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring Proactive FEC-FRR Triggering

Use this procedure to configure automatic triggering of Forward Error Correction-Fast Re-Route (FEC-FRR).

SUMMARY STEPS

- 1. configure
- 2. controller dwdm interface-path-id
- 3. proactive
- 4. logging signal file-name
- 5. proactive trigger threshold x-coefficient y-power
- 6. proactive trigger window window
- **7. proactive revert threshold** *x*-*coefficient y*-*power*
- 8. proactive revert window window
- 9. end or commit

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:Router# config	

	Command or Action	Purpose
Step 2	controller dwdm interface-path-id Example:	Specifies the DWDM controller and enters DWDM controller mode.
	RP/0/RSP0/CPU0:Router(config)# controller dwdm 0/1/0/1	
Step 3	proactive	Enables automatic triggering of FEC-FRR.
	Example:	
	RP/0/RSP0/CPU0:Router(config-dwdm)# proactive enable	
Step 4	logging signal file-name	Enables10 millisecond proactive monitoring of FEC-FRR.
	Example:	
	<pre>RP/0/RSP0/CPU0:Router(config-dwdm)# logging signal LogFile1</pre>	
Step 5	proactive trigger threshold x-coefficient y-power	Configures the trigger threshold of FEC-FRR in the form
	Example:	of xE-y.
	RP/0/RSP0/CPU0:Routerconfig-dwdm)# proactive trigger threshold 1 9	
Step 6	proactive trigger window window Example:	Configures the trigger window (in milliseconds) in which FRR may be triggered.
	RP/0/RSP0/CPU0:Router(config-dwdm)# proactive trigger window 10000	
Step 7	proactive revert threshold <i>x</i> -coefficient <i>y</i> -power	Configures the revert threshold (in the form of $xE-y$) to
	Example:	trigger reverting from the FEC-FRR route back to the original route.
	RP/0/RSP0/CPU0:Router(config-dwdm)# proactive revert threshold 1 9	
Step 8	proactive revert window window	Configures the revert window in which reverting from the
	Example:	FEC-FRR route back to the original route is triggered.
	RP/0/RSP0/CPU0:Router(config-dwdm)# proactive revert window 600000	
Step 9	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts
	RP/0/RSP0/CPU0:Router(config-dwdm)# end	you to commit changes:
	or	Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:

Command or Action	Purpose
 RP/0/RSP0/CPU0:Router(config-dwdm)# commit	• 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

This section includes these examples:

Turning On the Laser: Example

Note

This is a required configuration. The DWDM cards will not operate without this configuration.

This example shows how to turn on the laser and place a DWDM port in In Service (IS) state:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:Router(config)# controller dwdm 0/1/0/1
RP/0/RP0/CPU0:Router(config-dwdm)# admin-state in-service
RP/0/RP0/CPU0:Router(config-dwdm)# commit
```

Turning Off the Laser: Example

This example shows how to turn off the laser, stop all traffic and place a DWDM port in Out of Service (OOS) state:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:Router(config)# controller dwdm 0/1/0/1
RP/0/RP0/CPU0:Router(config-dwdm)# admin-state out-of-service
RP/0/RP0/CPU0:Router(config-dwdm)# commit
```

DWDM Controller Configuration: Examples

This example shows how to customize the alarm display and the thresholds for alerts and forward error correction (FEC):

```
RP/0/RSP0/CPU0:Router# configure
RP/0/RSP0/CPU0:Router(config)# controller dwdm 0/1/0/0
RP/0/RSP0/CPU0:Router(config-dwdm)# maintenance out-of-service
RP/0/RSP0/CPU0:Router(config-dwdm)# g709 disable
RP/0/RSP0/CPU0:Router(config-dwdm)# loopback internal
RP/0/RSP0/CPU0:Router(config-dwdm)# g709 fec standard
RP/0/RSP0/CPU0:Router(config-dwdm)# g709 odu bdi disable
RP/0/RSP0/CPU0:Router(config-dwdm)# g709 odu bdi disable
RP/0/RSP0/CPU0:Router(config-dwdm)# maintenance in-service
RP/0/RSP0/CPU0:Router(config-dwdm)# maintenance in-service
```

DWDM Performance Monitoring: Examples

This example shows how to configure performance monitoring for the optics parameters and how to display the configuration and current statistics:

```
RP/0/RSP0/CPU0:Router# configure
RP/0/RSP0/CPU0:Router(config) # controller dwdm 0/2/0/0
RP/0/RSP0/CPU0:Router(config-dwdm) # pm 15-min optics threshold opt max 2000000
RP/0/RSP0/CPU0:Router(config-dwdm) # pm 15-min optics threshold opt min 200
RP/0/RSP0/CPU0:Router(config-dwdm) # pm 15-min optics threshold lbc max 3000000
RP/0/RSP0/CPU0:Router(config-dwdm)# pm 15-min optics threshold lbc min 300
RP/0/RSP0/CPU0:Router(config-dwdm) # pm 15-min optics threshold opr max 4000000
RP/0/RSP0/CPU0:Router(config-dwdm) # pm 15-min optics threshold opr min 400
RP/0/RSP0/CPU0:Router(config-dwdm) # pm 15-min optics report opt max-tca enable
RP/0/RSP0/CPU0:Router(config-dwdm) # pm 15-min optics report opt min-tca enable
RP/0/RSP0/CPU0:Router(config-dwdm)# pm 15-min optics report opr max-tca enable
RP/0/RSP0/CPU0:Router(config-dwdm) # pm 15-min optics report opr min-tca enable
RP/0/RSP0/CPU0:Router(config-dwdm) # pm 15-min optics report lbc max-tca enable
RP/0/RSP0/CPU0:Router(config-dwdm) # pm 15-min optics report lbc min-tca enable
RP/0/RSP0/CPU0:Router(config-dwdm) # exit
RP/0/RSP0/CPU0:Router(config)# exit
Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]:y
LC/0/2/CPU0:Jul 12 04:10:47.252 : plim 4p 10qe dwdm[194]: %L1-PMENGINE-4-TCA : Port DWDM
0/2/0/0 reports OPTICS TX-PWR-MIN(NE) PM TCA with current value 0, threshold 200 in current
15-min interval window
LC/0/2/CPU0:Jul 12 04:10:47.255 : plim 4p 10ge dwdm[194]: %L1-PMENGINE-4-TCA : Port DWDM
0/2/0/0 reports OPTICS RX-PWR-MIN(NE) PM TCA with current value 68, threshold 400 in current
15-min interval window
RP/0/RP1/CPU0:Jul 12 04:09:05.443 : config[65678]: %MGBL-CONFIG-6-DB COMMIT : Configuration
committed by user 'lab'. Use 'show configuration commit changes 1000000001' to view the
changes.
RP/0/RP1/CPU0:Jul 12 04:09:05.604 : config[65678]: %MGBL-SYS-5-CONFIG I : Configured from
console by lab
RP/0/RSP0/CPU0:Router# show controllers dwdm 0/2/0/0 pm interval 15-min optics 0
Optics in the current interval [ 4:15:00 - 04:26:02 Wed Jul 12 2006]
                 AVG MAX Threshold TCA Threshold TCA
            MIN
                                 (min) (enable) (max) (enable)
LBC[mA]: 3605 4948 6453
                                 300
                                        YES 300000
                                                              YES
OPT[uW] : 2593 2593
                           2593 200
                                             YES
                                                    2000000 YES
                                            YES
OPR[uW] : 69
                   69
                           70
                                  400
                                                     4000000
                                                               YES
```

IPoDWDM Configuration: Examples

This section includes the following examples:

Optical Layer DWDM Port Configuration: Examples

This example shows how to configure Optical Layer DWDM ports.

```
RP/0/RSP0/CPU0:Router# configure
RP/0/RSP0/CPU0:Router(config)# controller dwdm 0/1/0/1
RP/0/RP0/CPU0:Router(config-dwdm)# network srlg value1 value2 value3
RP/0/RSP0/CPU0:Router(config-dwdm)# network port id 1/0/1/1
RP/0/RSP0/CPU0:Router(config-dwdm)# network connection id 1/1/1/1
```

Administrative State of DWDM Optical Ports Configuration: Examples

The following examples show how to configure the administrative state and optionally set the maintenance embargo flag:

For TenGigabit Interface

```
RP/0/RSP0/CPU0:Router# configure
RP/0/RSP0/CPU0:Router(config)# controller dwdm 0/1/0/1
RP/0/RSP0/CPU0:Router(config-dwdm)# admin-state in-service
RP/0/RSP0/CPU0:Router(config-dwdm)# exit
RP/0/RSP0/CPU0:Router(config)# interface tengige 1/0/1/1
RP/0/RSP0/CPU0:Router(config-if)# maintenance disable
RP/0/RSP0/CPU0:Router(config-if)# commit
```

Proactive FEC-FRR Triggering Configuration: Examples

This example shows how to configure automatic triggering of Forward Error Correction-Fast Re-Route (FEC-FRR):

```
RP/0/RSP0/CPU0:Router# configure
RP/0/RSP0/CPU0:Router(config)# controller dwdm 0/1/0/1
RP/0/RSP0/CPU0:Router(config-dwdm)#proactive
RP/0/RSP0/CPU0:Router(config-dwdm)# logging signal LogFile1
RP/0/RSP0/CPU0:Router(config-dwdm)# proactive trigger threshold 1 9
RP/0/RSP0/CPU0:Router(config-dwdm)# proactive trigger window 10000
RP/0/RSP0/CPU0:Router(config-dwdm)# proactive revert threshold 1 9
RP/0/RSP0/CPU0:Router(config-dwdm)# proactive revert threshold 1 9
```



Configuring 400G Digital Coherent Optics

Feature Name	Release Information	Description
Support for DP04QSDD-HE0 Optical Module	Release 7.9.1	This release introduces support for the Cisco 400G QSFP-DD High-Power (Bright) Optical Module, Ethernet Variant.
		The Cisco 400G QSFP-DD High-Power (Bright) Optical module is an enhanced version of the currently available QSFP-DD ZR+ Optical Module, leveraging the same operational modes but providing as a major enhancement the increase of the Tx Optical Power up to +1dBm.
		DP04QSDD-HE0 optical module is supported in the following hardware:
		• A9K-20HG-FLEX-SE/TR/FC
		• A9K-8HG-FLEX-SE/TR/FC
Extended Support for DP04QSDD-HE0 Optical Module	Release 7.10.1	With this release, the support for DP04QSDD-HE0 optical module is extended to the following hardware:
		• A99-10X400GE-X-SE/TR
		• A9903-20HG-PEC
		• A9903-20HG-PEC-FC
		• ASR9902 (100G)
		• ASR9902-FC (100G)
Support for 3x100G Muxponder Mode on A99-10X400CEXSE/A99-10X400CEX-TR Line Cards	Release 7.9.1	QDD-400G-ZRP-S optical module operating in A99-10X400GE-X-SE/A99-10X400GE-X-TR line cards now supports 3x100G muxponder mode.

Feature Name	Release Information	Description
oFEC Traffic Configuration for QDD-400G-ZRP-S	Release 7.9.1	QDD-400G-ZRP-S optical module can now support the following oFEC traffic configurations:
		• 400G-TXP-1x1 DAC-16 QAM
		• 3x100G-MXP-1x1 DAC-8 QAM
		This increases the interoperability of the QDD-400G-ZRP-S optical module across network components supporting these formats.
Support for 8 and 16 Quadrature Amplitude Modulation (QAM) Formats on QDD-400G-ZRP-S Optical Modules	Release 7.8.1	QDD-400G-ZRP-S optical module operating in 200G interfaces now supports the 8 QAM and 16 QAM modulation formats. This increases the interoperability of the optical module across network components supporting these formats.

Cisco offers a range of the new 400G Digital Coherent QSFP-DD optical modules. The two optical modules that are available are:

- QDD-400G-ZR-S
- QDD-400G-ZRP-S
- DP04QSDD-HE0

For more information about the optic module portfolio, see the <u>Cisco 400G Digital Coherent Optics QSFP-DD</u> <u>Optical Modules Data Sheet</u>.

This chapter describes the 400G Digital Coherent QSFP-DD optical modules and their supported configurations.

The following table shows the line cards, ports, and modes that support these two optical modules.

Table 17: Supported Line Cards, Ports, and Modes

Components	Default Port Speed	Supported Front Panel Ports	1x400G Transponder Mode	4x100G Muxponder Mode	3x100G Muxponder Mode	2x100G Muxponder Mode	1x100G Transponder Mode
AKZHCHEXSFAKZHCHEXIR	1x100G	0, 7, 8, 12, 19	Yes	Yes	Yes	Yes	Yes
AKHOHEXSEAKHOHEXIR	1x100G	0, 7	Yes	Yes	Yes	Yes	Yes
A910X100EX5EA930X100EXIR	1x400G	3, 5, 6, 7, 9	Yes	Yes	Yes	Yes	Yes
A9903-20HG-PEC	1x100G	0, 4, 8, 12, 16	Yes	Yes	Yes	Yes	Yes
ASR-9902	1x100G	11, 37	No	No	No	No	Yes

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, 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 traffic1 configuration and firmware download. The Cisco IOS XR software collects performance monitoring data and alarms.

Due to more power consumption by the 400G Digital Coherent QSFP-DD optical modules, the Cisco IOS XR software operates the fans at an higher speed to cool these optical modules.

Configuring frequency, chromatic dispersion, optical transmit power, digital to analog conversion (DAC) sampling rate, and FEC parameters impacts traffic. Also, configuring modulation is supported. Modulation is dependent on the mode of operation. See Table 20: QDD-400G-ZR-S Transponder and Muxponder Configuration Values, on page 392 and #unique_8 unique_8_Connect_42_table_u5x_f13_ppb, on page 393.

The 400G Digital Coherent QSFP-DD optical module configuration is divided into the following categories:

- Traffic configuration Comprises configuring DAC rate, breakout, and FEC parameters. Applicable for optics controllers:
 - Configuring DAC Rate, on page 394
 - Configuring Breakout, on page 396
 - Configuring FEC, on page 398
- Optical configuration Comprises configuring frequency, chromatic dispersion, and optical transmit power. Applicable for optics controllers:
 - Configuring Frequency, on page 400
 - Configuring Chromatic Dispersion, on page 401
 - Configuring Optical Transmit Power, on page 403
- 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 406
 - Configuring PM Parameters, on page 406
- Loopback configuration Configures loopback. Applicable for coherent DSP controller:
 - Configuring Loopback, on page 409
- 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:

I

• Configuring Alarms Threshold, on page 413

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 18: 400G Digital Coherent QSFP-DD) Traffic Configuration Values
---	--------------------------------

Optical Module	Client Speed	Trunk Speed	Frequency	FEC	Modulation	DAC-Rate	Chromatic Dispersion (CD)	Transmitted (Tx) Power
QID40GAS	1x400, 4x100	400G	C-Band, 196.1 To 191.3 THz	cFEC	16QAM	1x1	-2400 to +2400	Each optical module has its own transmitting (TX) power range. You can change the transmitting (TX) power value based on the module capability.
QDARCARS	1x400, 4x100, 3x100, 2x100, 1x100	400G, 300G, 200G, 100G	C-Band, 196.1 To 191.3 THz	oFEC, cFEC	16QAM, 8QAM, QPSK	1x1.25, 1x1	-160000 to +160000	Each optical module has its own transmitting (TX) power optimal values. You can change the transmitting (TX) power value based on the module capability.

Optical Module	Client Speed	Trunk Speed	Frequency	FEC	Modulation	DAC-Rate	Chromatic Dispersion (CD)	Transmitted (Tx) Power
DRACADDHHO	1x400, 4x100, 3x100, 2x100, 1x100	400G, 300G, 200G, 100G	C-Band, 196.1 To 191.3 THz	oFEC, cFEC	16QAM, 8QAM, QPSK	1x1.25, 1x1.5	-160000 to +160000	Each optical module has its own transmitting (TX) power optimal values. You can change the transmitting (TX) power value based on the module capability.

QDD-400G-ZR-S and QDD-400G-ZRP-S FPD Upgrade

Table 19: Feature History Table

Feature Name	Release Information	Feature Description
FPD Upgrades Enabled for QDD-400G-ZR-S and QDD-400G-ZRP-S Optical Modules	Release 7.3.2	This feature allows you to perform Field Programmable Device (FPD) upgrades on the QDD-400G-ZR-S and QDD-400G-ZRP-S optical modules to ensure they have the latest fixes and features. For more information about the optic module portfolio, see the <u>Cisco 400G</u> <u>Digital Coherent Optics QSFP-DD</u> <u>Optical Modules Data Sheet</u> .

Although an FPD upgrade is not mandatory in this release, we recommend upgrading the FPD to the latest version in the subsequent releases to ensure that all the latest fixes and features are enabled on the optical modules. Auto and parallel FPD upgrades are not supported. Only a manual FPD upgrade (one optical module at a time) using the **upgrade hw-module location** *node-id* **fpd***-name* command is supported.

The *fpd-name* can be obtained from the FPD description column of the **show fpd package** or the **show hw-module fpd** command. The *fpd-name* in the FPD description column displays QDD_*instance_port-number*. For example, depending on the instance and the port number, the FPD names for the QDD-400G-ZR-S and QDD-400G-ZRP-S modules will be QDD_0_3, QDD_0_0, and so on.



Note Only in case of Cisco ASR 9903 routers, the *fpd-name* in the FPD description column displays QDD_*bay_port-number*. The *bay* value for Cisco 9903 routers is always 1. Therefore, depending on the port number, the FPD names for the QDD-400G-ZR-S and QDD-400G-ZRP-S modules will be QDD_1_3, QDD_1_4, and so on.

See the "**Upgrading Field-Programmable Devices**" chapter in the *System Management Configuration Guide for Cisco ASR 9000 Series Routers* for details on the procedure to upgrade the FPD.

QDD-400G-ZR-S Transponder and Muxponder Configuration Values

The following table contains the possible Transponder and Muxponder configuration values for the QDD-400G-ZR-S optical module:

ТХР/МХР	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

Table 20: QDD-400G-ZR-S Transponder and Muxponder Configuration Values

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:

TXP/MXP	Client	Trunk	Modulation	FEC	DAC Rate
400G-TXP	1 Client, 400G speed	1 trunk, 400G speed	16 QAM	oFEC	1x1.25
400G-TXP	1 Client, 400G speed	1 trunk, 400G speed	16 QAM	cFEC	1x1
400G-TXP	1 Client, 400G speed	1 trunk, 400G speed	16 QAM	oFEC	1x1
4x100G- MXP	4 clients, 100G speed	1 trunk, 400G speed	16 QAM	oFEC	1x1.25
4x100G- MXP	4 clients, 100G speed	1 trunk, 400G speed	16 QAM	cFEC	1x1
3x100G-MXP	3 clients, 100G speed	1 trunk, 300G speed	8 QAM	oFEC	1x1.25

TXP/MXP	Client	Trunk	Modulation	FEC	DAC Rate
3x100G-MXP	3 clients, 100G speed	1 trunk, 300G speed	8 QAM	oFEC	1x1
2x100G-MXP	2 clients, 100G speed	1 trunk, 200G speed	QPSK	oFEC	1x1.50
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-MXP	1 client, 100G speed	1 trunk, 100G speed	QPSK	oFEC	1x1.50

DP04QSDD-HE0 Transponder and Muxponder Configuration Values

The following table contains the possible Transponder and Muxponder configuration values for the DP04QSDD-HE0 optical module:

ТХР/МХР	Client	Trunk	Modulation	FEC	DAC Rate
400G-TXP	1 Client, 400G speed	1 trunk, 400G speed	16 QAM	oFEC	1x1.25
400G-TXP	1 Client, 400G speed	1 trunk, 400G speed	16 QAM	cFEC	1x1.5
400G-TXP	1 Client, 400G speed	1 trunk, 400G speed	16 QAM	oFEC	1x1.5
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
4x100G-MXP	4 clients, 100G speed	1 trunk, 400G speed	16 QAM	oFEC	1x1.5
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
2x100G-MXP	2 clients, 100G speed	1 trunk, 200G speed	QPSK	oFEC	1x1.5
2x100G-MXP	2 clients, 100G speed	1 trunk, 200G speed	8 QAM	oFEC	1x1.25

TXP/MXP	Client	Trunk	Modulation	FEC	DAC Rate
2x100G-MXP	2 clients, 100G speed	1 trunk, 200G speed	16 QAM	oFEC	1x1.25
1x100G-MXP	1 client, 100G speed	1 trunk, 100G speed	QPSK	oFEC	1x1.5

- Configuring DAC Rate, on page 394
- Configuring Breakout, on page 396
- Configuring FEC, on page 398
- Configuring Frequency, on page 400
- Configuring Chromatic Dispersion, on page 401
- Configuring Optical Transmit Power, on page 403
- Configuring Performance Monitoring, on page 406
- Configuring PM Parameters, on page 406
- Configuring Loopback, on page 409
- Laser Squelching, on page 411
- Small Frame Padding, on page 412
- Configuring Alarms Threshold, on page 413

Configuring DAC Rate

You can set the DAC (digital to analog conversion) sampling rate on optics controllers. You can modify the DAC sampling rate only on the QDD-400G-ZRP-S optical module.



Note QDD-400G-ZR-S supports 1x1 dac-rate in cFEC mode. QDD-400G-ZRP-S supports 1x1 dac-rate in cFEC mode and 1x1.25 dac-rate in oFEC mode. DP04QSDD-HE0 optical modules support 1x1.5 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 0/19/0/3
Router(config-Optics)#dac-rate 1x1.25
```

Verification

This example shows the running configuration:

```
Router#show run controller optics 0/19/0/3
Thu May 13 12:52:35.020 UTC
controller Optics0/19/0/3
DAC-Rate 1x1.25
!
```

Verification

This example shows how to verify the configured DAC rate for the optics controller:

```
Router#show controller optics 0/19/0/3
Controller State: Up
Transport Admin State: In Service
Laser State: On
LED State: Green
FEC State: FEC OFEC
Optics Status
         Optics Type: 400G QSFP-DD 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:
         Laser Bias Current = 70.9 mA
         Actual TX Power = -7.06 dBm
         RX Power = -7.88 dBm
         RX Signal Power = -7.88 dBm
         Frequency Offset = -220 MHz
         Laser Temperature = 53.79 Celsius
         Laser Age = 0 \%
         DAC Rate = 1 \times 1.25
         Performance Monitoring: Enable
         THRESHOLD VALUES
         _____
         Parameter
                                   High Alarm Low Alarm High Warning Low Warning
                                    _____
          _____
                                                 -----
                                                                           _____

        Rx Power Threshold(dBm)
        1.9
        -28.2
        0.0

        Tx Power Threshold(dBm)
        0.0
        -18.0
        -2.0

        0.00
        0.00
        0.00
        0.00

                                                                                 -25.0
                                                                               -16.0
                                        0.00
                                                                   0.00
         LBC Threshold(mA)
                                                    0.00
                                                                                 0.00
         Temp. Threshold(celsius)80.00-5.00Voltage Threshold(volt)3.463.13
                                                                  75.00
                                                                               15.00
                                                                   3.43
                                                                                3.16
         LBC High Threshold = 98 %
         Configured Tx Power = -7.00 dBm
         Configured CD High Threshold = 80000 ps/nm
         Configured CD lower Threshold = -80000 \text{ ps/nm}
         Configured OSNR lower Threshold = 9.00 dB
         Configured DGD Higher Threshold = 180.00 ps
         Baud Rate = 30.0692729950 GBd
         Modulation Type: QPSK
         Chromatic Dispersion -2 ps/nm
         Configured CD-MIN -4000 ps/nm CD-MAX 4000 ps/nm
         Second Order Polarization Mode Dispersion = 109.00 ps^2
         Optical Signal to Noise Ratio = 32.30 dB
         Polarization Dependent Loss = 0.05 dB
         Polarization Change Rate = 0.00 rad/s
         Differential Group Delay = 0.00 ps
         Temperature = 48.00 Celsius
         Voltage = 3.30 V
 Transceiver Vendor Details
         Form Factor
                                : QSFP-DD
         Optics type
                               : OSFP-DD 400G-ZRP-S
         Name
                                : CISCO-ACACIA
         OUI Number
                                : 7c.b2.5c
                                : DP04QSDD-E30-19E
         Part Number
         Rev Number
                                : 01
         Serial Number
                                : ACA2503003X
```

```
: QDD-400G-ZRP-S
```

PID

VID	: ES03
Firmware Version	: 61.20 (Build : 13)
Date Code(yy/mm/dd)	: 21/01/22

Configuring Breakout

Feature Name	Release Information	Description
Support for 2x100G, 3x100G and 4x100G Breakout Configurations	Release 7.8.1	This release introduces 2x100G, 3x100G and 4x100G breakout configurations on the following line cards:
		• A9903-20HG-PEC (2x100G, 3x100G, and 4x100G)
		• A99-10X400GE-X-SE/TR (3x100G)
		• A9K-20HG-FLEX-SE/TR (2x100G, 3x100G, and 4x100G)
		• A9K-8HG-FLEX-SE/TR (2x100G, 3x100G, and 4x100G)

Using the **hw-module location** *node-id* port *port-number* **breakout** *muxponder-mode* command, you can configure breakout on the ports.

For configuring breakout on Cisco ASR 9903 router, use the following command:

hw-module location node-id bay bay-number port port-number breakout muxponder-mode

The value for bay-number is always 1.

For the supported line cards, ports, and modes, see Table 17: Supported Line Cards, Ports, and Modes, on page 388.

Configuring Breakout on the Line card with Default 100G Port Speed

The ports with default 100G speed can be configured to operate at 400G port speed using the following command:

This example shows how to configure 1x400G transponder mode on a port with default 100G port speed:

hw-module location 0/0/CPU0 port 0 breakout 1xFourHundredGigE

To remove the breakout configuration and revert to the default 100G port speed, use the **no** form of the command.

For example:

no hw-module location 0/0/CPU0 port 0 breakout 1xFourHundredGigE



Note In the A9903-20HG-PEC card, to configure 1x400G transponder mode, use the hw-module location 0/0/CPU0 bay 1 port 0 breakout 1xFourHundredGigE command. To remove the breakout configuration and revert to the default 100G port speed, use the no for the command - no hw-module location 0/0/CPU0 bay 1 port 0 breakout 1xFourHundredGigE.

To configure 4x100G or 2x100G muxponder mode, use the 4x100G or 2x100G breakout on the supported port(s).

For example:

- hw-module location 0/0/CPU0 port 0 breakout 4xHundredGigE
- hw-module location 0/0/CPU0 port 7 breakout 2xHundredGigE
- hw-module location 0/0/CPU0 port 7 breakout 3xHundredGigE

To remove the breakout configuration and revert to the default 100G port speed, use the **no** form of the command.

For example:

- no hw-module location 0/0/CPU0 port 0 breakout 4xHundredGigE
- no hw-module location 0/0/CPU0 port 7 breakout 2xHundredGigE
- no hw-module location 0/0/CPU0 port 7 breakout 3xHundredGigE

Configuring Breakout on the Line card with Default 400G Port Speed

The ports with default 400G speed can be configured to operate at 100G port speed using the following command:

This example shows how to configure 1x100G transponder mode on a port with default 400G port speed:

For example:

• hw-module location 0/0/CPU0 port 0 breakout 1xHundredGigE

To remove the breakout configuration and revert to the default 400G port speed, use the **no** form of the command.

For example:

• no hw-module location 0/0/CPU0 port 0 breakout 1xHundredGigE

To configure 4x100G or 2x100G or 3x100G muxponder mode, use the 4x100G or 2x100G or 3x100G breakout on the supported port(s).

For example:

- hw-module location 0/0/CPU0 port 0 breakout 4xHundredGigE
- hw-module location 0/0/CPU0 port 7 breakout 2xHundredGigE
- hw-module location 0/0/CPU0 port 7 breakout 3xHundredGigE

To remove the breakout configuration and revert to the default 400G port speed, use the **no** form of the command.

For example:

- no hw-module location 0/0/CPU0 port 0 breakout 4xHundredGigE
- no hw-module location 0/0/CPU0 port 7 breakout 2xHundredGigE
- no hw-module location 0/0/CPU0 port 7 breakout 3xHundredGigE

Muxponder Mode Configuration Example

The following example shows how to configure muxponder mode on the optics controller:

```
Router#config
Router(config)#hw-module location 0/0/CPU0 port 3 breakout 4xHundredGigE
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 ipv4 interface brief location** | **include** *R/S/I/P* command.

Running Configuration

This example shows the running configuration for the optics controller:

```
Router#show running-config | inc break
Building configuration...
hw-module location 0/1/CPU0 port 1 breakout 1xHundredGigE
hw-module location 0/1/CPU0 port 3 breakout 4xHundredGigE
hw-module location 0/1/CPU0 port 8 breakout 1xHundredGigE
!
```

Verification

This example shows how to verify the muxponder mode configuration:

```
Router#show ipv4 interface brief location 0/1/CPU0 | inc 0/1/0/3
HundredGigE0/1/0/3/0 unassigned Shutdown Down default
HundredGigE0/1/0/3/1 unassigned Shutdown Down default
HundredGigE0/1/0/3/2 unassigned Shutdown Down default
HundredGigE0/1/0/3/3 unassigned Shutdown Down default
```

Configuring FEC

You can configure forward error correction (FEC) only on optics controllers. You can modify FEC only on the QDD-400G-ZRP-S optical 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) and QDD-400G-ZRP-S 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 0/19/0/5
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 run controllers optics 0/19/0/5
controller 0/19/0/5
fec CFEC
!
```

Verification

This example shows how to verify the FEC configuration for the optics controller:

```
Router#show controller coherentdsp 0/19/0/5
```

```
: CoherentDSP 0/19/0/5
Port.
Controller State
                                               : Up
Inherited Secondary State
                                               : Normal
Configured Secondary State
                                              : Normal
Derived State
                                              : In Service
Loopback mode
                                              : None
BER Thresholds
                                              : SF = 1.0E-5 SD = 1.0E-7
Performance Monitoring
                                               : Enable
Bandwidth
                                               : 400.0Gb/s
Alarm Information:
LOS = 0 LOF = 0 LOM = 0
OOF = 0 OOM = 0 AIS = 0
IAE = 0 BIAE = 0
                    SF BER = 0
SD BER = 0 BDI = 0 TIM = 0
FECMISMATCH = 0 FEC-UNC = 0 FLEXO GIDM = 0
FLEXO-MM = 0 FLEXO-LOM = 0 FLEXO-RDI = 0
FLEXO-LOF = 1
Detected Alarms
                                               : None
Bit Error Rate Information
PREFEC BER
                                               : 4.5E-04
POSTFEC BER
                                               : 0.0E+00
                                               : 10.40 dB
Q-Factor
Q-Margin
                                               : 3.20dB
OTU TTI Received
FEC mode
                                               : C FEC
```

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 caculates 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 0/1/0/9
Router(config-Optics)#dwdm-carrier 100MHz-grid frequency 1921500
Router(config-Optics)#commit
Router(config-Optics)#exit
Router(config)#exit
```

Running Configuration

Laser Age = 0 %

This example shows the running configuration:

```
Router#show run controller optics 0/1/0/9
controller Optics0/1/0/9
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 controllers optics 0/1/0/9
Controller State: Up
Transport Admin State: In Service
Laser State: On
LED State: Green
FEC State: FEC OFEC
Optics Status
    Optics Type: 400G QSFP-DD 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:
   Laser Bias Current = 52.5 mA
    Actual TX Power = -11.03 dBm
    RX Power = -12.15 dBm
   RX Signal Power = -12.15 dBm
   Frequency Offset = -140 MHz
    Laser Temperature = 52.45 Celsius
```

L

```
DAC Rate = 1x1.25
Performance Monitoring: Enable
```

THRESHOLD VALUES

Parameter	High Alarm	Low Alarm	High Warning	Low Warning
Rx Power Threshold(dBm)	1.9	-23.0	0.0	-20.0
Tx Power Threshold(dBm)	0.0	-18.0	-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

```
LEC High Threshold = 98 %
Configured Tx Power = -7.00 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 = 180.00 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 = 36.00 ps^2
Optical Signal to Noise Ratio = 35.80 dB
Polarization Dependent Loss = 0.08 dB
Polarization Change Rate = 0.00 rad/s
Differential Group Delay = 1.00 ps
```

Temperature = 54.00 Celsius Voltage = 3.29 V

Transceiver Vendor Details

```
Form Factor : QSFP-DD
Optics type : QSFP-DD 400G-ZRP-S
Name : CISCO-ACACIA
OUI Number : 7c.b2.5c
Part Number : DP04QSDD-E30-19D
Rev Number : B0
Serial Number : ACA2440001M
PID : QDD-400G-ZRP-S
VID : VES1
Firmware Version : 61.20 (Build : 13)
Date Code(yy/mm/dd) : 20/10/02
```

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:

Muxponder Rate	FEC Value	Default CD Search Range (Min-Max)
400	OFEC	-26000 to +26000

Muxponder Rate	FEC Value	Default CD Search Range (Min-Max)
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 0/19/0/5
Router(config-Optics)#cd-max 2500
Router(config-Optics)#cd-min -2500
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 0/19/0/5
controller Optics0/19/0/5
cd-min -2500
cd-max 2500
!
```

Verification

This example shows how to verify the configured chromatic dispersion values for the optics controller:

```
Router#show controller optics 0/19/0/5
Controller State: Up
Transport Admin State: In Service
Laser State: On
LED State: Red
 FEC State: FEC CFEC
Optics Status
         Optics Type: 400G QSFP-DD 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:
        Laser Bias Current = 52.5 mA
        Actual TX Power = 2.02 dBm
        RX Power = 0.91 dBm
        RX Signal Power = -7.44 dBm
        Frequency Offset = -65 MHz
        Laser Temperature = 54.54 Celsius
        Laser Age = 0 %
        DAC Rate = 1x1
        Performance Monitoring: Enable
```

L

Parameter	High Alarm	Lo
THRESHOLD VALUES		

Parameter	High Alarm	Low Alarm	High Warning	Low Warning
Rx Power Threshold(dBm)	1.9	-23.0	0.0	-20.0
Tx Power Threshold(dBm)	0.0	-18.0	-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 = -7.00 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 = 180.00 ps
Baud Rate = 59.8437500000 GBd
Modulation Type: 16QAM
Chromatic Dispersion 1 ps/nm
Configured CD-MIN -2500 ps/nm CD-MAX 2500 ps/nm
Second Order Polarization Mode Dispersion = 39.00 ps^2
Optical Signal to Noise Ratio = 36.40 dB
Polarization Dependent Loss = 0.07 dB
Polarization Change Rate = 0.00 rad/s
Differential Group Delay = 5.00 ps
```

.

Temperature = 51.94 Celsius Voltage = 3.31 V

Transceiver Vendor Details

_ .

Form Factor	: QSFP-DD
Optics type	: QSFP-DD 400G-ZR-S
Name	: CISCO-ACACIA
OUI Number	: 7c.b2.5c
Part Number	: DP04QSDD-E20-190
Rev Number	: A
Serial Number	: ACA25220027
PID	: QDD-400G-ZR-S
VID	: V01
Firmware Version	: 61.20 (Build : 13)
Date Code(yy/mm/dd)	: 21/06/01

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. User 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*.

Optical Module	Trunk Optical Speed Transmit Bower (Tx)	Optical Transmit Bower (Tx)	Interval	Supported Range of Optical Transmit Power (Tx) Values (in units of 0.1dBm) ¹		
		Shaping		Minimum Value	Maximum Typical Value	Maximum Worst CaseValue
QDD-400G-ZR-S	400G	No	1	-150	-100	-100
QDD-400G-ZRP-S	400G	Yes	1	-150	-110	-130
				-150	-104	-119
				-150	-90	-105
				-150	-59	-75
DP04QSDD-HE0	400G	Yes	1	-150	15	25
	300G	-				
	200G					
	100G					

Table 23: Optical Transmit Power Values

1. The default optical transmit power (Tx) value is -10 dBm, however with TX shaping enabled the maximum power in 1x400G, 4x100G, 3x100G, and 2x100G modes may be less than -10 dBm.

Transmitting Power Configuration Example

The following example shows how to configure the optical transmit (TX) power on the optics controller:

```
Router#config
Router(config)#controller optics 0/19/0/3
Router(config-Optics)#transmit-power -70
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 0/19/0/3
Thu May 13 12:52:35.020 UTC
controller Optics0/19/0/3
cd-min -4000
cd-max 4000
transmit-power -70
```

Verification

This example shows how to verify the configured optical transmit power for the optics controller:

```
Router#show controller optics 0/19/0/3
Controller State: Up
Transport Admin State: In Service
Laser State: On
LED State: Green
FEC State: FEC OFEC
```

Optics Status Optics Type: 400G QSFP-DD 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: Laser Bias Current = 70.9 mA Actual TX Power = -7.06 dBm RX Power = -7.88 dBm RX Signal Power = -7.88 dBm Frequency Offset = -220 MHz Laser Temperature = 53.79 Celsius Laser Age = 0 % DAC Rate = 1x1.25Performance Monitoring: Enable THRESHOLD VALUES High Alarm Low Alarm High Warning Low Warning Parameter
 Rx Power Threshold(dBm)
 1.9
 -28.2
 0.0
 -25.0

 Tx Power Threshold(dBm)
 0.0
 -18.0

 LBC Threshold(mA)
 0.00
 0.00

 Temp. Threshold(celsius)
 80.00
 -5.00

 Voltage Threshold(volt)
 3.46
 3.13
 -16.0 -2.0 0.00 0.00 -5.00 3.13 75.00 15.00 3.43 3.16 LBC High Threshold = 98 % Configured Tx Power = -7.00 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 = 180.00 ps Baud Rate = 30.0692729950 GBd Modulation Type: QPSK Chromatic Dispersion -2 ps/nm Configured CD-MIN -4000 ps/nm CD-MAX 4000 ps/nm Second Order Polarization Mode Dispersion = 109.00 ps^2 Optical Signal to Noise Ratio = 32.30 dB Polarization Dependent Loss = 0.05 dB Polarization Change Rate = 0.00 rad/s Differential Group Delay = 0.00 ps Temperature = 48.00 Celsius Voltage = 3.30 V Transceiver Vendor Details : QSFP-DD Form Factor Optics type : QSFP-DD 400G-ZRP-S Name : CISCO-ACACIA : 7c.b2.5c OUI Number Part Number : DP04QSDD-E30-19E : 01 Rev Number Serial Number : ACA2503003X

Interface and Hardware Component Configuration Guide for Cisco ASR 9000 Series Routers, IOS XR Release 7.10.x

: QDD-400G-ZRP-S

: ES03

Firmware Version : 61.20 (Build : 13) Date Code(yy/mm/dd) : 21/01/22

PID

VTD

Firmware Version

Configuring Performance Monitoring

Performance monitoring (PM) parameters are used by service providers to gather, store, set thresholds for, and report performance data for early detection of problems. The user can retrieve both current and historical PM counters for the various controllers in 30-second, 15-minute, and 24-hour intervals.

Performance monitoring can be configured on optics controllers and coherent DSP controllers.

To stop performance monitoring on optics or coherent DSP controllers, use the perf-mon disable keyword.

Configuring PM Parameters

The performance monitorning (PM) threshold and the threshold crossing alert (TCA) reporting status can be configured for optics controllers and coherent DSP 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.

Table 24: PM Thresholds and TCA Report Status for Optics Controllers

PM Parameters	Description
SOPMD	Sets the SOPMD (second order polarization mode dispersion) threshold or TCA reporting status.

Table 25: PM Thresholds TCA Report Status for Coherent DSP Controllers

PM Parameters	Description
Q	Sets the Q threshold or TCA reporting status.
Q-margin	Sets the Q margin threshold or TCA reporting status.
EC-BITS	Sets the EC-BITS (error corrected bits) threshold or TCA reporting status.
PostFEC BER	Sets the post-FEC BER threshold or TCA reporting status.
PreFEC BER	Sets the pre-FEC BER threshold or TCA reporting status.
UC-WORDS	Sets the UC-WORDS (uncorrected words) threshold or TCA reporting status.

Performance Monitoring Configuration Example

This example shows how to enable performance monitoring and set PM thresholds on the optics controller:

```
Router#config
Router(config)#controller optics 0/2/0/16
Router(config-Optics)#perf-mon enable
Router(config-Optics)#pm 30-sec optics threshold cd max 100
Router(config-Optics)#pm 30-sec optics threshold cd min -100
Router(config-Optics)#commit
```

Running Configuration

This example shows the running configuration on optics controllers:

```
Router#show run controller optics 0/2/0/16
Thu May 13 20:18:55.957 UTC
controller Optics0/2/0/16
pm 30-sec optics threshold cd max 100
pm 30-sec optics threshold cd min -100
perf-mon enable
```

Verification

This example shows how to verify the PM parameters on optics controllers. Verify the configuration changes in the Configured Threshold fields:

```
Router#show controller optics 0/2/0/16 pm current 30-sec optics 1

Thu May 27 17:58:49.889 UTC

Optics in the current interval [17:58:30 - 17:58:49 Thu May 27 2021]

Optics current bucket type : Valid

MIN AVG MAX Operational Configured TCA Operational

Configured TCA

Threshold(min) Threshold(min) (min) Threshold(max)
```

Threshol	Ld (n	nax) (max)						
LBC[mA]	NA	: 0.0	0.0 NO	0.0	0.0	NA		NO	100.0
OPT[dBm] N	: JA	-9.98	-9.98 NO	-9.98	-15.09	NA		NO	0.00
OPR[dBm]	: IA	-40.00	-40.00 NO	-40.00	-30.00	NA		NO	8.00
CD[ps/nm] 100	:	: 0 NO	0	0	-80000	-100	NO	100	
DGD[ps] N	: IA	0.00	0.00 NO	0.00	0.00	NA		NO	80.00
SOPMD[ps^2]	: JA	0.00	0.00 NO	0.00	0.00	NA		NO	2000.00
OSNR[dB] N	: JA	0.00	0.00 NO	0.00	0.00	NA		NO	40.00
PDL[dB] N	: JA	0.00	0.00 NO	0.00	0.00	NA		NO	7.00
PCR[rad/s] NA	:	0.00	0.00 NO	0.00	0.00	NA	NO	250	00000.00
RX_SIG[dBm]	1 : 1 A	-40.00	-40.00 NO	-40.00	-30.00	NA		NO	1.00
FREQ_OFF[Mh	nz]: JA	: 0	0 NO	0	-3600	NA		NO	3600
SNR[dB] N	IA A	0.00	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 min-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 min 900
Router (config-CoDSP) #pm 30-sec fec threshold Q-margin min 280
Router (config-CoDSP) #pm 30-sec fec threshold Q-margin min 280
```

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
                                           AVG
                                                    MAX
                                  MTN
                                                              Threshold
                                                                           TCA
 Threshold
            TCA
                                                                 (min)
                                                                         (enable)
           (enable)
   (max)
                                 0E-15
                                           0E-15
                                                    0E-15
                                                              0E-15
PreFEC BER
                            :
                                                                          NO
0E - 15
           NO
PostFEC BER
                            :
                                 0E-15
                                          0E-15 0E-15
                                                              0E-15
                                                                          NO
           NO
0E-15
                                       0.00 0.00
                                  0.00
                                                              9.00 YES 120.00 YES
Q[dB]
                            :
                                  0.00
                                           0.00
                                                   0.00
                                                              2.80 YES 5.00
Q_Margin[dB]
                                                                              YES
                            :
```

Configuring Loopback

You can configure internal or line loopback on coherent DSP controllers. Loopback can be performed only in the maintenance mode.

Loopback Configuration Example

This example shows how to enable internal loopback configuration on coherent DSP controllers:

```
Router#config
Router(config)#controller coherentDSP 0/0/0/4
Router(config-CoDSP)#secondary-admin-state maintenance
Router(config-CoDSP)#loopback internal
Router(config-CoDSP)#commit
```

Running Configuration

This example shows the running configuration on coherent DSP controllers:

```
Router#show run controller coherentdsp 0/0/0/4
controller CoherentDSP0/0/0/4
secondary-admin-state maintenance
loopback internal
!
```

Verification

This example shows how to verify the loopback configuration on coherent DSP controllers:

```
Router#show controller coherentdsp 0/0/0/4
Thu May 27 17:28:51.960 UTC
                                                : CoherentDSP 0/0/0/4
Port
Controller State
                                                : Down
Inherited Secondary State
                                                : Normal
Configured Secondary State
                                             : Maintenance
Derived State
                                                : Maintenance
Loopback mode
                                              : Internal
                                                : SF = 1.0E-5 SD = 1.0E-7
BER Thresholds
```

Performance Monitoring : Enable : 400.0Gb/s Bandwidth Alarm Information: LOS = 6 LOF = 0 LOM = 0OOF = 0 OOM = 0 AIS = 0FLEXO-MM = 0 FLEXO-LOM = 0 FLEXO-RDI = 0 FLEXO-LOF = 5Detected Alarms : LOS Bit Error Rate Information PREFEC BER : 5.0E-01 POSTFEC BER : 0.0E+00 Q-Factor : 0.00 dB : -7.20dB Q-Margin OTU TTI Received FEC mode : C_FEC
Laser Squelching

Table 26: Feature History Table

Feature Name	Release Information	Description
Disable Auto-Squelching	Release 7.10.1	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.
		The feature introduces these changes:
		CLI:
		• The controller optics command is modified to support a new optional keyword, host auto-squelch disable
		YANG DATA models:
		• New XPaths for Cisco-ICS-XR-controller-aptics-cfg (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 output 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
```

Small Frame Padding

Table 27: Feature History Table

Feature Name	Release Information	Description
Small Frame Padding	Release 7.10.1	Small frame padding prevents packet drops caused by network congestion. If the minimum frame size requirement isn't met, the frames drop, this enhancement ensures that your hardware ASIC adds extra bytes to the payload, thereby fulfilling the 68-byte minimum frame size requirement. By doing so, small frame padding significantly enhances network reliability and minimizes the risk of dropped frames due to congestion-related issues. Previously, this feature was supported on the second and third generations of the ASR 9000 Series Ethernet line cards. Starting from this release, we extend this feature support to the fourth and fifth generations of the ASR 9000 Series Ethernet line cards.

Small frame padding refers to the technique of adding extra bytes to a transmitted data frame to ensure that the frame meets a minimum size requirement. In an ethernet based data transmission, a minimum of 68 bytes frame size is required to ensure reliable transmission and reception of data. If the minimum Ethernet frame

length requirement isn't met, the frames are treated as runt frames and dropped. This feature enables the Cisco ASR 9000 routers to add frames to meet the transmission requirement of 68 bytes on the egress interface to seamlessly work with other interconnected networks.

Implementing small frame padding offers several key benefits:

- Enhances network reliability: Small frame padding ensures that the minimum frame size is met, thereby helping to prevent data loss or corruption caused by network congestion, which can result in dropped frames.
- Enhances network security: Small frame padding obscures the true size of data frames, making it difficult for attackers to identify patterns in network traffic, thereby contributing to a more secure network.
- Improves performance: Small frame padding ensures more frequent transmission of frames, actively
 utilizing communication channels more often and optimizing network resource usage.

Small frame padding is supported on the following line cards:

- Second generation of the ASR 9000 Series Ethernet line cards
- Third generation of the ASR 9000 Series Ethernet line cards
- Fourth generation of the ASR 9000 Series Ethernet line cards
- Fifth generation of the ASR 9000 Series Ethernet line cards

Configuring small frame padding

This example shows how to configure the small-frame-padding command:

```
router(config)# interface HundredGigE 0/1/0/0
router(config-if)# small-frame-padding
```

Verification

This example shows how to verify the small-frame-padding configuration:

```
router(config)# show controller asic ls-np instance 0 76 register location 0/1/CPU0 | i
HexPad
```

0x010480580 elf1Reg_elf1Sf0Cfg_elf1HexPadCfg 0x00000001 (4 bytes) 0x010480780 elf1Reg_elf1Sf1Cfg_elf1HexPadCfg 0x00000000 (4 bytes)

The NPU registers contain configurations for physical port numbers.

- For 'elf1Reg_elf1Sf0Cfg', bits 0–19 are for physical port numbers ranging 0–19.
- For 'elf1Reg_elf1Sf1Cfg', the bits 0–19 are for physical port numbers ranging from 20 to 39.

In the given example, HundredGigE 0/1/0/0 has a physical port number of 0. Hence, bit 0 on 'elf1Reg elf1Sf0Cfg' is set to 1 after the small-frame-padding configuration.

Configuring Alarms Threshold

The alarms threshold can be configured for monitoring alarms on optics controllers:

Table 28: 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 -5000
Router(config-Optics)#cd-high-threshold 5000
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
controller Optics0/2/0/16
cd-low-threshold 5000
cd-high-threshold 5000
```

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 OFEC
Optics Status
        Optics Type: 400G QSFP-DD 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
                                  OSNR = 0
        OOR-CD = 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 = 1×1.25 Performance Monitoring: Enable THRESHOLD VALUES _____ Parameter High Alarm Low Alarm High Warning Low Warning ----- -----_____ _____ _____ 13.0 -24.0 10.0 Rx Power Threshold(dBm) -22.0 0.0 -16.0 0.00 0.00 80.00 -5.00 -14.0 Tx Power Threshold(dBm) -2.0 0.00 LBC Threshold(mA) 0.00 0.00 Temp. Threshold(celsius) 80.00 -5.00 75.00 0.00 3.13 3.43 Voltage Threshold(volt) 3.46 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 S

Optics type	: QSFP-DD 400G-ZRP-S
Name	: CISCO-ACACIA
OUI Number	: 7c.b2.5c
Part Number	: DP04QSDD-E30-19E
Rev Number	: 01
Serial Number	: ACA2503003X
PID	: QDD-400G-ZRP-S
VID	: ES03
Firmware Version	: 61.20 (Build : 13)
Date Code(yy/mm/dd)	: 21/01/22



Configuring 802.10 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, and 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 802.10 VLAN Interfaces

Release	Modification
Release 3.7.2	This feature was introduced on the Cisco ASR 9000 Series Router.
Release 3.9.0	Layer 2 dot1q was updated. Encapsulation dot1q was added.

• Prerequisites for Configuring 802.1Q VLAN Interfaces, on page 417

- Information About Configuring 802.1Q VLAN Interfaces, on page 418
- How to Configure 802.1Q VLAN Interfaces, on page 420
- Configuration Examples for VLAN Interfaces, on page 427

Prerequisites for Configuring 802.10 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, be sure that the following conditions are met:

• You must have configured a Gigabit Ethernet interface, a 10-Gigabit Ethernet interface, or an Ethernet bundle interface.

Information About Configuring 802.10 VLAN Interfaces

To configure 802.1Q VLAN interfaces, you must understand the following concepts:

802.10 VLAN Overview

A VLAN is a group of devices on one or more LANs that are configured so that they can communicate as if they were attached to the same wire, when in fact they are located on a number of different LAN segments. Because VLANs are based on logical instead of physical connections, they are very 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 Gigabit Ethernet and 10-Gigabit Ethernet interfaces.

802.10 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.

CFM on 802.10 VLAN Interfaces

Configuring Connectivity Fault Management (CFM) for monitoring 802.1Q VLAN interfaces is identical to configuring CFM for monitoring Ethernet interfaces.

For information on configuring CFM for Ethernet interfaces, refer to the following sections in the Configuring Ethernet OAM, on page 81 module:

Subinterfaces

Subinterfaces are logical interfaces created on a hardware interface. These software-defined interfaces allow for segregation of traffic into separate logical channels on a single hardware interface as well as allowing for better utilization of the available bandwidth on the physical interface.

Subinterfaces are distinguished from one another by adding an extension on the end of the interface name and designation. For instance, the Ethernet subinterface 23 on the physical interface designated TenGigE 0/1/0/0 would be indicated by TenGigE 0/1/0/0.23.

Before a subinterface is allowed 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, the VLAN identifier must be explicitly defined.

Subinterface MTU

The subinterface maximum transmission unit (MTU) is inherited from the physical interface with an additional four bytes allowed for the 802.1Q VLAN tag.

Native VLAN

The Cisco ASR 9000 Series Router does not support a native VLAN. However, the equivalent functionality is accomplished using an **encapsulation** command as follows:

encapsulation dot1q TAG-ID, untagged

EFPs

An Ethernet Flow Point (EFP) is a Metro Ethernet Forum (MEF) term describing abstract router architecture. On the Cisco ASR 9000 Series Router, an EFP is implemented by an L2 subinterface with a VLAN encapsulation. The term EFP is used synonymously with an VLAN tagged L2 subinterface.

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 **l2transport** 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—The AC covers all frames received and sent with a specific outer VLAN tag and a specific inner VLAN tag. QinQ is an extension to Dot1Q that uses a stack of two tags.
- Q-in-Any AC—The AC covers all frames received and sent with a specific outer VLAN tag and any inner VLAN tag, as long as that inner VLAN tag is not L3 terminated. Q-in-Any is an extension to QinQ that uses wildcarding to match any second tag.



Note]

The Q-in-Any mode is a variation of the basic Dot1Q mode. In Q-in-Any mode, the frames have a basic QinQ encapsulation; however, in Q-in-Any mode the inner tag is not relevant, except for the fact that a few specific inner VLAN tags are siphoned for specific services. For example, a tag may be used to provide L3 services for general internet access.

Each VLAN on a CE-to-PE link can be configured as a separate L2VPN connection (using either VC type 4 or VC type 5). To configure L2VPN on VLANs, see the Configuring an Attachment Circuit on a VLAN section.

Keep the following in mind when configuring L2VPN on a VLAN:

- Cisco IOS XR software supports 4k ACs per LC.
- In a point-to-point connection, the two ACs do not have to be of the same type. For example, a port mode Ethernet AC can be connected to a Dot1Q Ethernet AC.
- Pseudowires can run in VLAN mode or in port mode. A pseudowire running in VLAN mode has a single Dot1Q tag, while a pseudo-wire running in port mode has no tags. Some interworking is required to connect these different types of circuits together. This interworking takes the form of popping, pushing, and rewriting tags. The advantage of Layer 2 VPN is that is simplifies the interworking required to connect completely different media types together.
- The ACs on either side of an MPLS pseudowire can be different types. In this case, the appropriate conversion is carried out at one or both ends of the AC to pseudowire connection.

Use the show interfaces command to display AC and pseudowire information.

Note

For detailed information about configuring an L2VPN network, see the "Implementing MPLS Layer 2 VPNs" VPNsmodule of the Cisco IOS XR

Other Layer 2 VPN Features

For information on the following Layer 2 VPN features, refer to the *Cisco ASR 9000 Series Aggregation* Services Router L2VPN and Ethernet Services Configuration Guide and the Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Command Reference:

- Provider Backbone Bridge (PBB) 802.1ah
- Policy-Based Forwarding (PBF)
- MVRP 802.1 (MVRP-lite)

How to Configure 802.10 VLAN Interfaces

This section contains the following procedures:

Configuring 802.10 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.

L

\mathcal{P}

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 ASR 9000 Series Routers*.

SUMMARY STEPS

- 1. configure
- 2. interface {GigabitEthernet | TenGigE | Bundle-Ether} interface-path-id.subinterface
- **3**. encapsulation dot1q
- 4. ipv4 address ip-address mask
- 5. exit
- 6. Repeat Step 2 through Step 5 to define the rest of the VLAN subinterfaces.
- 7. end or commit
- 8. show ethernet trunk bundle-ether *instance*
- **9**. **show vlan interface** [type interface-path-id][**location** instance]
- **10.** show vlan trunks [brief] [location *instance*] [{GigabitEthernet | TenGigE | Bundle-Ether | } *interface-path-id*] [summary]

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	interface {GigabitEthernet TenGigE Bundle-Ether} interface-path-id.subinterface	Enters subinterface configuration mode and specifies the interface type, location, and subinterface number.
	Example:	• Replace the <i>interface-path-id</i> argument with one of the following instances:
	<pre>RP/0/RSP0/CPU0:router(config)# interface TenGigE 0/2/0/4.10</pre>	• 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>interface-path-id.subinterface</i> , and a period between arguments is required as part of the notation.

	Command or Action	Purpose
Step 3	encapsulation dot1q	Sets the Layer 2 encapsulation of an interface.
	<pre>Example: RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 100, untagged</pre>	Note • The dot1q vlan command is replaced by the encapsulation dot1q command on the Cisco ASR 9000 Series Router. It is still available for backward-compatibility, but only for Layer 3 interfaces.
Step 4	ipv4 address ip-address mask	Assigns an IP address and subnet mask to the subinterface.
	Example: RP/0/RSP0/CPU0:router(config-subif)# ipv4 address 178.18.169.23/24	 Replace <i>ip-address</i> with the primary IPv4 address for an interface. Replace <i>mask</i> 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 5	exit	(Optional) Exits the subinterface configuration mode.
	Example:	• The exit command is not explicitly required.
	RP/0/RSP0/CPU0:router(config-subif)# exit	
Step 6	Repeat Step 2 through Step 5 to define the rest of the VLAN subinterfaces.	
Step 7	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	or	Uncommitted changes found, commit them before
		<pre>exiting(yes/no/cancel)? [cancel]:</pre>
	RP/0/RSP0/CPU0:router(config)# commit	- Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		- Entering no exits the configuration session and returns the router to EXEC mode without committing the configuration changes.

	Command or Action	Purpose
		- Entering cancel leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the commit command to save the configuration changes to the running configuration file and remain within the configuration session.
Step 8	show ethernet trunk bundle-ether instance	(Optional) Displays the interface configuration.
	Example:	The Ethernet bundle instance range is from 1 through 65535.
	RP/0/RSP0/CPU0:router# show ethernet trunk bundle-ether 5	
Step 9	show vlan interface [type interface-path-id][location <i>instance</i>]	(Optional) Displays the interface configuration.
	Example:	the location keyword.
	RP/0//CPU0:router# show vlan interface 5	• To display the configuration for the specified interface or subinterface, use the interface keyword.
Step 10	<pre>show vlan trunks [brief] [location instance] [{GigabitEthernet TenGigE Bundle-Ether }</pre>	(Optional) Displays summary information about each of the VLAN trunk interfaces.
	interface-path-id] [summary]	• The keywords have the following meanings:
	Example:	• brief —Displays a brief summary.
	RP/0//CPU0:router# show vlan trunk summary	• summary —Displays a full summary.
		• location —Displays information about the VLAN trunk interface on the given port.
		• interface —Displays information about the specified interface or subinterface.

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 [GigabitEthernet | TenGigE | Bundle-Ether | TenGigE] interface-path] id.subinterface l2transport
- 3.
- 4. l2protocol cpsv {tunnel | reverse-tunnel}
- 5. end or commit
- 6. show interfaces [GigabitEthernet | TenGigE] interface-path-id.subinterface

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure terminal	
Step 2	interface [GigabitEthernet TenGigE Bundle-Ether TenGigE] interface-path] id.subinterface l2transport	Enters subinterface configuration and specifies the interface type, location, and subinterface number.
	Example:	• Replace the argument with one of the following instances:
	RP/0/RSP0/CPU0:router(config)# interface TenGigE 0/1/0/0.1 l2transport	• 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.
		• You must include the l2transport keyword in the command string; otherwise, the configuration creates a Layer 3 subinterface rather that an AC.
Step 3	Example:	Sets the Layer 2 encapsulation of an interface.
	RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 100, untagged	Note • The dot1q vlan command is replaced by the encapsulation dot1q command on the Cisco ASR 9000 Series Router. It is still available for backward-compatibility, but only for Layer 3 interfaces.
Step 4	12protocol cpsv {tunnel reverse-tunnel}	Configures Layer 2 protocol tunneling and protocol data
	Example:	unit (PDU) filtering on an Ethernet interface for the following protocols: CDP, PVST+, STP, VTP, where:
	RP/0/RSP0/CPU0:router(config-if-12)# l2protocol cpsv tunnel	• tunnel —Specifies L2PT encapsulation on frames as they enter the interface, and de-encapsulation on frames as they exit they interface.
Step 3 Step 4	<pre>Example: RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 100, untagged I2protocol cpsv {tunnel reverse-tunnel} Example: RP/0/RSP0/CPU0:router(config-if-12)# 12protocol cpsv tunnel</pre>	 You must include the l2transport keyword in the command string; otherwise, the configuration creater a Layer 3 subinterface rather that an AC. Sets the Layer 2 encapsulation of an interface. Note The dot1q vlan command is replaced by the encapsulation dot1q command on the Cisco ASR 9000 Series Router. It is still available for backward-compatibility, but only for Layer 3 interfaces. Configures Layer 2 protocol tunneling and protocol datuit (PDU) filtering on an Ethernet interface for the following protocols: CDP, PVST+, STP, VTP, where: tunnel—Specifies L2PT encapsulation on frames they enter the interface, and de-encapsulation on frames as they exit they interface.

	Command or Action	Purpose
		• reverse-tunnel —Specifies L2PT encapsulation on frames as they exit the interface, and de-encapsulation on frames as they enter the interface.
Step 5	end or commit	Saves configuration changes.
	Example: RP/0/RSP0/CPU0:router(config-if-l2)# end	• When you issue the end command, the system prompts you to commit changes:
	or	Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:
	RP/0/RSP0/CPU0:router(config-if-12)# commit	- 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 6	show interfaces [GigabitEthernet TenGigE] <i>interface-path-id.subinterface</i>	(Optional) Displays statistics for interfaces on the router.
	Example:	
	RP/0/RSP0/CPU0:router# show interfaces TenGigE 0/3/0/0.1	

What to do next

- To configure a point-to-point pseudowire cross connect on the AC, see the "Implementing MPLS Layer 2 VPNs" VPNs module of the Cisco ASR 9000 Series Router Multiprotocol Label Switching Configuration Guide.
- To attach Layer 3 service policies, such as Multiprotocol Label Switching (MPLS) or Quality of Service (QoS), to the VLAN, refer to the appropriate Cisco ASR 9000 Series Router software configuration guide.

Removing an 802.10 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.

SUMMARY STEPS

- 1. configure
- 2. no interface {GigabitEthernet | TenGigE | Bundle-Ether] interface-path-id.subinterface
- 3. Repeat Step 2 to remove other VLAN subinterfaces.
- 4. end or commit
- 5. show vlan interface [{GigabitEthernet | TenGigE | Bundle-Ether} interface-path-id | location instance]

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	configure	Enters global configuration mode.
	Example:	
	RP/0/RSP0/CPU0:router# configure	
Step 2	no interface {GigabitEthernet TenGigE Bundle-Ether] <i>interface-path-id.subinterface</i>	Removes the subinterface, which also automatically deletes all the configuration applied to the subinterface.
	Example:	• Replace the <i>instance</i> argument with one of the following instances:
	RP/0/RSP0/CPU0:router(config)# no interface TenGigE 0/2/0/4.10	• 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.
Step 3	Repeat Step 2 to remove other VLAN subinterfaces.	—
Step 4	end or commit	Saves configuration changes.
	Example:	• When you issue the end command, the system prompts you to commit changes:
	RP/0/RSP0/CPU0:router(config)# end	
	or	<pre>Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:</pre>
	RP/0/RSP0/CPU0:router(config)# commit	- Entering yes saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.

L

	Command or Action	Purpose
		- 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 vlan interface [{GigabitEthernet TenGigE	(Optional) Displays the interface configuration.
B	Bundle-Ether } interface-path-id location instance]	• To display the configuration for a port, use the location
	Example:	keyword.
	RP/0//CPU0:router# show vlan trunk summary	• To display the configuration for the specified interface or subinterface, use the interface keyword.
		1

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/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config) # interface TenGigE 0/2/0/4.1
RP/0/RSP0/CPU0:router(config-subif) # encapsulation dot1q 100
RP/0/RSP0/CPU0:router(config-subif)# ipv4 address 10.0.10.1/24
RP/0/RSP0/CPU0:router(config-subif) # interface TenGigE0/2/0/4.2
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 101
RP/0/RSP0/CPU0:router(config-subif)# ipv4 address 10.0.20.1/24
RP/0/RSP0/CPU0:router(config-subif)# interface TenGigE0/2/0/4.3
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 102
RP/0/RSP0/CPU0:router(config-subif)# ipv4 address 10.0.30.1/24
RP/0/RSP0/CPU0:router(config-subif) # commit
RP/0/RSP0/CPU0:router(config-subif) # exit
RP/0/RSP0/CPU0:router(config)# exit
RP/0/RSP0/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//CPU0:router#	show vla	n inter	face Interface	St Ly	MTU	Subs	L2
L3 Up D	own Ad-Dow	n					
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//CPU0:router#	show vla	n trunk	s briefBE1	Up :	L3 151	14 1000	0
1000 1000	0	0					
Summary		1000	0 1000	1000	0	0	
Te0/2/0/4	802.1Q (E	ther)	up				

The following example shows how to create two VLAN subinterfaces on an Ethernet bundle:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface bundle-ether 2
RP/0/RSP0/CPU0:router(config-if)# ipv4 address 192.168.2.1/24
RP/0/RSP0/CPU0:router(config-if)# exit
RP/0/RSP0/CPU0:router(config)# interface bundle-ether 2.1
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 100
RP/0/RSP0/CPU0:router(config-subif)# ipv4 address 192.168.100.1/24
RP/0/RSP0/CPU0:router(config-subif)# exit
RP/0/RSP0/CPU0:router(config-subif)# exit
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 200
RP/0/RSP0/CPU0:router(config-subif)# ipv4 address 192.168.200.1/24
RP/0/RSP0/CPU0:router(config-subif)# ipv4 address 192.168.200.1/24
RP/0/RSP0/CPU0:router(config-subif)# ipv4 address 192.168.200.1/24
RP/0/RSP0/CPU0:router(config-subif)# ipv4 address 192.168.200.1/24
RP/0/RSP0/CPU0:router(config-subif)# exit
RP/0/RSP0/CPU0:router(config-subif)# exit
RP/0/RSP0/CPU0:router(config-subif)# exit
```

The following example shows how to create a basic dot1Q AC:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/0/0/0.1
RP/0/RSP0/CPU0:router(config-subif)# l2transport
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 100
RP/0/RSP0/CPU0:router(config-subif)# commit
RP/0/RSP0/CPU0:router(config-subif)# exit
RP/0/RSP0/CPU0:router(config-subif)# exit
```

The following example shows how to create a Q-in-Q AC:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/0/0/0.2
RP/0/RSP0/CPU0:router(config-subif)# l2transport
```

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

```
RP/0/RSP0/CPU0:router(config-subif)# commit
RP/0/RSP0/CPU0:router(config-subif)# exit
RP/0/RSP0/CPU0:router(config)# exit
```

The following example shows how to create a Q-in-Any AC:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/0/0/0.3
RP/0/RSP0/CPU0:router(config-subif)# l2transport
RP/0/0/CPU0:router(config-subif)# dot1q vlan 30 vlan any
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 300 second-dot1q any
RP/0/RSP0/CPU0:router(config-subif)# commit
RP/0/RSP0/CPU0:router(config-subif)# exit
RP/0/RSP0/CPU0:router(config-subif)# exit
```



Managing Router Hardware

This chapter describes the command-line interface (CLI) techniques and commands used to manage and configure the hardware components of a router running the Cisco IOS XR software.

For complete descriptions of the commands listed in this module, see Additional References, on page 480. To locate documentation for other commands that might appear in the course of performing a configuration task, search online in *Cisco ASR 9000 Series Aggregation Services Router Commands Master List*.

Table 29: Feature History for Managing Router Hardware with Cisco IOS XR Software

Release	Modification
Release 3.7.2	This feature was introduced.

This module contains the following topics:

- Prerequisites for Managing Router Hardware, on page 432
- Displaying Hardware Status, on page 432
- RSP Redundancy and Switchover, on page 448
- Configuring Breakout on a 400GE Port, on page 453
- Reloading, Shutting Down, or Power Cycling a Node, on page 457
- Flash Disk Recovery, on page 459
- Using Controller Commands to Manage Hardware Components, on page 460
- Formatting Hard Drives, Flash Drives, and Other Storage Devices, on page 460
- Removing and Replacing Cards, on page 461
- Proactive Line Card Shut Down, on page 463
- Advanced Power Management, on page 465
- Overview of Erase and Wipeout Disk Memory, on page 467
- Upgrading the CPU Controller Bits, on page 470
- Configuring Port Modes, on page 470
- Configure Single Feed Power Mode, on page 479
- Additional References, on page 480

Prerequisites for Managing Router Hardware

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.

Displaying Hardware Status

This section describes how to display different types of hardware status information.

Displaying SDR Hardware Version Information

To display hardware version information for the components assigned to a secure domain router (SDR), connect to the designated shelf controller (DSC) and enter the **show diag** command in EXEC mode. The displayed information includes the card serial number and the ROMMON software version.

The syntax for the show diag command in EXEC mode is:

show diag [node-id | details | summary]

In the following example, the show diag command displays information for all nodes in the SDR:

```
RP/0/RSP0/CPU0:router# show diag
Mon Jun 29 00:36:41.576 PST
NODE module 0/RSP0/CPU0 :
  MAIN: board type 0x100302
  S/N: FOC1230803H
  Top Assy. Number: 68-3160-04
  PID: A2K-RSP-4G-HDD=
  UDI VID: VP4
  HwRev: V4.8
  New Deviation Number: 0
  CLEI: IPUCARJBAA
  Board State : IOS XR RUN
       Motherboard: N/A, Processor: 0x8004 (rev: 2.2), Power: N/A
  PLD:
 MONLIB: ONXFFS Monlib Version 3.2
  ROMMON: Version 1.0 (20081208:173612) [ASR9K ROMMON]
  Board FPGA/CPLD/ASIC Hardware Revision:
    Compact Flash : V1.0
    XbarSwitch0 : V1.3
    XbarSwitch1 : V1.3
    XbarArbiter : V1.0
    XbarInterface : V0.0
    IntCtrl : V1.14
    ClkCtrl : V1.13
     PuntFPGA : V1.5
    HD : V3.0
    USB0 : V77.20
    USB1 : V77.20
    CPUCtrl : V1.17
     UTI : V1.6
    LIU : V1.0
```

```
MLANSwitch : V0.0
    EOBCSwitch : V2.0
    CBC (active partition) : v1.2
    CBC (inactive partition) : v1.1
NODE module 0/1/CPU0 :
 MAIN: board type 0x20207
  S/N: FOC123081J6
                     68-3182-03
  Top Assy. Number:
  PID: A9K-40GE-B
  UDI VID: V1D
  HwRev: V0.0
  New Deviation Number: 0
  CLET:
  Board State : IOS XR RUN
  PLD:
         Motherboard: N/A, Processor: 0x8004 (rev: 2.2), Power: N/A
  ROMMON: Version 1.0(20081208:174521) [ASR9K ROMMON]
  Board FPGA/CPLD/ASIC Hardware Revision:
    NPO : V3.194
    NP1 : V3.194
    NP2 : V3.194
    NP3 : V3.194
    XbarInterface : V18.4
    Bridge0 : V0.38
    Bridgel : V0.38
    CPUCtrl : V0.15
     USB : V77.20
    PortCtrl : V0.8
    PHYCtrl : V0.6
     40 Port Gigabit Ethernet Daughter board : V0.0
    CBC (active partition) : v2.2
    CBC (inactive partition) : v2.1
NODE module 0/4/CPU0 :
  MAIN: board type 0x2020a
       FOC123081JA
  S/N:
  Top Assy. Number:
                     68-3183-02
  PID: A9K-8T/4-B
  UDI VID: V1D
  HwRev: V0.0
  New Deviation Number: 0
  CLEI: IPU3AE0CAA
  Board State : IOS XR RUN
  PLD: Motherboard: N/A, Processor: 0x8004 (rev: 2.2), Power: N/A
  ROMMON: Version 1.0(20081208:174521) [ASR9K ROMMON]
  Board FPGA/CPLD/ASIC Hardware Revision:
    NPO : V3.194
    NP1
         : V3.194
    NP2 : V3.194
    NP3 : V3.194
    XbarInterface : V18.4
    Bridge0 : V0.38
    Bridgel : V0.38
CPUCtrl : V0.15
    USB : V77.20
     PortCtrl : V0.10
    PHYCtrl : V0.7
     PHY0 : V0.16
     PHY1
          : V0.16
          : V0.16
     PHY2
    PHY3 : V0.16
     PHY4 : V0.16
```

```
PHY5 : V0.16
     PHY6 : V0.16
     PHY7 : V0.16
     8 Port Ten Gigabit Ethernet Daughter board : V0.0
     CBC (active partition) : v2.2
     CBC (inactive partition) : v2.1
NODE module 0/6/CPU0 :
 MAIN: board type 0x20208
  S/N: FHH12250033
  Top Assy. Number:
                      68-3184-02
  PID: A9K-4T-B
  UDI VID: V1D
 HwRev: V0.0
 New Deviation Number: 0
  CLEI:
  Board State : IOS XR RUN
  PLD: Motherboard: N/A, Processor: 0x8004 (rev: 2.2), Power: N/A
  ROMMON: Version 1.0(20081208:174521) [ASR9K ROMMON]
  Board FPGA/CPLD/ASIC Hardware Revision:
    NPO : V3.194
     NP1 : V3.194
    NP2 : V3.194
    NP3 : V3.194
    XbarInterface : V18.4
    Bridge0 : V0.38
    Bridgel : V0.38
CPUCtrl : V0.15
     USB : V77.20
     PHY0 : V0.16
     PHY1 : V0.16
    PHY2 : V0.16
PHY3 : V0.16
     PortCtrl : V0.10
     PHYCtrl : V0.7
     4 Port Ten Gigabit Ethernet Daughter board : V0.0
     CBC (active partition) : v2.2
     CBC (inactive partition) : v2.1
```

In the following example, the **show diag** command displays information for a single node:

```
RP/0/RSP0/CPU0:router# show diag 0/6/cpu0
Mon Jun 29 00:41:43.450 PST
NODE module 0/6/CPU0 :
 MAIN: board type 0x20208
  S/N: FHH12250033
  Top Assy. Number: 68-3184-02
  PID: A9K-4T-B
  UDI VID: V1D
  HwRev: V0.0
 New Deviation Number: 0
  CLEI:
  Board State : IOS XR RUN
  PLD:
         Motherboard: N/A, Processor: 0x8004 (rev: 2.2), Power: N/A
  ROMMON: Version 1.0(20081208:174521) [ASR9K ROMMON]
  Board FPGA/CPLD/ASIC Hardware Revision:
    NPO : V3.194
```

```
NP1 : V3.194
NP2 : V3.194
NP3 : V3.194
XbarInterface : V18.4
Bridge0 : V0.38
Bridgel : V0.38
CPUCtrl : V0.15
USB : V77.20
PHY0 : V0.16
PHY1 : V0.16
     : V0.16
PHY2
PHY3 : V0.16
PortCtrl : V0.10
PHYCtrl : V0.7
4 Port Ten Gigabit Ethernet Daughter board : V0.0
CBC (active partition) : v2.2
CBC (inactive partition) : v2.1
```

Displaying System Hardware Version Information

To display hardware version information for all or some of the components assigned in a system, connect to the designated shelf controller (DSC) and enter the **show diag** command in administration EXEC mode. When this command is entered in administration EXEC mode, you can display information on RSPs, line cards, and system components such as the chassis, fan trays, and power supplies.



Note If you enter the **show diag** command in EXEC mode, the software displays only the hardware assigned to the SDR to which you are connected.

The syntax for the **show diag** command in administration EXEC mode is:

show diag [node-id | chassis | details | fans | memory | power-supply | summary]



Tip For information on the software version, use the show version command.

In the following example, the **show diag** command displays information for all nodes in the system:

```
RP/0/RSP0/CPU0:router(admin) # show diag
Mon Jun 29 01:21:04.571 PST
NODE module 0/RSP0/CPU0 :
  MAIN: board type 0x100302
  S/N:
        FOC1230803H
  Top Assy. Number:
                      68-3160-04
  PID: A2K-RSP-4G-HDD=
  UDI VID:
            VP4
  HwRev: V4.8
  New Deviation Number: 0
  CLEI: IPUCARJBAA
  Board State : IOS XR RUN
        Motherboard: N/A, Processor: 0x8004 (rev: 2.2), Power: N/A
  PLD:
```

```
MONLIB: QNXFFS Monlib Version 3.2
  ROMMON: Version 1.0 (20081208:173612) [ASR9K ROMMON]
  Board FPGA/CPLD/ASIC Hardware Revision:
    Compact Flash : V1.0
    XbarSwitch0 : V1.3
    XbarSwitch1 : V1.3
    XbarArbiter : V1.0
    XbarInterface : V0.0
    IntCtrl : V1.14
    ClkCtrl : V1.13
    PuntFPGA : V1.5
    HD : V3.0
    USB0 : V77.20
    USB1 : V77.20
    CPUCtrl : V1.17
    UTI : V1.6
    LIU
         : V1.0
    MLANSwitch : V0.0
    EOBCSwitch : V2.0
    CBC (active partition) : v1.2
    CBC (inactive partition) : v1.1
NODE fantray 0/FT0/SP :
  MAIN: board type 0x900211
  S/N:
  Top Assy. Number: 32-0000-00
  PID:
  UDI VID:
 HwRev: V32.0
 New Deviation Number: 0
  CLEI:
  PLD:
         Motherboard: N/A, Processor: N/A, Power: N/A
  ROMMON:
  Board FPGA/CPLD/ASIC Hardware Revision:
    CBC (active partition) : v4.0
    CBC (inactive partition) : v0.13
NODE fantray 0/FT1/SP :
 MAIN: board type 0x900211
  S/N:
 Top Assy. Number: 32-0000-00
  PID:
  UDI VID:
 HwRev: V32.0
 New Deviation Number: 0
  CLEI:
 PTD:
         Motherboard: N/A, Processor: N/A, Power: N/A
  ROMMON:
  Board FPGA/CPLD/ASIC Hardware Revision:
    CBC (active partition) : v4.0
    CBC (inactive partition) : v0.13
NODE module 0/1/CPU0 :
  MAIN: board type 0x20207
  S/N: FOC123081J6
  Top Assy. Number:
                     68-3182-03
  PID: A9K-40GE-B
  UDI VID: V1D
 HwRev: V0.0
 New Deviation Number: 0
  CLEI:
```

```
Board State : IOS XR RUN
  PLD: Motherboard: N/A, Processor: 0x8004 (rev: 2.2), Power: N/A
  ROMMON: Version 1.0(20081208:174521) [ASR9K ROMMON]
  Board FPGA/CPLD/ASIC Hardware Revision:
    NPO : V3.194
         : V3.194
    NP1
    NP2 : V3.194
    NP3 : V3.194
    XbarInterface : V18.4
    Bridge0 : V0.38
    Bridgel : V0.38
CPUCtrl : V0.15
    USB : V77.20
     PortCtrl : V0.8
    PHYCtrl : V0.6
     40 Port Gigabit Ethernet Daughter board : V0.0
     CBC (active partition) : v2.2
     CBC (inactive partition) : v2.1
NODE module 0/4/CPU0 :
  MAIN: board type 0x2020a
  S/N:
        FOC123081JA
  Top Assy. Number:
                     68-3183-02
  PID: A9K-8T/4-B
  UDI VID: V1D
  HwRev: V0.0
  New Deviation Number: 0
  CLEI: IPU3AE0CAA
  Board State : IOS XR RUN
  PLD: Motherboard: N/A, Processor: 0x8004 (rev: 2.2), Power: N/A
  ROMMON: Version 1.0(20081208:174521) [ASR9K ROMMON]
  Board FPGA/CPLD/ASIC Hardware Revision:
    NPO : V3.194
    NP1 : V3.194
    NP2 : V3.194
    NP3 : V3.194
    XbarInterface : V18.4
    Bridge0 : V0.38
    Bridgel : V0.38
    CPUCtrl : V0.15
    USB : V77.20
    PortCtrl : V0.10
     PHYCtrl : V0.7
    PHY0 : V0.16
    PHY1 : V0.16
     PHY2 : V0.16
    PHY3 : V0.16
     PHY4 : V0.16
     PHY5
          : V0.16
    PHY6 : V0.16
    PHY7 : V0.16
     8 Port Ten Gigabit Ethernet Daughter board : V0.0
    CBC (active partition) : v2.2
    CBC (inactive partition) : v2.1
NODE module 0/6/\mbox{CPU0} :
  MAIN: board type 0x20208
  S/N:
       FHH12250033
  Top Assy. Number:
                     68-3184-02
  PID: A9K-4T-B
  UDI VID: V1D
  HwRev: V0.0
```

```
New Deviation Number: 0
  CLET:
  Board State : IOS XR RUN
  PLD: Motherboard: N/A, Processor: 0x8004 (rev: 2.2), Power: N/A
  ROMMON: Version 1.0(20081208:174521) [ASR9K ROMMON]
  Board FPGA/CPLD/ASIC Hardware Revision:
    NPO : V3.194
    NP1 : V3.194
    NP2 : V3.194
    NP3 : V3.194
    XbarInterface : V18.4
    Bridge0 : V0.38
    Bridgel : V0.38
    CPUCtrl : V0.15
    USB : V77.20
    PHY0 : V0.16
     PHY1
          : V0.16
    PHY2 : V0.16
    PHY3 : V0.16
    PortCtrl : V0.10
    PHYCtrl : V0.7
     4 Port Ten Gigabit Ethernet Daughter board : V0.0
    CBC (active partition) : v2.2
    CBC (inactive partition) : v2.1
NODE power-module 0/PM0/SP :
  MAIN: board type 0xf00188
  S/N:
  Top Assy. Number:
                    341-00032-01
  PID: A9K-3KW-AC
  UDI VID: V00
  HwRev: V0.0
  New Deviation Number: 0
 CLEI: ACACACACAC
  PLD: Motherboard: N/A, Processor: N/A, Power: N/A
  ROMMON:
  Board FPGA/CPLD/ASIC Hardware Revision:
NODE power-module 0/PM1/SP :
  MAIN: board type 0xf00188
 S/N:
  Top Assy. Number:
                     341-00032-01
  PID: A9K-3KW-AC
 UDI VID: V00
 HwRev: V0.0
  New Deviation Number: 0
  CLEI: ACACACACAC
         Motherboard: N/A, Processor: N/A, Power: N/A
  PLD:
  ROMMON:
  Board FPGA/CPLD/ASIC Hardware Revision:
NODE power-module 0/PM2/SP :
  MAIN: board type 0xf00188
  S/N:
  Top Assy. Number:
                     341-00032-01
  PID: A9K-3KW-AC
  UDI VID: V00
  HwRev: V0.0
  New Deviation Number: 0
  CLEI: ACACACACAC
       Motherboard: N/A, Processor: N/A, Power: N/A
  PLD:
```

```
ROMMON:
Board FPGA/CPLD/ASIC Hardware Revision:
Rack 0 - ASR-9010 Chassis, Includes Accessories
RACK NUM: 0
S/N:
PID: ASR-9010 Backplane
VID: 0.1
Desc: ASR-9010 Chassis, Includes Accessories
CLEI: NOCLEI
Top Assy. Number: 68-1234-56
```

In the following example, the **show diag** command displays information for a single system component:

```
RP/0/RSP0/CPU0:router(admin)# show diag chassis
Mon Jun 29 01:25:05.711 PST
Rack 0 - ASR-9010 Chassis, Includes Accessories
RACK NUM: 0
S/N:
PID: ASR-9010 Backplane
VID: 0.1
Desc: ASR-9010 Chassis, Includes Accessories
CLEI: NOCLEI
Top Assy. Number: 68-1234-56
```

Displaying Software and Hardware Information

The **show version** command displays a variety of system information, including the hardware and software versions, router uptime, boot settings (including the configuration register), and active software.

The following is sample output from the show version command:

For Cisco ASR 9000 64 bit environment:

```
RP/0/RSP0/CPU0:Router#show version
Cisco IOS XR Software, Version 7.5.2
Copyright (c) 2013-2022 by Cisco Systems, Inc.
Build Information:
Built By : ingunawa
Built On
             : Tue Apr 26 18:26:36 PDT 2022
 Built Host : iox-ucs-055
             : /auto/srcarchive14/prod/7.5.2/asr9k-x64/ws
Workspace
            : 7.5.2
Version
Location : /opt/cisco/XR/packages/
Label
            : 7.5.2
cisco ASR9K () processor
System uptime is 4 weeks 2 hours 31 minutes
```

For Cisco ASR9000 32 bit environment:

```
RP/0/RSP0/CPU0:Router#show version
Sat Aug 1 22:52:39.089 DST
Cisco IOS XR Software, Version 3.9.0.16I[DT_IMAGE]
Copyright (c) 2009 by Cisco Systems, Inc.
```

```
ROM: System Bootstrap, Version 1.1(20090521:183759) [ASR9K ROMMON],
router uptime is 1 day, 2 hours, 34 minutes
System image file is "bootflash:disk0/asr9k-os-mbi-3.9.0.16I/mbiasr9k-rp.vm"
cisco ASR9K Series (MPC8641D) processor with 4194304K bytes of memory.
MPC8641D processor at 1333MHz, Revision 2.2
2 Management Ethernet
12 TenGigE
40 GigabitEthernet
219k bytes of non-volatile configuration memory.
975M bytes of compact flash card.
33994M bytes of hard disk.
1605616k bytes of disk0: (Sector size 512 bytes).
1605616k bytes of disk1: (Sector size 512 bytes).
Configuration register on node 0/RSP0/CPU0 is 0x102
Boot device on node 0/RSP0/CPU0 is disk0:
Package active on node 0/RSP0/CPU0:
asr9k-scfclient, V 3.9.0.16I[DT IMAGE], Cisco Systems, at disk0:asr9k-scfclient-3.9.0.16I
   Built on Thu Jul 30 12:09:40 DST 2009
   By sjc-lds-208 in /auto/ioxbuild7/production/3.9.0.16I.DT IMAGE/asr9k/workspace for
c4.2.1-p0
asr9k-adv-video, V 3.9.0.16I[DT_IMAGE], Cisco Systems, at disk0:asr9k-adv-video-3.9.0.16I
   Built on Thu Jul 30 13:49:37 DST 2009
    By sjc-lds-208 in /auto/ioxbuild7/production/3.9.0.16I.DT IMAGE/asr9k/workspace for
c4.2.1-p0
asr9k-fpd, V 3.9.0.16I[DT IMAGE], Cisco Systems, at disk0:asr9k-fpd-3.9.0.16I
   Built on Thu Jul 30 12:26:21 DST 2009
   By sjc-lds-208 in /auto/ioxbuild7/production/3.9.0.16I.DT IMAGE/asr9k/workspace for
c4.2.1-p0
asr9k-diags, V 3.9.0.16I[DT IMAGE], Cisco Systems, at disk0:asr9k-diags-3.9.0.16I
   Built on Thu Jul 30 12:09:43 DST 2009
   By sjc-lds-208 in /auto/ioxbuild7/production/3.9.0.16I.DT IMAGE/asr9k/workspace for
c4.2.1-p0
asr9k-k9sec, V 3.9.0.16I[DT IMAGE], Cisco Systems, at disk0:asr9k-k9sec-3.9.0.16I
    Built on Thu Jul 30 12:25:25 DST 2009
   By sjc-lds-208 in /auto/ioxbuild7/production/3.9.0.16I.DT IMAGE/asr9k/workspace for
c4.2.1-p0
asr9k-mgbl, V 3.9.0.16I[DT IMAGE], Cisco Systems, at disk0:asr9k-mgbl-3.9.0.16I
   Built on Thu Jul 30 13:48:16 DST 2009
 --More--
```

Displaying SDR Node IDs and Status

In EXEC mode, the **show platform** command displays information for all nodes assigned to the owner SDR. For each node, this information includes the host card type, the operational state, and the configuration state. To display information on a single node, enter the command with a node ID.

The syntax for the **show platform** command is:

show platform [node-id]

The following example displays the status for all nodes in the SDR to which you are connected:

RP/0/RSP0/CPU0:router# show platform Mon Aug 3 07:39:01.416 DST						
Node	Туре	State	Config State			
0/RSP0/CPU0	A9K-RSP-4G(Active)	IOS XR RUN	PWR,NSHUT,MON			
0/1/CPU0	A9K-40GE-B	IOS XR RUN	PWR,NSHUT,MON			
0/4/CPU0	A9K-8T/4-B	IOS XR RUN	PWR,NSHUT,MON			
0/6/CPU0	А9К-4Т-В	IOS XR RUN	PWR, NSHUT, MON			

The node-id appears in the rack/slot/module notation, and the node-id components are as follows:

- rack —In a single-shelf system the rack number is always "0."
- slot —Number of the physical slot in which the card is installed.
- module ---Subslot number of a system hardware component.

Table 30: Node ID Components, on page 441 summarizes the node-id for each type of card.

Table 30: Node ID Components

Card Type (the card to which your are issuing commands)	Rack (always "0")	Slot (the physical slot in which the card is installed)	Module (the entity on the card that is the target of the command)
Route switch processor	0	RSP0 and RSP1	CPU0
40-Port Gigabit Ethernet Line Card8-Port 10-Gigabit Ethernet Line Card4-Port 10-Gigabit Ethernet Line Card	0-255	4-7 (6-slot chassis) 0–7 (10-slot chassis)	0-X (SFP and XFP module number on the line card)
Power Modules Fan controller cards	0 0 0	PM0-PM5 (10-slot chassis) PM0-PM2 (6-slot chassis FC0-FC1	

Displaying Router Node IDs and Status

In administration EXEC mode, the **show platform** command displays information for all router nodes. In administration EXEC mode, the command display also includes additional node IDs such as those for fabric cards, alarm modules, and fan controllers. For each node, this information includes the host card type, the operational state, and the configuration state. To display information on a single node, enter the command with a node ID.

The syntax for the show platform command is:

show platform [node-id]

The following example displays the status for all nodes in the system:

RP/0/RSP0/CPU0:router(admin)# show platform

Sat Mar 24 05:02 Node	2:18.569 DST Type	State	Config State
0/RSP0/CPU0	A9K-RSP-4G(Active)	IOS XR RUN	PWR, NSHUT, MON
0/1/CPU0	A9K-40GE-B	IOS XR RUN	PWR, NSHUT, MON
0/4/CPU0	A9K-8T/4-B	IOS XR RUN	PWR, NSHUT, MON

0/6/CPU0 A9K-4T-B

IOS XR RUN

PWR,NSHUT,MON

The *node-id* appears in the *rack/slot/module* notation, and the *node-id* components are as follows:

- rack In a single-shelf system the rack number is always "0."
- slot —Number of the physical slot in which the card is installed.
- module ---Subslot number of a system hardware component.

Table 30: Node ID Components, on page 441 summarizes the node-id argument for each type of card.

Displaying Router Environment Information

The **show environment** command displays hardware information for the system, including fan speeds, LED indications, power supply voltage and current information, and temperatures.

The syntax for the show environment command is:

show environment [options]

You can use the **show environment** command options to limit the detail in the command display. To view the command options, enter the **show environment** ? command. The following example shows the full environment status report:

RP/0/RSP0/CPU0:router(admin) # show environment

Mon Jun 29 04:32:07.587 PST

Temperature Information

R/S/I	Modules	Inlet Temperature (deg C)	Hotspot Temperature (deg C)
0/1/*	host	31 5	30 5
0/RSP0/	*	51.5	55.5
0 / 4 / +	host	26.6	36.6
0/4/*	host	29.8	38.8
0/6/*	host	32.7	42.0
0/FT0/*	host	27.2	28.2
0/FT1/*	host	27.4	30.2
Voltage	Information		
R/S/I	Modules Sensor		(mV) Margin

I

0/1/*	
	ho

	host	IBV	10647	n/a
	host	5.0V	4929	n/a
	host	VP3P3 CAN	3288	n/a
	heat	2 217	2200	n/a
	nost	3.30	550I	11/a
	nost	2.5V	2516	n/a
	host	1.8VB	1810	n/a
	host	1.2VB	1193	n/a
	host	1.8VA	1800	n/a
	host	0.9VB	884	n/a
	host	1.2V LDO BRG0	1193	n/a
	host	1 2V LDO BRG1	1195	n/a
	host		1011	n/a
	liost	1.8VC	1011	11/a
	nost	1.5VB	1505	n/a
	host	1.5VA	1503	n/a
	host	1.1V(1.05V_CPU)	1052	n/a
	host	0.75VA	751	n/a
	host	0.75VB 0.75VC	754	n/a
	host	1.1VB	1102	n/a
	host	1.2V TCAM0	1003	n/a
	host	1 2V TCAM1	1000	n/a
	hest	1 OV Dridge IDO	1000	n/a
	nost	1.0V_BIIdge_LDO	990	II/d
	nost	I.UVB	1043	n/a
	host	0.75VD_and_0.75VE	752	n/a
	host	1.2V_TCAM2	1005	n/a
	host	1.2V_TCAM3	1002	n/a
	host	1.5VC	1504	n/a
	host	1.8VD	1803	n/a
	host	1.1VC	1099	n/a
	host	ZARLINK 3 3V	3272	n/a
	hest	ZADI INK 1 017	1000	n/a
	nost	LARLINK_1.0V	11000	II/d
	host	1.2V_DB	1195	n/a
	host	3.3V_DB	3316	n/a
	host	2.5V_DB	2534	n/a
	host	1.5V_DB	1509	n/a
0/RSP0/	*			
	host	0.75VTT	749	n/a
	host	0.9VTT A	910	n/a
	host	0.9VTT B	904	n/a
	host	TBV	10586	n/a
	hest	E OV	E012	n/a
	hest	J.UV	2013	11/a
	nost	VP3P3_CAN	3277	n/a
	host	3.3V	3299	n/a
	host	2.5V	2518	n/a
	host	1.8VB	1807	n/a
	host	1.2VA	1205	n/a
	host	1.2VB	1202	n/a
	host	1.05V	1047	n/a
	host	1.2VD	1205	n/a
	host	1 877	1811	n/a
	hest	1 EV	1400	11/a
	nost	1.5V	1496	n/a
	host	1.90	1887	n/a
0/4/*				
	host	IBV	10627	n/a
	host	5.0V	4917	n/a
	host	VP3P3 CAN	3279	n/a
	host	3.3V -	3296	n/a
	host.	2.5V	2522	n/a
	host	1 8VB	1805	n/a
	host	1 2VB	1100	n/~
	host	1 0177	1700	11/d
	nost	L.OVA	T / 96	11/a
	nost	U.9VB	88T	n/a

0/6/*

I

host	1.2V LDO BRG0	1192	n/a
host	1 2V IDO PPC1	1105	n/-
1030	1.0VG	1000	11/a
nost	1.800	1806	n/a
host	1.5VB	1510	n/a
host	1.5VA	1503	n/a
host	1.1V(1.05V CPU)	1048	n/a
host	0 7577	753	n/2
1030	0.75VA	755	11/a
nost	0./5VB_0./5VC	/5/	n/a
host	1.1VB	1105	n/a
host	1.2V TCAM0	1003	n/a
host	1.2V TCAM1	1000	n/a
host	1 OV Bridge LDO	997	n/a
heet	1.0V_DI10ge_DD0	1027	11/u
nost	1.000	1037	II/a
host	0.75VD_and_0.75VE	755	n/a
host	1.2V_TCAM2	1004	n/a
host	1.2V TCAM3	1005	n/a
host	1.5VC	1505	n/a
host	1 800	1808	n/a
1030	1 110	1104	11/a
nost	1.100	1104	n/a
host	ZARLINK_3.3V	3285	n/a
host	ZARLINK 1.8V	1806	n/a
host	1.2V DB	1205	n/a
host	3 3V DB	3318	n/a
heet		2402	
nost	2.5V_DB	2493	n/a
host	1.5V_DB	1497	n/a
host	1.8V_DB	1825	n/a
host	5.0V XFP DB	5001	n/a
host	1.2VB DB	1228	n/a
	· _		
host	IBV	10628	n/a
host	5.0V	4893	n/a
host	VD3D3 CAN	3281	n/a
1030	0 377	3201	11/a
nost	3.30	3297	n/a
host	2.5V	2524	n/a
host	1.8VB	1804	n/a
host	1.2VB	1204	n/a
host	1.8VA	1795	n/a
host	0 0778	881	n/a
hest		1104	11/a
nost	1.2V_LDO_BRGU	1194	n/a
host	1.2V_LDO_BRG1	1193	n/a
host	1.8VC	1815	n/a
host	1.5VB	1495	n/a
host	1.5VA	1503	n/a
host	1 1V(1 05V CPII)	1052	n/a
1030	0.7517	1052	11/a
nost	0.75VA	152	n/a
host	0.75VB_0.75VC	749	n/a
host	1.1VB	1001	n/a
host	1.2V TCAM0	999	n/a
host	1.2V TCAM1	1002	n/a
hogt	1 OV Pridro IDO	005	n/a
nost	1.0V_BIIdge_LDO	1050	11/a
host	1.0VB	1050	n/a
host	0.75VD_and_0.75VE	752	n/a
host	1.2V TCAM2	1002	n/a
host	1.2V TCAM3	995	n/a
host			.,
	1.5VC	1502	n/a
host	1.5VC	1502	n/a
host	1.5VC 1.8VD	1502 1802	n/a n/a
host host	1.5VC 1.8VD 1.1VC	1502 1802 1101	n/a n/a n/a
host host host	1.5VC 1.8VD 1.1VC ZARLINK_3.3V	1502 1802 1101 3273	n/a n/a n/a n/a
host host host host	1.5VC 1.8VD 1.1VC ZARLINK_3.3V ZARLINK 1.8V	1502 1802 1101 3273 1804	n/a n/a n/a n/a n/a
host host host host host	1.5VC 1.8VD 1.1VC ZARLINK_3.3V ZARLINK_1.8V 1.2V DB	1502 1802 1101 3273 1804 1200	n/a n/a n/a n/a n/a
host host host host host	1.5VC 1.8VD 1.1VC ZARLINK_3.3V ZARLINK_1.8V 1.2V_DB 3.3V_DB	1502 1802 1101 3273 1804 1200 3314	n/a n/a n/a n/a n/a n/a
host host host host host host	1.5VC 1.8VD 1.1VC ZARLINK_3.3V ZARLINK_1.8V 1.2V_DB 3.3V_DB 2.5V_DB	1502 1802 1101 3273 1804 1200 3314 2496	n/a n/a n/a n/a n/a n/a
host host host host host host	1.5VC 1.8VD 1.1VC ZARLINK_3.3V ZARLINK_1.8V 1.2V_DB 3.3V_DB 2.5V_DB	1502 1802 1101 3273 1804 1200 3314 2496	n/a n/a n/a n/a n/a n/a

1824 n/a 5004 n/a host 1.8V_DB host 5.0V_XFP_DB host 1.2VB DB 1227 n/a LED Information _____ R/S/I Modules LED Status 0/RSP0/* Critical-Alarm Off Major-Alarm Off Minor-Alarm Off host host host host ACO Off Fan Information -----Fan speed (rpm): FANO FAN1 FAN2 FAN3 FAN4 FAN5 FAN6 FAN7 FAN8 FAN9 FAN10 FAN11 0/FT0/* 3510 3510 3510 3540 3510 3570 3480 3570 3510 3510 3510 3510 0/FT1/* 3540 3510 3450 3540 3480 3600 3480 3450 3540 3540 3480 3540 Power Supply Information _____ R/S/I Modules Sensor Watts 0/PM0/* host PM 3000 0/PM1/* host PM 3000 0/PM2/* 3000 host PM Power Shelves Type: AC Total Power Capacity: 9000W Protected Power Capacity: 4500W 3145W Worst Case Power Used: Slot Max Watts _____ 0/1/CPU0 375 0/RSP0/CPU0 250 0/RSP1/CPU0 350 0/4/CPU0 375 0/6/CPU0 375 710 (default) 0/FT0/SP 710 (default) 0/FT1/SP Worst Case Protected Power Available: 1355W

Displaying RP Redundancy Status

The **show redundancy** command displays the redundancy status of the route switch processors (RSPs). This command also displays the boot and switch-over history for the RSPs.

The show redundancy operates in EXEC and administration EXEC mode.

In the following example, the **show redundancy** command displays the redundancy status for a redundant RSP pair:

RP/0/RSP0/CPU0:router(admin) # show redundancy

Active node reload "Cause: Turboboot completed successfully"

Displaying Field-Programmable Device Compatibility

The **show hw-module fpd** command displays field-programmable device (FPD) compatibility for all modules or a specific module.

The syntax for the **show hw-module fpd** command is:

show hw-module fpd location {all | node-id}

The show hw-module fpd operates in EXEC and administration EXEC mode.

The following example shows how to display FPD compatibility for all modules in the router:

```
RP/0/RSP0/CPU0:router# ios#show hw-module fpd
Tue Jan 22 13:56:55.082 UTC
```

						FPD Vers	lons
Location	Card type	HWver	FPD device	ATR	Status	Running	Programd
0/RP0	NCS-55A2-MOD-S	0.3	MB-MIFPGA		CURRENT	0.19	0.19
0/RP0	NCS-55A2-MOD-S	0.3	Bootloader		CURRENT	1.10	1.10
0/RP0	NCS-55A2-MOD-S	0.3	CPU-IOFPGA		CURRENT	1.18	1.18
0/RP0	NCS-55A2-MOD-S	0.3	MB-IOFPGA		CURRENT	0.18	0.18
0/PM0	NC55-1200W-ACFW	1.0	LIT-PriMCU-ACFW		NEED UPG	D 2.08	2.08
0/PM1	NC55-1200W-ACFW	1.0	LIT-PriMCU-ACFW		NEED UPG	D 2.08	2.08
RP/0/RP0/0	CPU0:ios#.						



Note After Release 5.3.x, Upg/Dng? will display Yes only for upgrade.

The following example shows the FPD for which upgrage will be skipped.
RP/0/RSP1/CPU0:router# show hw-module fpd location all

Mon Jun 29 05:38:50.332 PST

		Existing Field Programmable Devices					
Location	Card Type	HW HW Version	Туре	Subtype	Inst	Current SW Version	Upg/ Dng?
0/RSP0/CPU0	A9K-RSP-4G	4.8	lc lc lc lc lc lc lc	fpga3 fpga1 fpga2 cbc fpga4 rommon	0 0 0 0 0 0 0	1.13 1.5 1.14 1.2 1.6 1.0	No No No No No No
0/RSP0/CPU0	ASR-9010-FAN	1.0	lc	cbc	1	4.0	No
0/RSP0/CPU0	ASR-9010-FAN	1.0	lc	cbc	2	4.0	No
0/1/CPU0	A9K-40GE-B	1.0	lc lc lc lc lc lc	fpgal fpga2 cbc cpld1 rommon	0 0 0 0 0	0.38 0.8 2.2 0.15 1.0	No No No No No
0/1/CPU0	A9K-40GE-B	1.0	lc	fpgal	1	0.38	No
0/4/CPU0	А9К-8Т/4-В	1.0	1c 1c 1c 1c 1c 1c 1c 1c	fpgal fpga2 cbc cpld2 cpld1 cpld3 rommon fpga3	0 0 0 0 0 0 0 0	0.38 0.10 2.2 0.7 0.15 0.3 1.0 14.42	No No No No No No No
0/4/CPU0	A9K-8T/4-B	1.0	lc	fpga1	1	0.38	No
0/6/CPU0	А9К-4Т-В	1.0	lc lc lc lc lc lc lc lc lc lc lc	fpgal fpga2 cbc cpld2 cpld1 cpld3 rommon fpga3	0 0 0 0 0 0 0 0 0 0	0.38 0.10 2.2 0.7 0.15 0.3 1.0 14.42	No No No No No No No
0/6/CPU0	А9К-4Т-В	1.0	lc	fpgal	1	0.38	No

The following example shows how to display FPD compatibility for a specific module in the router:

Table 31: show hw-module fpd Field Descriptions

Field	Description
Location	Location of the module in the <i>rack/slot/module</i> notation.
Card Type	Module part number.

Field	Description
HW Version	Hardware model version for the module.
Туре	Hardware type. Can be one of the following types:
	• spa—Shared port adapter
	• lc—Line card
Subtype	FPD type. Can be one of the following types:
	• fabldr—Fabric downloader
	• fpga1—Field-programmable gate array
	• fpga2—Field-programmable gate array 2
	• fpga3—Field-programmable gate array 3
	• fpga4—Field-programmable gate array 4
	• fpga5—Field-programmable gate array 5
	rommonA—Read-only memory monitor A
	rommon—Read-only memory monitor B
Inst	FPD instance. The FPD instance uniquely identifies an FPD and is used by the FPD process to register an FPD.
Current SW Version	Currently running FPD image version.
Upg/Dng?	Specifies whether an FPD upgrade or downgrade is required. A downgrade is required in rare cases when the version of the FPD image has a higher major revision than the version of the FPD image in the current Cisco IOS XR software package.

RSP Redundancy and Switchover

This section describes RSP redundancy and switchover commands and issues.

Establishing RSP Redundancy

Your router has two slots for RSPs: RSP0 and RSP1 (see Figure 35: Redundant Set of RSPs Installed in Slots RSP0 and RSP1 in an 8-Slot Chassis, on page 449). RSP0 is the slot on the left, facing the front of the chassis, and RSP1 is the slot on right. These slots are configured for redundancy by default, and the redundancy cannot be eliminated. To establish RSP redundancy, install RSPs into both slots.



Figure 35: Redundant Set of RSPs Installed in Slots RSP0 and RSP1 in an 8-Slot Chassis

Determining the Active RP in a Redundant Pair

During system startup, one RSP in each redundant pair becomes the active RSP. You can tell which RSP is the active RSP in the following ways:

- The active RSP can be identified by the green Primary LED on the faceplate of the card. The active RSP is indicated when the Primary LED is on. The alphanumeric LED display on the RSP displays ACTV RP.
- The slot of the active RSP is indicated in the CLI prompt. For example:

RP/0/RSP1/CPU0:router#

In this example, the prompt indicates that you are communicating with the active RSP in slot RSP1. See *Cisco ASR 9000 Series Aggregation Services Router Getting Started Guide* for a complete description of the CLI prompt.

• Enter the **show redundancy** command in EXEC mode to display a summary of the active and standby RSP status. For example:

RP/0/RSP0/CPU0:router(admin)# show redundancy

```
Mon Jun 29 04:49:26.098 PST
Redundancy information for node 0/RSP0/CPU0:
```

Role of the Standby RSP

The second RSP to boot in a redundant pair automatically becomes the standby RSP. While the active RSP manages the system and communicates with the user interface, the standby RSP maintains a complete backup of the software and configurations for all cards in the system. If the active RSP fails or goes off line for any reason, the standby RSP immediately takes control of the system.

Summary of Redundancy Commands

RSP redundancy is enabled by default in the Cisco IOS XR software, but you can use the commands described in Table 32: RSP Redundancy Commands, on page 450 to display the redundancy status of the cards or force a manual switchover.

Command	Description
show redundancy	Displays the redundancy status of the RSPs. This command also displays the boot and switch-over history for the RSPs.
redundancy switchover	Forces a manual switchover to the standby RSP. This command works only if the standby RSP is installed and in the "ready" state.
show platform	Displays the status for node, including the redundancy status of the RSP 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.

Table 32: RSP Redundancy Commands

Automatic Switchover

Automatic switchover from the active RSP to the standby RSP occurs only if the active RSP encounters a serious system error, such as the loss of a mandatory process or a hardware failure. When an automatic switchover occurs, the RSPs respond as follows:

- If a standby RSP is installed and "ready" for switchover, the standby RSP becomes the active RSP. The original active RSP attempts to reboot.
- If the standby RSP is not in "ready" state, then both RSPs reboot. The first RSP to boot successfully assumes the role of active RSP.

RSP Redundancy During RSP Reload

The **reload** command causes the active RSP to reload the Cisco IOS XR software. When an RSP reload occurs, the RSPs respond as follows:

- If a standby RSP is installed and "ready" for switchover, the standby RSP becomes the active RSP. The original active RSP reboots and becomes the standby RSP.
- If the standby RSP is not in the "ready" state, then both RSPs reboot. The first RSP to boot successfully assumes the role of active RSP.

<u>/!\</u>

Caution

You should not use the **reload** command to force an RSP switchover because the result could be a significant loss of router operations. Instead, use the **redundancy switchover** command to fail over to the standby RSP, then use the **hw-module location** *node-id* **reload** command to reload the new standby RSP.

Related Topics

Reloading, Shutting Down, or Power Cycling a Node, on page 457

Manual Switchover

You can force a manual switchover from the active RSP to the standby RSP using the **redundancy switchover** command.

If a standby RSP is installed and ready for switchover, the standby RSP becomes the active RSP. The original active RSP becomes the standby RSP. In the following example, partial output for a successful redundancy switchover operation is shown:

RP/0/RSP0/CPU0:router# show redundancy

```
This node (0/RSP0/CPU0) is in ACTIVE role
Partner node (0/RSP1/CPU0) is in STANDBY role
Standby node in 0/RSP1/CPU0 is ready
RP/0/RSP0/CPU0:router# redundancy switchover
Updating Commit Database. Please wait...[OK]
Proceed with switchover 0/RSP0/CPU0 -> 0/RSP1/CPU0? [confirm]
Initiating switch-over.
RP/0/RSP0/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/RSP1/CPU0:router#

If the standby RSP 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/RSP0/CPU0:router# show redundancy

Switchover disallowed: Standby node is not ready.

System Logs during RSP Switchover

Table 33: Feature History Table

Feature Name	Release Information	Feature Description
RSP Slot Location in Syslog	Release 7.8.1	 When an RSP switchover occurs, the router logs the active RSP slot location in the syslog message. This helps you quickly identify the active RSP slot from your router's system log messages. In earlier releases, the RSP switchover syslog message didn't include the active RSP slot location.

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

standby, location RP/1/1/CPU0

From Cisco IOS XR Release 7.8.1 onwards, the RSP switchover syslog message for the active RSP includes the RSP slot location as well:

RP/0/1/CPU0:Mar 8 11:42:50.876 UTC: rmf_svr[165]: %HA-REDCON-6-GO_ACTIVE : this card going active , location RP/0/1/CPU0:

Communicating with a Standby RP

The active RSP automatically synchronizes all system software, settings, and configurations with the standby RSP.

If you connect to the standby RSP through the console port, you can view the status messages for the standby RSP. The standby RSP 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 RSP through the management Ethernet port, the prompt that appears is for the active RSP, and you can manage the router the same as if you had connected through the management Ethernet port on the active RSP.

Configuring Breakout on a 400GE Port

Feature Name	Release Information	Description
Breakout in a 400GE Port	Release 7.4.1	You can configure breakout on a 400GE port for use the QSFP-28 transceiver module in a 1x100G mode. This feature is supported on Cisco ASR 9900 5th Generation 4Tb line cards (A99-10X400GE-X-SE/TR).
Configuring 4x100GE Breakout for ASR 9000 2T Combo Line Card (A9K-20HG-FLEX) and 800G Combo Line Card (A9K-8HG-FLEX)	Release 7.4.1	This release introduces support for 4x100 breakout on the ASR 9000 2T Combo Line Card (A9K-20HG-FLEX) and 800G Combo Line Card (A9K-8HG-FLEX). You can configure 4x100G breakout on a QSFP-DD port and use it as four individual 100G interfaces.

Table 34: Feature History Table

1x100GE Breakout

You can configure breakout on a 400GE port for use the QSFP-28 transceiver module in a 1x100G mode. This feature is supported on Cisco ASR 9900 5th Generation 4Tb line cards (A99-10X400GE-X-SE/TR). As part of this enhancement, 400GE ports support these QSFP28 optics:

- QSFP-100G-SR4-S
- QSFP-100G-PSM4-S
- QSFP-100G-CWDM4-S
- QSFP-100G-FR-S
- QSFP-100G-LR4-S
- QSFP-100G-ER4L-S
- QSFP-40/100-SRBD
- QSFP-100G-SM-SR

• QSFP-100G-LR-S

This task enables you to configure the breakout option:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# hw-module location 0/2/CPU0 port 1 breakout 1xHundredGige
```

Here's the **show** command output indicating the breakout configuration:

RP/0/RSP0/CPU0:router# show	ipv4 interface br	ief includ	le Hun	
Tue May 4 07:59:56.996 UTC				
FourHundredGigE0/1/0/2	unassigned	Down	Down	default
HundredGigE0/2/0/0	198.127.6.1	Up	Up	vpn6
HundredGigE0/2/0/1	198.127.12.1	Up	Up	vpn12
HundredGigE0/2/0/2	198.127.10.1	Up	Up	vpn10
HundredGigE0/2/0/3	198.127.11.1	Up	Up	vpn11
HundredGigE0/2/0/4	198.127.9.1	Up	Up	vpn9
HundredGigE0/2/0/5	198.127.5.1	Up	Up	vpn5
HundredGigE0/2/0/6	198.127.1.1	Up	Up	vpn1
HundredGigE0/2/0/7	198.127.7.1	Up	Up	vpn7
HundredGigE0/2/0/8	198.127.4.1	Up	Up	vpn4

Here's the **show** command output indicating the 100G optics plugged in the QSFP-DD port number 2:

RP/0/RSP0/CPU0:router#show inventory loc 0/2 Tue May 4 07:58:58.461 UTC NAME: "0/2", DESCR: "ASR 9900 10-port 400GE SE linecard" PID: A99-10X400GE-X-SE , VID: V00, SN: FOC2423NXUH NAME: "HundredGigE0/2/0/0", DESCR: "100GE-SR4-S QSFP Module" PID: QSFP-100G-SR4-S , VID: V03, SN: INL23332207 NAME: "HundredGigE0/2/0/1", DESCR: "100GE-LR4-S QSFP Module" PID: QSFP-100G-LR4-S , VID: V01, SN: FNS213210US NAME: "HundredGigE0/2/0/2", DESCR: "100GE-SM-SR QSFP Module" PID: QSFP-100G-SM-SR , VID: V02, SN: FNS22151794 NAME: "HundredGigE0/2/0/3", DESCR: "100GE-FR-S QSFP Module" , VID: V01, SN: INL24080848 PID: QSFP-100G-FR-S NAME: "HundredGigE0/2/0/4", DESCR: "100GE-CWDM4-S QSFP Module" PID: QSFP-100G-CWDM4-S , VID: V01, SN: JFQ210530CN NAME: "HundredGigE0/2/0/5", DESCR: "100GE-ER4L QSFP Module" PID: QSFP-100G-ER4L-S , VID: V01, SN: FLJ2150J00V NAME: "HundredGigE0/2/0/6", DESCR: "100GE-40/100-SRBD QSFP Module" PID: QSFP-40/100-SRBD , VID: V01, SN: FOF2145N0LA NAME: "HundredGigE0/2/0/7", DESCR: "100GE-PSM4 QSFP Module" PID: QSFP-100G-PSM4-S , VID: V01, SN: CVR220500MB NAME: "HundredGigE0/2/0/8", DESCR: "100GE-LR-S QSFP Module" PID: QSFP-100G-LR-S , VID: V01, SN: FBN2415B004

4x100GE Breakout

The QSFP-DD modules support 400Gbps. You can configure 4x100G breakout on a QSFP-DD port and use it as four individual 100G interfaces.

This feature is supported on the following variant of Cisco ASR 9000 Series Line Cards:

- A99-10X400GE-X-SE
- A99-10X400GE-X-TR
- A9K-20HG-FLEX-SE
- A9K-20HG-FLEX-TR
- A9K-8HG-FLEX-SE
- A9K-8HG-FLEX-TR

This example shows how to enable 4x100GE breakout:

RP/0/RSP0/CPU0:router(config) # hw-module location 0/8/CPU0 port 0 breakout 4xHundredGigE

The following output shows the breakout configuration:

RP/0/RP0/CPU0:router# show ipv4 interfaces brief | i 0/8/0/0/

HundredGigE0/8/0/0/0	10.0.9.1	Up	Up	vpn9
HundredGigE0/8/0/0/1	10.0.8.1	Up	Up	vpn8
HundredGigE0/8/0/0/2	10.0.7.1	Up	Up	vpn7
HundredGigE0/8/0/0/3	10.0.2.1	Up	Up	vpn2

RP/0/RP0/CPU0:router# show interface hundredGigE 0/8/0/0/*

HundredGigE0/8/0/0/0 is up, line protocol is up Interface state transitions: 1 Hardware is HundredGigE, address is 7c21.0e33.35f0 (bia 7c21.0e33.35f0) Layer 1 Transport Mode is LAN Description: noshut-interface Internet address is 10.0.9.1/24 MTU 9216 bytes, BW 10000000 Kbit (Max: 10000000 Kbit) reliability 255/255, txload 0/255, rxload 0/255 Encapsulation ARPA, Full-duplex, 100000Mb/s, 400GBASE-DR4, 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 08:07:28 ARP type ARPA, ARP timeout 04:00:00 Last input 00:00:00, output 00:00:00 Last clearing of "show interface" counters never 5 minute input rate 1000 bits/sec, 1 packets/sec 5 minute output rate 1000 bits/sec, 1 packets/sec 18400 packets input, 4669038 bytes, 0 total input drops 0 drops for unrecognized upper-level protocol Received 1 broadcast packets, 18397 multicast packets 0 runts, 0 giants, 0 throttles, 0 parity 0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort 19128 packets output, 4732388 bytes, 0 total output drops Output 1 broadcast packets, 19125 multicast packets 0 output errors, 0 underruns, 0 applique, 0 resets 0 output buffer failures, 0 output buffers swapped out

1 carrier transitions HundredGigE0/8/0/0/1 is up, line protocol is up Interface state transitions: 1 Hardware is HundredGigE, address is 7c21.0e33.35f1 (bia 7c21.0e33.35f1) Layer 1 Transport Mode is LAN Description: noshut-interface Internet address is 10.0.8.1/24 MTU 9216 bytes, BW 10000000 Kbit (Max: 10000000 Kbit) reliability 255/255, txload 0/255, rxload 0/255 Encapsulation ARPA, Full-duplex, 100000Mb/s, 400GBASE-DR4, 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 08:07:28 ARP type ARPA, ARP timeout 04:00:00 Last input 00:00:00, output 00:00:00 Last clearing of "show interface" counters never 5 minute input rate 1000 bits/sec, 1 packets/sec 5 minute output rate 1000 bits/sec, 1 packets/sec 18408 packets input, 4677424 bytes, 5 total input drops 0 drops for unrecognized upper-level protocol Received 1 broadcast packets, 18405 multicast packets 0 runts, 0 giants, 0 throttles, 0 parity 0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort 19118 packets output, 4725316 bytes, 0 total output drops Output 1 broadcast packets, 19115 multicast packets 0 output errors, 0 underruns, 0 applique, 0 resets 0 output buffer failures, 0 output buffers swapped out 1 carrier transitions HundredGigE0/8/0/0/2 is up, line protocol is up Interface state transitions: 1 Hardware is HundredGigE, address is 7c21.0e33.35f2 (bia 7c21.0e33.35f2) Layer 1 Transport Mode is LAN Description: noshut-interface Internet address is 10.0.7.1/24 MTU 9216 bytes, BW 10000000 Kbit (Max: 10000000 Kbit) reliability 255/255, txload 0/255, rxload 0/255 Encapsulation ARPA, Full-duplex, 100000Mb/s, 400GBASE-DR4, 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 08:07:28 ARP type ARPA, ARP timeout 04:00:00 Last input 00:00:00, output 00:00:00 Last clearing of "show interface" counters never 5 minute input rate 1000 bits/sec, 1 packets/sec 5 minute output rate 1000 bits/sec, 1 packets/sec 18446 packets input, 4676812 bytes, 0 total input drops 0 drops for unrecognized upper-level protocol Received 1 broadcast packets, 18443 multicast packets 0 runts, 0 giants, 0 throttles, 0 parity 0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort 19184 packets output, 4729960 bytes, 0 total output drops Output 1 broadcast packets, 19181 multicast packets 0 output errors, 0 underruns, 0 applique, 0 resets 0 output buffer failures, 0 output buffers swapped out 1 carrier transitions HundredGigE0/8/0/0/3 is up, line protocol is up

Interface state transitions: 1

```
Hardware is HundredGigE, address is 7c21.0e33.35f3 (bia 7c21.0e33.35f3)
Layer 1 Transport Mode is LAN
Description: noshut-interface
Internet address is 10.0.2.1/24
MTU 9216 bytes, BW 10000000 Kbit (Max: 10000000 Kbit)
  reliability 255/255, txload 0/255, rxload 0/255
Encapsulation ARPA,
Full-duplex, 100000Mb/s, 400GBASE-DR4, 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 08:07:28
ARP type ARPA, ARP timeout 04:00:00
Last input 00:00:00, output 00:00:00
Last clearing of "show interface" counters never
5 minute input rate 1000 bits/sec, 1 packets/sec
5 minute output rate 1000 bits/sec, 1 packets/sec
   18444 packets input, 4677024 bytes, 0 total input drops
   0 drops for unrecognized upper-level protocol
   Received 1 broadcast packets, 18441 multicast packets
            0 runts, 0 giants, 0 throttles, 0 parity
   0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
   19170 packets output, 4734446 bytes, 0 total output drops
   Output 1 broadcast packets, 19167 multicast packets
   0 output errors, 0 underruns, 0 applique, 0 resets
   0 output buffer failures, 0 output buffers swapped out
   1 carrier transitions
```

Reloading, Shutting Down, or Power Cycling a Node

Use the commands described in this section to reload the Cisco IOS XR software on the active RSP or on any specified node in the system. This section also describes the commands used to administratively shut down a node and power a node on or off.

Command	Description
hw-module location <i>node-id</i> power disable	This command administratively turns the power off for a node. It is entered in administration configuration mode. The changes do not take effect until you enter the commit command.
	To power on a node, use the no form of this command.
	Note This command cannot be used to disable power on the RSP from which the command is entered.
hw-module location node-id reload	This command works in EXEC mode and reloads the Cisco IOS XR software on a specific node or all nodes. To specify all nodes, enter the all keyword in place of the <i>node-id</i> argument. The node reloads with the current running configuration and active software set for that node.
hw-module unshut location node-id	This command must be entered in the configuration mode. This command is used to administratively bring up the specified node.

Table 35: Commands to Reload, Shut Down, or Power Cycle a Node



Note

When you use the **hw-module shutdown location** *node-id* command to a line card, you must wait until the configuration is applied before removing the line card. Removal of the line card before the shutdown may result in a hardware issue.

Reloading the Active RSP

The **reload** command causes the active RSP to reload the Cisco IOS XR software according to the configuration register setting. This setting determines how the active RSP acts when reloaded.

This section contains instructions to reload the Cisco IOS XR software and return to EXEC mode. For instructions to use the **reload** command for entering ROM Monitor bootstrap mode, see *ROM Monitor Configuration Guide for Cisco ASR 9000 Routers*.

<u>/</u>!

Caution

Because the **reload** command causes the active RSP to go off line and either reload the Cisco IOS XR software or enter ROM Monitor mode, the router experiences a loss of service unless a redundant standby RSP is installed and in "ready" state. To display the status of the standby RSP, use the **show redundancy** command in EXEC mode.

SUMMARY STEPS

- 1. show redundancy
- 2. admin
- **3**. show variables boot
- 4. (Optional) config-register register-value
- 5. admin
- 6. reload

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	show redundancy	Displays the RSP redundancy status.
	Example: RP/0/RSP0/CPU0:router# show redundancy	• If a standby RSP is in "ready" redundancy state, the reload command also causes the router to gracefully fail over to the standby RSP.
Step 2	admin	Enters administration EXEC mode.
	Example:	
	RP/0/RSP0/CPU0:router# admin	
Step 3	show variables boot	Displays the configuration register setting.

	Command or Action	Purpose
	Example: RP/0/RSP0/CPU0:router(admin)# show variables boot	 Enter this command in administration EXEC mode. For normal operations, the configuration register setting is 0x102 or 0x2102, which causes the active RSP to reload the Cisco IOS XR software. Verify that the configuration register setting is 0x102 or 0x2102. If it is not, complete Step 4, on page 459 to reset the configuration register to 0x102 or 0x2102. Note For instructions on how to enter ROM Monitor bootstrap mode, see <i>ROM Monitor Configuration Guide for Cisco ASR 9000 Routers</i>.
Step 4	(Optional) config-register register-value Example: RP/0/RSP0/CPU0:router(admin)# config-register 0x102	Sets the configuration register to the respective value. This step is necessary only if the register is not set to the respective value ($0x102$ or $0x2102$) in the running configuration. You can use either $0x102$ or $0x2102$. Both these values specify the same functionality, as bit 13 in $0x2102$ is not significant for Cisco IOS XR software.
Step 5	admin Example: RP/0/RSP0/CPU0:router# admin	Enters administration EXEC mode.
Step 6	reload Example: RP/0/RSP0/CPU0:router# reload	 Reloads the active RSP according to the configuration register setting. If the setting is 0x102 or 0x2102, then the RSP reloads the Cisco IOS XR software. If the standby RSP is in "ready" redundancy state, the router switches over to the standby RSP. If a standby RSP is not installed or not in a "ready" state, the router experiences a loss of service while the active RSP is reloading the Cisco IOS XR software.

Flash Disk Recovery

When an RSP is power cycled or experiences an ungraceful reset, the boot disk (PCMCIA flash disk used to boot the card) may experience a file-system corruption. If this occurs, an error message is displayed and the RSP fails to boot. The corrupted flash disk is automatically reformatted and the Cisco IOS XR software is restored from the designated system controller (DSC) for the system.

For example, if a flash disk for an RSP is corrupted, the RP fails to boot and the following error message is displayed:

If this occurs, then the flash disk is automatically reformatted and the Cisco IOS XR software is restored to the flash disk.

8

Note If the flash disk is badly damaged and cannot be reformatted, the disk must be replaced.

If the corrupted flash disk is the DSC, then the router fails over to the standby DSC. If no standby DSC is installed, then the system fails to boot.

Using Controller Commands to Manage Hardware Components

The **controller**, **controllers**, and **show controllers** commands are used to manage and display settings for various hardware components, including the switch fabric management, Ethernet control plane, and interface manager. These commands are primarily diagnostic and related to driver-level details. The information available with these commands varies widely and is hardware specific.

For information on the use of these commands, see Interface and Hardware Component Command Reference for Cisco ASR 9000 Series Routers.

Formatting Hard Drives, Flash Drives, and Other Storage Devices

To format a storage device on the router, use the **format** command in EXEC mode.

Caution Formatting a storage device deletes all data on that device.

The following command syntax is used:

format filesystem: [options]

Table 36: format command Syntax Description, on page 461 describes the format command syntax.

Variable	Description
filesystem	Specifies the memory device to format. The supported file systems are:
	• bootflash:
	• compactflash:
	• configflash:
• harddisk:	
• harddiska:	
	• disk0:
	• disk1:
	Enter format? to see the devices supported on your router.
options	Enter format <i>filesystem:</i> ? to see the available options.
	For more information, see System Management Command Reference for Cisco ASR 9000 Series Routers.

Table 36: format command Syntax Description

In the following example, the **format** command is used to format the hard disk:

RP/0/RSP0/CPU0:router# format harddisk:

Removing and Replacing Cards

This section describes card replacement issues and procedures.

Removing Line Cards

Line cards are designed for online insertion and removal (OIR). A line card is a single card that contains all service processing functions and physical line interfaces.

The OIR feature allows you to remove and replace cards without removing power to the card or chassis. Removing a card interrupts all traffic passing through the card, but it does not remove the card configuration.

When you remove a card, the configuration remains for all interfaces, but the interfaces do not appear in the output of the **show interfaces** command. You can view interface configurations by entering the **show running-config** command. The following example shows how the configuration appears when a card is removed:

RP/0/RSP0/CPU0:router# show running-config

```
Building configuration...
hostname router
router ospf 3269
area 0
interface POS0/3/0/0
cost 20
!
interface preconfigure POS0/3/0/0
```

```
ipv4 address 10.10.50.1 255.255.255.0
!
interface preconfigure POS0/3/0/1
description POS0/3/0/1
shutdown
!
interface preconfigure POS0/3/0/2
description POS0/3/0/2
shutdown
!
interface preconfigure POS0/3/0/3
description POS0/3/0/3
shutdown
!
```

In this example, the line card in slot 3 is removed, and the interface configuration for all four interfaces changes to "interface preconfigure." However, the "router ospf" reference to a slot 3 interface does not change. If you replace a line card with another line card that uses the same media type and port count, the configuration becomes active on the replacement card.

To remove the configuration for a slot after a card is removed, use the **no interface preconfigure** command to remove all interface configuration statements for that card in the running configuration. In addition, search the configuration for any references to the removed interfaces, such as the "router ospf" reference to slot 3 in the preceding example.

To remove the configuration for a slot when a card is installed, use the **no interface** command to remove all interface configuration statements for that card in the running configuration. In addition, search the configuration for any references to the removed interfaces.

Each line card supports a specific media type (Packet over SONET/SDH [POS] or Ethernet, for example) and port count. If you replace a line card with one that supports a different media type or port count, you should review the configuration and revise it to support the replacement line card.

Replacing a Line Card with the Same Media Type and Port Count

When you replace a line card or PLIM with a card that is of the same media type and has the same port count as the replaced card, the guidelines in the Removing Line Cards, on page 461 apply. Because the replacement card is of the same media type and port count, no special procedures are required for card removal and replacement.

Replacing a Line Card with the Same Media Type and a Different Port Count

When you replace a line card with a card that is of the same media type with a different port count, the guidelines in Removing Line Cards, on page 461 apply.

If the new card has a greater port count than the replaced card, the configuration applies to the corresponding lower port numbers, and the ports that did not exist on the replaced card have no configuration and come up in the shutdown state.

If the new card supports fewer ports, the existing configuration for the corresponding number of ports on the new card set is applied. The previous configuration for the removed ports remains in interface preconfigure state, as shown in the following example:

```
RP/0/RSP0/CPU0:router# show running-config
```

```
Building configuration...
```

```
hostname rtp-gsr1
interface POS0/3/0/0
ipv4 address 10.10.50.1 255.255.255.0
!
interface preconfigure POS0/3/0/1
description POS0/3/0/1
shutdown
!
interface preconfigure POS0/3/0/2
description POS0/3/0/2
shutdown
!
interface preconfigure POS0/3/0/3
description POS0/3/0/3
shutdown
!
```

In the preceding example, a four-port card has been replaced with a single-port card. The configuration from port 1 on the four-port card is applied to the single port on the replacement card, and the remaining port configurations change to "interface preconfigure." To remove the configuration for the missing interfaces, use the **no interface preconfigure** command. In addition, search for and remove any configuration references to the removed interfaces.

Whenever you replace a line card with the same media type and a different port count, review the running configuration in the router and revise the configuration as necessary.

Replacing a Line Card or PLIM with a Different Media Type

When you replace a line card or PLIM with a card that is of a different media type (for example, if you replace a POS PLIM with an Ethernet PLIM), the guidelines in Removing Line Cards, on page 461 apply. Review the running configuration in the router and revise the configuration as necessary for the new media type.

Proactive Line Card Shut Down

The proactive line card shutdown feature enables powering down line cards automatically when the power that is drawn by the router exceeds configured threshold. The sequence of powering down LCs is based on the shutdown priorities that are assigned to them. The LC, however, does not boot automatically even after the router power draw is back to normal below the shutdown threshold. Therefore, you must manually bring up the LC by either reloading or OIR method.

Instead of provisioning more power to the router in worst case power draw scenarios, this feature helps in saving power by powering down the LC.

As part of this feature, you must configure two threshold values:

- Syslog Threshold—This value serves as a syslog warning threshold. If the router power draw exceeds the syslog threshold, then a warning error message is captured at the console. This warning message facilitates the user to take any preventive action.
- Shutdown Threshold—This value triggers the shutdown of line cards. If the router power draw exceeds the shutdown threshold, then the line cards are shutdown based on the priorities that are assigned to them. The system monitors the power draw for every 10 seconds.

Shut Down Priorities for Powering Down the LC

You can configure a shutdown priority value of 0 to 19. A line card with lower value has the highest priority. By default, a priority of 20 is assigned to all the LC in the router.

When two or more LCs have equal priorities that are assigned, then the slot number takes precedence in the priority calculation. A lower slot number has the highest priority. For instance, when two LCs at slot 0 and slot 19 have the priority set as 10, then LC in slot 0 has higher priority than the LC in slot 19.

Proactive Line Card Shut Down Implementation Consideration

Consider the following points while configuring proactive line card shut down feature:

- Shutdown threshold must be greater than the current system power draw.
- Shutdown threshold must be greater than the syslog threshold.
- Shutdown threshold must not be less than 3500 Watts.
- In Cisco IOS XR 32-bit OS, the priority of a LC is checked only when the LC is in IOS XR RUN state.
- In Cisco IOS XR 64-bit OS, the priority of a LC is checked only when the LC is in Operational state.

Configure Proactive Line Card Shut Down

Cisco IOS XR 32-bit

Configuring proactive line card shutdown includes:

- · Assigning priorities to the line cards
- Configuring syslog threshold
- Configuring shutdown threshold

In this example, syslog threshold of 5000 W and shutdown threshold of 6000 W is configured along with the LC priorities:

```
config
power budget enforcement progressive
priority 1
    location 0/1/CPU0
    location 0/5/CPU0
!
priority 5
    location 0/4/CPU0
!
priority 11
    location 0/2/CPU0
!
priority 20
    location 0/3/CPU0
!
syslog-threshhold 5000 W
shutdown-threshhold 6000 W
```

The following error message is seen when power draw exceeds the shutdown threshold:

```
RP/0/RP0/CPU0:Mar 8 11:42:00.146 : shelfmgr[406]: %PLATFORM-SHELFMGR-1-INRESET_ALARM :
Power off node 0/10/CPU0 due to multiple critical alarms, putting into IN_RESET state
RP/0/RP0/CPU0:Mar 8 11:42:10.948 : envmon[209]: %PLATFORM-ENVMON-2-PWR_EXCEEDED_SHUTDOWN
: Slot 0/3/CPU0 priority 20 is being shutdown,current power usage 6746 W exceeds the
configured threshold of 6000 W
```

Cisco IOS XR 64-bit

Configuring proactive line card shutdown includes:

- Configuring syslog threshold
- Configuring shutdown threshold
- Assigning priorities to the line cards

In this example, syslog threshold of 3300 W and shutdown threshold of 3500 W is configured along with LC priorities:

The following error message is seen when power draw exceed the syslog threshold:

0/RSP0/ADMIN0:Feb 22 11:44:38.566 UTC: envmon[4202]: %PWR_MGMT-ENVMON-3-PWR_EXCEEDED_WARN
:
Chassis power usage 3448 W has exceeded the configured warning threshold of 3300 W

The following error message is seen when power draw exceeds the shutdown threshold:

0/RSP0/ADMIN0:Feb 22 11:44:38.567 UTC: envmon[4202]: %PKT_INFRA-FM-3-FAULT_MAJOR : ALARM_MAJOR :Node shutdown by Progressive power-mgmt mode :DECLARE :0/0: Node priority 10, Chassis power draw 3570 W exceeded shutdown threshold 3500 W

Advanced Power Management

Advanced power management feature enables powering down the unused line card slices.

This feature helps to manage power consumption, as the slices that do not have any services enabled are power down. Later these slices can be powered when a new service is enabled on them.

This feature is supported on the Cisco ASR 9000 4th Generation Ethernet line cards.

Configuring Advance Power Management

This procedure shows how to configure advance power management.

In this example these slices are powered down:

- slice 0, and 7 of the line card in the node 0 location
- slice 3, and 6 of the line card in the node 1 location

You should reload the line card for the configuration changes to take effect.

```
Router# configure
Router (config)# hw-module location 0/0/CPU0 slice 0 power-down
Router (config)# hw-module location 0/0/CPU0 slice 7 power-down
Router (config)# hw-module location 0/1/CPU0 slice 3 power-down
Router (config)# hw-module location 0/1/CPU0 slice 6 power-down
Router (config)# commit
Router (config)# end
Router # admin
Router (sysadmin-vm)# hw-module location 0/0 reload
Router (sysadmin-vm)# hw-module location 0/1 reload
```

Running Configuration

config hw-module location 0/0/CPU0 slice 0 power-down hw-module location 0/0/CPU0 slice 7 power-down hw-module location 0/1/CPU0 slice 3 power-down hw-module location 0/1/CPU0 slice 6 power-down

Verification

Router# show apm psm status

```
PSM Status

------

PSM Client Status

DIAG0: Not registered

DIAG1: Registered

0/1 PSA: Registered
```

LC Status

Line Card	Slice	Config	Status	DIAG0	DIAG1	PSA
0/0/CPU0	0	On	Completed	Not registered	Completed	Not present
	1	On	Completed	Not registered	Completed	Not present
	2	On	Completed	Not registered	Completed	Not present
	3	On	Completed	Not registered	Completed	Not present

	4	On	Completed	Not registered	Completed	Not present
	5	On	Completed	Not registered	Completed	Not present
	6	On	Completed	Not registered	Completed	Not present
	7	On	Completed	Not registered	Completed	Not present
0/1/CPU0	0	On	Completed	Not registered	Completed	Completed
	1	On	Completed	Not registered	Completed	Completed
	2	On	Completed	Not registered	Completed	Completed
	3	On	Completed	Not registered	Completed	Completed
	4	On	Completed	Not registered	Completed	Completed
	5	On	Completed	Not registered	Completed	Completed
	6	On	Completed	Not registered	Completed	Completed
	7	On	Completed	Not registered	Completed	Completed

```
      New configuration after line card reboots

      Line Card
      Slice
      New Config

      0/0/CPU0
      0
      Down

      7
      Down

      0/1/CPU0
      3
      Down
```

6

Overview of Erase and Wipeout Disk Memory

Down

Below two methods are used to delete the data from a RSP and line card. These methods are used based on your requirements:

- Erase Disk Memory
- · Wipe Out Disk Memory

Erase Disk Memory

The Erase Disk Memory operation clears the disk memory of RSPs and line cards. However, the deleted data is recoverable using recovery tools. The erase disk memory operation can be performed for quick sanitization of the card before reusing it in another device within the control space of your network or organization.

How to Erase Disk Memory

Erasing disk memory operation uses zapdisk feature to erase the disk memory from the RSP and line card.

Erasing disk memory is done in three steps. First, you enable the zapdisk feature, later identify the card where zapdisk is supported. Next, activate the zapdisk operation on the card:

1. Enable zapdisk feature on the router.

Example:

sysadmin-vm# zapdisk set

2. Find out the card location where the zapdisk feature is supported using the **show zapdisk locations** command.

Example:

Router# show zapdisk locations

0/RSP1 Fully qualified location specification
0/7 Fully qualified location specification
0/4 Fully qualified location specification
all all locations

3. Start the zapdisk operation on a specific node location or all node locations to erase disk memory.

Note You can run the zapdisk operation on all RSPs and line cards except the active RSP where zapdisk service is running in an active role.

After the zapdisk process is completed, the system clears all data and shuts down the card.

This example runs the zapdisk operation on the node location 0/4:

```
Router# zapdisk start location 0/4
Action on designated location is in progress, more detail logs will be located in sysadmin
at
/misc/disk1/tftpboot/zapdisk.log once action is completed
```

Note

After deleting the data, remove the card from the slot, and do not reload the card or the router. If you reload the card or the router without removing the card, the data is reloaded into the card.

In the event when you must return or trash a card, the data in the disk memory should be permanently deleted. Therefore, the erase disk memory feature is not advice. You should use the enhanced version of the erase disk memory feature called Wipe Out Disk Memory.

Wipe Out Disk Memory

The Wipe Out Disk Memory feature deletes data permanently from the disk memory of RSPs and line cards. The erased data is non-recoverable. We recommend this action when you perform a return material authorization (RMA) of a card to prevent pilferage of sensitive data.

How to Wipe out Disk Memory

Wiping out disk memory actions are performed in the ROMMON mode. Generally to boot into ROMMON mode, the **config-register boot-mode rom-monitor** command is executed from the admin mode. However, the command is not available in Cisco IOS XR 64 bit OS. Therefore you must follow the below sequence to boot into ROMMON mode:

- 1. Reload the router
- 2. Break into the BIOS menu and select ROMMON
- **3.** Wipe out disk memory in ROMMON

Reload the router

Before reloading the router, ensure that the redundant RP is disabled in dual-RP routers and console is connected:

```
sysadmin-vm:0_RSP0# hw-module location all reload
```

Break into the BIOS menu and select ROMMON

- 1. While the router boots, press CTRL+C to break into BIOS menu.
- 2. To enter into ROMMON mode, select the Boot to ROMMON option from the available boot options:

```
Please select the operating system and the boot device:
    1) Boot to ROMMON
    2) IOS-XR 64 bit Boot previously installed image
    3) IOS-XR 64 bit Mgmt Network boot using DHCP server
    4) IOS-XR 64 bit Mgmt Network boot using local settings (iPXE)
    (Press 'p' for more option)
Selection [1/2/3/4]: 1
Selected Boot to ROMMON , Continue ? Y/N: y
```

rommon 1 >

Wipe out disk memory in ROMMON

1. Go to Privilege Mode.

rommon > priv

2. Select the hderase option.

```
rommon > hderase
        SATA HD(0x4,0x0,0x0):
        Model : <Model number>
    Serial No : <serial number>
    Sanitize Crypto Scramble Erase Supported
    Sanitize State : Idle
All the contents on this Drive will be Erased
Do you wish to continue?(Y/N)
Y
```

The data is permanently erased.

Upgrading the CPU Controller Bits

Use this procedure to upgrade the CPU controller bits on all nodes that are installed in the router or on a specific node.

SUMMARY STEPS

- 1. admin
- **2.** upgrade cpuctribits {all | location *node-id*}

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	admin	Enters administration EXEC mode.
	Example:	
	RP/0/RSP0/CPU0:router# admin	
Step 2	upgrade cpuctribits {all location node-id}	Upgrades the CPU controller bits on all nodes in the router.
	Example:	Use the location <i>node-id</i> keyword and argument to upgrade the CPU controller bits on a specific node.
	<pre>RP/0/RSP0/CPU0:router(admin)# upgrade cpuctrlbits all</pre>	

Examples

The following example shows how to upgrade the CPU controller bits on all nodes in a router:

```
RP/0/RSP0/CPU0:router# admin
RP/0/RSP0/CPU0:router(admin)# upgrade cpucrtlbits all
Please do not power cycle, reload the router or reset any nodes until all upgrades are
completed.
Please check the syslog to make sure that all nodes are upgraded successfully.
If you need to perform multiple upgrades, please wait for current upgrade to be completed
before proceeding to another upgrade. Failure to do so may render the cards under upgrade
to be unusable.
```

Configuring Port Modes

This section describes how to configure the various port modes on a router, port expansion card, or a line card.

Feature Name	Release Information	Description
1GbE Port Mode on SFP+ and SFP28 Ports	Release 7.9.1	It is now possible to use the 10GbE SFP+ and SFP28 ports to transmit 1GbE traffic on the routers or line cards. The number of ports on which you can enable 1GbE port mode has been increased from the previous releases.
		The details of the number of ports on which you can configure 1GbE port mode on the Cisco ASR 9000 Series routers, port expansion card, and line cards is as follows:
		• Cisco ASR 9903 - 20 SFP+ ports
		Cisco ASR 9902 - 24 SFP+ and 16 SFP28 ports
		• A9903-8HG-PEC - 40 SFP+ ports
		• A9K-4HG-FLEX-SE/TR - 24 SFP+ and 16 SFP28 ports
		• A99-4HG-FLEX-SE/TR - 24 SFP+ and 16 SFP28 ports
		A9K-4HG-FLEX-FC - 24 SFP+ and 16 SFP28 ports
		A99-4HG-FLEX-FC - 24 SFP+ and 16 SFP28 ports
		Use the hw-module location breakout and hw-module location slice config mode commands to enable 1GbE port mode configuration.

Table 37: Feature History Table

Configuring Port Mode in Cisco A9903-8HG-PEC

You can configure the SFP+ ports on the Cisco A9903-8HG-PEC (0.8T PEC card) in 10GbE, 25GbE, or 1GbE mode. By default, all ports are in 10GbE mode.

For more information on the ports, see the Installing Modules and Cables in the Chassis chapter in the *Cisco* ASR 9000 Series Fixed-Port Routers Hardware Installation Guide.

To configure a port, use the **hw-module location slice config mode** command.

The port mode configuration is per slice and not per port. Also, the 48 ports on Cisco A9903-8HG-PEC are available in slices 4 and 5. Each slice can be configured for 10GbE, 25GbE, or 1GbE port mode.

The number of ports available on the card depends on the port mode configured on each slice. This table shows the port mode, and available ports on Cisco A9903-8HG-PEC:

Port Mode	Number of Ports	Available Port Numbers
1GbE	48	On slice 4: port numbers 0–23
	24 on each slice	On slice 5: port numbers 24–47
10GbE	48	On slice 4: port numbers 0–23
	24 on each slice	On slice 5: port numbers 24–47
25GbE	32	On slice 4: port numbers 0–15
	16 on each slice	On slice 5: port numbers 32–47

Table 38: Port Modes, and Available Ports on Cisco A9903-8HG-PEC

Configuration

This sample configuration shows how to configure all ports on Cisco A9903-8HG-PEC in 25GbE mode:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#hw-module location 0/0/CPU0 slice 4 config-mode
4x25GE,4x25GE,4x25GE
RP/0/RP0/CPU0:ios(config)#hw-module location 0/0/CPU0 slice 5 config-mode
4x25GE,4x25GE,4x25GE
RP/0/RP0/CPU0:ios(config)#commit
```

This sample configuration shows how to configure all ports on Cisco A9903-8HG-PEC in 1GbE mode:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#hw-module location 0/0/CPU0 slice 4 config-mode
10x1GE,10x1GE,4x10GE,UNUSED
RP/0/RP0/CPU0:ios(config)#hw-module location 0/0/CPU0 slice 5 config-mode
10x1GE,10x1GE,4x10GE,UNUSED
RP/0/RP0/CPU0:ios(config)#commit
```

By default, all ports are in 10GbE mode. To revert to the default mode, use **no** form of the hw-module location <node> slice <number> config-mode command. This sample shows how to revert slice 4 ports to the default 10GbE mode:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#no hw-module location 0/0/CPU0 slice 4 config-mode
4x25GE,4x25GE,4x25GE,4x25GE
RP/0/RP0/CPU0:ios(config)#commit
```

Verification

Use the **show interface brief** command to verify the ports in slice 4 and 5 configured in 25GbE mode:

```
RP/0/RP0/CPU0:ios#show interfaces brief
Intf
       Intf
                  LineP
                             Encap MTU
                                            BW
        State
                   State
                              Type (byte) (Kbps)
Name
_____
           _____
[SNIP]
TF0/0/1/0 admin-down admin-down ARPA 1514 25000000 /* slice 4 port numbers starts here */
TF0/0/1/1 admin-down admin-down ARPA 1514 25000000
TF0/0/1/2 admin-down admin-down ARPA 1514 25000000
TF0/0/1/3 admin-down admin-down ARPA 1514 25000000
```

```
TF0/0/1/4 admin-down admin-down ARPA 1514 25000000
TF0/0/1/5 admin-down admin-down ARPA 1514 25000000
TF0/0/1/6 admin-down admin-down ARPA 1514 25000000
TF0/0/1/7 admin-down admin-down ARPA 1514 25000000
TF0/0/1/8 admin-down admin-down ARPA 1514 25000000
TF0/0/1/9 admin-down admin-down ARPA 1514 25000000
TF0/0/1/10 admin-down admin-down ARPA 1514 25000000
TF0/0/1/11 admin-down admin-down ARPA 1514 25000000
TF0/0/1/12 admin-down admin-down ARPA 1514 25000000
TF0/0/1/13 admin-down admin-down ARPA 1514 25000000
TF0/0/1/14 admin-down admin-down ARPA 1514 25000000
TF0/0/1/15 admin-down admin-down ARPA 1514 25000000
TF0/0/1/32 admin-down admin-down ARPA 1514 25000000 /* slice 5 port numbers starts here */
TF0/0/1/33 admin-down admin-down ARPA 1514 25000000
TF0/0/1/34 admin-down admin-down ARPA 1514 25000000
TF0/0/1/35 admin-down admin-down ARPA 1514 25000000
TF0/0/1/36 admin-down admin-down ARPA 1514 25000000
TF0/0/1/37 admin-down admin-down ARPA 1514 25000000
TF0/0/1/38 admin-down admin-down ARPA 1514 25000000
TF0/0/1/39 admin-down admin-down ARPA 1514 25000000
TF0/0/1/40 admin-down admin-down ARPA 1514 25000000
TF0/0/1/41 admin-down admin-down ARPA 1514 25000000
TF0/0/1/42 admin-down admin-down ARPA 1514 25000000
TF0/0/1/43 admin-down admin-down ARPA 1514 25000000
TF0/0/1/44 admin-down admin-down ARPA 1514 25000000
TF0/0/1/45 admin-down admin-down ARPA 1514 25000000
TF0/0/1/46 admin-down admin-down ARPA 1514 25000000
TF0/0/1/47 admin-down admin-down ARPA 1514 25000000
```

This show command displays all 48 ports on Cisco A9903-8HG-PEC in 10GbE mode:

RP/0/RP0/ Intf	CPU0:ios# sh o	ow interface	es bri Encar	ief o MTU		BW								
Name	State	State	Туре	(byte	∋)	(Kbps)								
Te0/0/1/0) admin-down	admin-down	ARPA	1514	10	000000	/*	slice	4	port	numbers	starts	here	*/
Te0/0/1/1	admin-down	admin-down	ARPA	1514	100	000000								
Te0/0/1/2	2 admin-down	admin-down	ARPA	1514	100	000000								
Te0/0/1/3	8 admin-down	admin-down	ARPA	1514	100	000000								
Te0/0/1/4	l admin-down	admin-down	ARPA	1514	100	000000								
Te0/0/1/5	ā admin-down	admin-down	ARPA	1514	100	000000								
Te0/0/1/6	admin-down	admin-down	ARPA	1514	100	000000								
Te0/0/1/7	/ admin-down	admin-down	ARPA	1514	100	000000								
Te0/0/1/8	3 admin-down	admin-down	ARPA	1514	100	000000								
Te0/0/1/9	admin-down	admin-down	ARPA	1514	100	000000								
Te0/0/1/1	10 admin-down	n admin-down	n ARPA	A 1514	1 1 (0000000)							
Te0/0/1/1	ll admin-down	n admin-down	n ARPA	A 1514	1 1 (0000000)							
Te0/0/1/1	12 admin-down	n admin-down	n ARPA	A 1514	1 1 (0000000)							
Te0/0/1/1	13 admin-down	n admin-down	n ARPA	A 1514	1 1 (0000000)							
Te0/0/1/1	4 admin-down	n admin-down	n ARPA	A 1514	1 1 (0000000)							
Te0/0/1/1	15 admin-down	n admin-down	n ARPA	A 1514	1 1 (0000000)							
Te0/0/1/1	l6 admin-down	n admin-down	n ARPA	A 1514	1 1 (0000000)							
Te0/0/1/1	17 admin-down	n admin-down	n ARPA	A 1514	1 1 (0000000)							
Te0/0/1/1	18 admin-down	n admin-down	n ARPA	A 1514	1 1 (0000000)							
Te0/0/1/1	19 admin-down	n admin-down	n ARPA	A 1514	1 1 (0000000)							
Te0/0/1/2	20 admin-down	n admin-down	n ARPA	A 1514	1 1 (0000000)							
Te0/0/1/2	21 admin-down	n admin-down	n ARPA	A 1514	1 1 (0000000)							
Te0/0/1/2	22 admin-down	n admin-down	n ARPA	A 1514	1 1 (0000000)							
Te0/0/1/2	23 admin-down	n admin-down	n ARPA	A 1514	1 1 (0000000)							
Te0/0/1/2	24 admin-down	n admin-down	n ARPZ	A 1514	1 1(0000000) /*	slice	e 5	5 port	number:	s start:	s here	*/
Te0/0/1/2	25 admin-down	n admin-down	n ARPA	A 1514	1 1 (0000000)							
Te0/0/1/2	26 admin-down	n admin-down	n ARPA	A 1514	1 1 (0000000)							
Te0/0/1/2	27 admin-down	n admin-dow	n ARPA	A 1514	1 1(0000000)							

Te0/0/1/28	admin-down	admin-down	ARPA	1514	10000000
Te0/0/1/29	admin-down	admin-down	ARPA	1514	10000000
Te0/0/1/30	admin-down	admin-down	ARPA	1514	10000000
Te0/0/1/31	admin-down	admin-down	ARPA	1514	10000000
Te0/0/1/32	admin-down	admin-down	ARPA	1514	10000000
Te0/0/1/33	admin-down	admin-down	ARPA	1514	10000000
Te0/0/1/34	admin-down	admin-down	ARPA	1514	10000000
Te0/0/1/35	admin-down	admin-down	ARPA	1514	10000000
Te0/0/1/36	admin-down	admin-down	ARPA	1514	10000000
Te0/0/1/37	admin-down	admin-down	ARPA	1514	10000000
Te0/0/1/38	admin-down	admin-down	ARPA	1514	10000000
Te0/0/1/39	admin-down	admin-down	ARPA	1514	10000000
Te0/0/1/40	admin-down	admin-down	ARPA	1514	10000000
Te0/0/1/41	admin-down	admin-down	ARPA	1514	10000000
Te0/0/1/42	admin-down	admin-down	ARPA	1514	10000000
Te0/0/1/43	admin-down	admin-down	ARPA	1514	10000000
Te0/0/1/44	admin-down	admin-down	ARPA	1514	10000000
Te0/0/1/45	admin-down	admin-down	ARPA	1514	10000000
Te0/0/1/46	admin-down	admin-down	ARPA	1514	10000000
Te0/0/1/47	admin-down	admin-down	ARPA	1514	10000000

This show command displays ports in slice 4 configured in 25GbE mode and slice 5 in 10GbE mode:

RP/0/RP0/	CPU0:ios# shc	w interface	es bri	ef									
Intf	Intf	LineP	Encap	MTU	BW								
Name	State	State	Туре	(byte	e) (Kbps)								
TF0/0/1/0	admin-down	admin-down	ARPA	1514	25000000	/*	slice	4	port	numbers	starts	here	*/
TF0/0/1/1	admin-down	admin-down	ARPA	1514	25000000				1				
TF0/0/1/2	admin-down	admin-down	ARPA	1514	25000000								
TF0/0/1/3	admin-down	admin-down	ARPA	1514	25000000								
TF0/0/1/4	admin-down	admin-down	ARPA	1514	25000000								
TF0/0/1/5	admin-down	admin-down	ARPA	1514	25000000								
TF0/0/1/6	admin-down	admin-down	ARPA	1514	25000000								
TF0/0/1/7	admin-down	admin-down	ARPA	1514	25000000								
TF0/0/1/8	admin-down	admin-down	ARPA	1514	25000000								
TF0/0/1/9	admin-down	admin-down	ARPA	1514	25000000								
TF0/0/1/1	0 admin-dowr	admin-dowr	n ARPA	. 1514	1 25000000								
TF0/0/1/1	1 admin-dowr	admin-dowr	n ARPA	. 1514	1 25000000								
TF0/0/1/1	2 admin-dowr	admin-dowr	n ARPA	. 1514	1 25000000								
TF0/0/1/1	3 admin-dowr	admin-dowr	n ARPA	. 1514	1 25000000								
TF0/0/1/1	4 admin-dowr	admin-dowr	n ARPA	. 1514	1 25000000								
TF0/0/1/1	5 admin-dowr	admin-dowr	n ARPA	. 1514	1 25000000								
Te0/0/1/2	4 admin-dowr	admin-dowr	n ARPA	1514	1000000	/*	slice	5	port	numbers	starts	here	*/
Te0/0/1/2	5 admin-dowr	ı admin-dowr	n ARPA	. 1514	10000000								
Te0/0/1/2	6 admin-dowr	ı admin-dowr	n ARPA	. 1514	10000000								
Te0/0/1/2	7 admin-dowr	ı admin-dowr	n ARPA	. 1514	10000000								
Te0/0/1/2	8 admin-dowr	ı admin-dowr	n ARPA	. 1514	10000000								
Te0/0/1/2	9 admin-dowr	admin-dowr	n ARPA	. 1514	10000000								
Te0/0/1/3	0 admin-dowr	ı admin-dowr	n ARPA	. 1514	10000000								
Te0/0/1/3	l admin-dowr	admin-dowr	n ARPA	. 1514	10000000								
Te0/0/1/3	2 admin-dowr	admin-dowr	n ARPA	. 1514	10000000								
Te0/0/1/3	3 admin-dowr	admin-dowr	n ARPA	. 1514	10000000								
Te0/0/1/3	4 admin-dowr	admin-dowr	n ARPA	. 1514	10000000								
Te0/0/1/3	5 admin-dowr	admin-dowr	n ARPA	. 1514	10000000								
Te0/0/1/3	6 admin-dowr	admin-dowr	n ARPA	. 1514	10000000								
Te0/0/1/3	/ admin-dowr	admin-dowr	n ARPA	1514	± 10000000								
TeU/U/1/3	8 admin-dowr	admin-dowr	1 ARPA	1514	± 10000000								
Te0/0/1/3	9 admin-dowr	admin-dowr	n ARPA	. 1514	± 10000000								
Te0/0/1/4	U admin-dowr	admin-dowr	n ARPA	1514	± 10000000								
TeU/U/1/4	1 admin-dowr	admin-dowr	1 ARPA	1514	± 10000000								
Te0/0/1/4	2 admin-dowr	admin-dowr	n ARPA	1514	± 10000000								
TeU/U/1/4	3 admin-dowr	i admin-dowr	n AKPA	. 1514	F TOOOOOOO								

Te0/0/1/44 admin-down admin-down ARPA 1514 1000000 Te0/0/1/45 admin-down admin-down ARPA 1514 1000000 Te0/0/1/46 admin-down admin-down ARPA 1514 1000000 Te0/0/1/47 admin-down admin-down ARPA 1514 1000000

This show command displays all 48 ports on Cisco A9903-8HG-PEC in 1GbE mode:

RP/0/RP0/CPU0:ios#show interfaces brief

Fri Feb 24 10:33:06.517 UTC

Intf Name	Intf State	LineP State	Encap Type	MTU (byte)	BW (Kbps)						
 N110	 מוו		Null	1500							
Mq0/RP0/CE	qu0/0Up	up	ARPA	1514	1000000						
Hu0/0/0/0	down	down	ARPA	1514	100000000						
Hu0/0/0/1	down	down	ARPA	1514	100000000						
Hu0/0/0/2	down	down	ARPA	1514	100000000						
Hu0/0/0/3	down	down	ARPA	1514	100000000						
Hu0/0/0/4	down	down	ARPA	1514	100000000						
Hu0/0/0/5	up	up	ARPA	9216	100000000						
Hu0/0/0/6	down	down	ARPA	1514	100000000						
Hu0/0/0/7	down	down	ARPA	1514	100000000						
Hu0/0/0/8	down	down	ARPA	1514	100000000						
Hu0/0/0/9	down	down	ARPA	1514	100000000						
Hu0/0/0/11	down	down	ARPA	1514	100000000						
Hu0/0/0/12	down	down	ARPA	1514	100000000						
Hu0/0/0/13	down	down	ARPA	1514	100000000						
Hu0/0/0/15	i up	up	ARPA	9216	100000000						
Te0/0/0/16	down	down	ARPA	1514	10000000						
Te0/0/0/17	down	down	ARPA	1514	10000000						
Te0/0/0/18	down	down	ARPA	1514	10000000						
Te0/0/0/19) down	down	ARPA	1514	10000000						
Te0/0/0/20) down	down	ARPA	1514	10000000						
Te0/0/0/21	. down	down	ARPA	1514	10000000						
Te0/0/0/22	down	down	ARPA	1514	10000000						
Te0/0/0/23	down	down	ARPA	1514	10000000						
Te0/0/0/24	down	down	ARPA	1514	10000000						
Te0/0/0/25	down	down	ARPA	1514	10000000						
Te0/0/0/26	down	down	ARPA	1514	10000000						
Te0/0/0/27	down	down	ARPA	1514	10000000						
Te0/0/0/28	down	down	ARPA	1514	10000000						
Te0/0/0/29	down	down	ARPA	1514	10000000						
Te0/0/0/30) down	down	ARPA	1514	10000000						
'l'e0/0/0/31	. down	down	ARPA	1514	10000000						
Te0/0/0/32	down	down	ARPA	1514	10000000						
Te0/0/0/33	down	down	ARPA	1514	10000000						
Te0/0/0/34	aown	down	ARPA	1514	10000000						
Te0/0/0/35	down	down	ARPA	1514	10000000	* alian	1 2027	numbowo	a+ = = + a	horo	* /
G10/0/1/0	dourn	down	ARPA	1514	1000000/	~ SIICe 4	a borr	numbers	Starts	nere	~ /
G10/0/1/1 Ci0/0/1/2	down	down	ARPA ADDA	1514	1000000						
G10/0/1/2	down	down	AREA	1514	1000000						
Gi0/0/1/3	down	down	ARPA	1514	1000000						
G10/0/1/4	down	down	ADDA	1514	1000000						
Gi0/0/1/6	down	down	ARPA	1514	1000000						
Gi0/0/1/7	down	down	ARPA	1514	1000000						
Gi0/0/1/8	down	down	ARPA	1514	1000000						
Gi0/0/1/9	down	down	ARPA	1514	1000000						
Gi0/0/1/10) down	down	ARPA	1514	1000000						
Gi0/0/1/11	. down	down	ARPA	1514	1000000						
Gi0/0/1/12	down	down	ARPA	1514	1000000						
Gi0/0/1/13	3 up	up	ARPA	9216	1000000						
Gi0/0/1/14	down	down	ARPA	1514	1000000						

Gi0/0/1/15	up	up	ARPA	9216	1000000						
Gi0/0/1/16	down	down	ARPA	1514	1000000						
Gi0/0/1/17	down	down	ARPA	1514	1000000						
Gi0/0/1/18	down	down	ARPA	1514	1000000						
Gi0/0/1/19	down	down	ARPA	1514	1000000						
Gi0/0/1/24	down	down	ARPA	1514	1000000/*	slice	5 port	numbers	starts	here	*/
Gi0/0/1/25	down	down	ARPA	1514	1000000						
Gi0/0/1/26	down	down	ARPA	1514	1000000						
Gi0/0/1/27	down	down	ARPA	1514	1000000						
Gi0/0/1/28	down	down	ARPA	1514	1000000						
Gi0/0/1/29	down	down	ARPA	1514	1000000						
Gi0/0/1/30	down	down	ARPA	1514	1000000						
Gi0/0/1/31	down	down	ARPA	1514	1000000						
Gi0/0/1/32	down	down	ARPA	1514	1000000						
Gi0/0/1/33	down	down	ARPA	1514	1000000						
Gi0/0/1/34	down	down	ARPA	1514	1000000						
Gi0/0/1/35	down	down	ARPA	1514	1000000						
Gi0/0/1/36	down	down	ARPA	1514	1000000						
Gi0/0/1/37	down	down	ARPA	1514	1000000						
Gi0/0/1/38	down	down	ARPA	1514	1000000						
Gi0/0/1/39	down	down	ARPA	1514	1000000						
Gi0/0/1/40	down	down	ARPA	1514	1000000						
Gi0/0/1/41	down	down	ARPA	1514	1000000						
Gi0/0/1/42	down	down	ARPA	1514	1000000						
Gi0/0/1/43	down	down	ARPA	1514	1000000						
Te0/0/1/20	down	down	ARPA	1514	10000000						
Te0/0/1/21	down	down	ARPA	1514	10000000						
Te0/0/1/22	down	down	ARPA	1514	10000000						
Te0/0/1/23	down	down	ARPA	1514	10000000						
Te0/0/1/44	down	down	ARPA	1514	10000000						
Te0/0/1/45	down	down	ARPA	1514	10000000						
Te0/0/1/46	down	down	ARPA	1514	10000000						
Te0/0/1/47	down	down	ARPA	1514	10000000						

Configuring Port Mode in Cisco ASR 9902 Router

The router has 48 ports with maximum of 800G data bandwidth capacity. You can configure the 48 ports in various port modes (100GbE, 25GbE, 10GbE, 40GbE and 1GbE) using the **hw-module location <node**> slice <number> config-mode command.

The port mode configuration is per slice and not per port. Also, the 48 ports on the Cisco ASR 9902 router are available in slice 0 and 1. Each slice can be configured for a mix of 1GbE, 10GbE, 25GbE, 40GbE or 100GbE port modes. The default configuration is 1x100GE, 1x100GE, 10x10GE, and 10x10GE.

For more information on the ports, see the Installing Modules and Cables in the Chassis chapter in the *Cisco* ASR 9000 Series Fixed-Port Routers Hardware Installation Guide.

The number of ports available on the router depends on the port mode configured on each slice. This table shows the configuration options and supported port modes on Cisco ASR 9902 router:

Configuration Options	Supported Port Modes on Slice 0 and Slice 1
Default	1x100GE, 1x100GE, 10x10GE, and 10x10GE
Option 1	1x100GE/1X40G/4X10GE, 1x100GE/1X40GE/4X10GE, 10x10GE or 5x1GE_5x10GE or 10x1GE, and 10x10GE or 5x1GE_5x10GE or 10x1GE

Table 39: Configuration Options, and Supported Port Modes on Cisco ASR 9902 Router

Configuration Options	Supported Port Modes on Slice 0 and Slice 1
Option 2	1x100GE/1X40GE/4X10GE, 1x100GE/1X40G/4X10GE, 4x25GE, and 10x10GE or 5x1GE_5x10GE
Option 3	1x100GE/1X40GE/4X10GE, 4x25GE, 4x25GE, and 1x100GE/1x40GE/4x10GE
Option 4	1x100GE/1X40GE/4X10GE, 1x100GE/1X40GE/4X10GE, 1x100GE/1X40GE, and 1x100GE/1X40GE/4X10GE

This sample shows how to configure Configuration Option 1:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#hw-module location 0/0/CPU0 slice 0 config-mode
lx100GE,1x100GE,5x1GE_5x10GE,5x1GE_5x10GE
RP/0/RP0/CPU0:ios(config)#commit
```

This sample shows how to configure Configuration Option 1 with 10x1GE optics speed:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#hw-module location 0/0/CPU0 slice 0 config-mode
1x100GE,1x100GE,10x1GE,10x1GE
RP/0/RP0/CPU0:ios(config)#commit
```

To revert to the default mode, use **no** form of the **hw-module location <node> slice <number> config-mode** command.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#no hw-module location 0/0/CPU0 slice 0 config-mode
1x100GE,1x100GE,5x1GE_5x10GE,5x1GE_5x10GE
RP/0/RP0/CPU0:ios(config)#commit
```

Use the **show interface brief** command to verify the ports configured using **config-mode** command.



```
Note
```

The 40G port mode on the Cisco ASR 9902 router works only with Cisco IOS XR software version 7.8.x or higher. So, if you want to use the 40G port mode on the Cisco ASR 9902 router, you should update the Cisco IOS XR software version to 7.8.x or higher.

Configuring Port Modes in Cisco ASR 9903 Router and Cisco ASR 9000 Series Line Cards

Table 40: Feature History Table

Feature Name	Release Information	Description
1GbE Port Mode	Release 7.7.1	You can now use the 1GbE port mode option to configure 10GbE optics speed on the SFP+ ports. This allows you to enable SFP+ ports on the line cards or router to deliver 1GbE standard traffic by using the hw-module location breakout command.
		This option is enabled on the following Cisco ASR 9000 Series router and line cards:
		• ASR 9903
		• A9K-4HG-FLEX-SE/TR
		• A99-4HG-FLEX-SE/TR
		• A9K-4HG-FLEX-FC
		• A99-4HG-FLEX-FC

The Cisco ASR 9903 router has 16 QSFP28 ports and 20 integrated SFP+ ports with a maximum of 1.6T data bandwidth capacity. You can configure the ports in various port modes (100GbE, 25GbE, 10GbE, and 1GbE) using the **hw-module location** *<node id>* **bay** *<bay number>* **port** *port number>* **breakout** *<interface>* command.

For more information on the ports, see the Installing Modules and Cables in the Chassis chapter in the *Cisco* ASR 9000 Series Fixed-Port Routers Hardware Installation Guide.

This table shows the configuration options and supported port modes on Cisco ASR 9903 router and Cisco ASR 9000 Series line cards.

Configuration Options	Supported Port Modes	
Default	1x100GE, 1x100GE, 10x10GE, and 10x10GE	
Option 1	1x100GE, 1x100GE, 10x10GE or 5x1GE_5x10GE or 10x1GE, and 10x10GE or 5x1GE_5x10GE or 10x1GE	
Option 2	1x100GE, 1x100GE, 4x25GE, and 10x10GE or 5x1GE_5x10GE or 10x1GE	
Option 3	1x100GE, 4x25GE, 4x25GE, and 1x100GE	
Option 4	1x100GE, 1x100GE, 1x100GE, and 1x100GE	

Table 41: Confi	ouration O	ptions and	Supported	Port Modes
14010 111 001111	garadion o	paone ana	oupportou	1 011 1110400

Configuring Supported Port Modes on Routers

To configure 1GbE port mode on Cisco ASR 9903 router, use the QSFP28 ports 10 and 14. You can configure the ports for a mix of 1GbE, 10GbE, 25GbE, or 100GbE port modes.

This sample shows how to configure port mode of 1GbE on ports 10 with 5x1GE-5x10GE option for Cisco ASR 9903 router:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#hw-module location 0/0/CPU0 bay 0 port 10 breakout
5x1GE-5x10GE
RP/0/RP0/CPU0:ios(config)#commit
```

This sample shows how to configure port mode of 1GbE on ports 10 with 10x1GE option for Cisco ASR 9903 router:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#hw-module location 0/0/CPU0 bay 0 port 10 breakout 10x1GE
RP/0/RP0/CPU0:ios(config)#commit
```

Configuring Supported Port Modes on Line Cards

To configure 1GbE port mode on the Cisco ASR 9000 Series line cards, use the **hw-module location** *<node>* **slice** *<number>* **config-mode** command. On these cards, the port mode configuration is done per slice and not per port.

For more information on the ports, see the Overview chapter in the Cisco ASR 9000 Series Aggregation Services Router Ethernet Line Card Installation Guide.

This sample shows how to configure Option 1 on slice 0 with 1x100GE, 1x100GE, 5x1GE_5x10GE, and 5x1GE_5x10GE port modes on the line cards:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#hw-module location 0/0/CPU0 slice 0 config-mode
1x100GE,1x100GE,5x1GE_5x10GE,5x1GE_5x10GE
RP/0/RP0/CPU0:ios(config)#commit
```

This sample shows how to configure Option 1 on slice 0 with 1x100GE, 1x100GE, 10x10GE, and 10x1GE port modes on the line cards:

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#hw-module location 0/0/CPU0 slice 0 config-mode
1x100GE,1x100GE,10x1GE,10x1GE
RP/0/RP0/CPU0:ios(config)#commit
```

To revert to the default mode, use the **no** form of the **hw-module location** *<node>* **slice** *<number>* config-mode command.

```
RP/0/RP0/CPU0:ios#configure
RP/0/RP0/CPU0:ios(config)#no hw-module location 0/0/CPU0 slice 0 config-mode
1x100GE,1x100GE,5x1GE_5x10GE,5x1GE_5x10GE
RP/0/RP0/CPU0:ios(config)#commit
```

Configure Single Feed Power Mode

Cisco ASR 9000 series router supports the operating of one or all power modules. For example, V1 DC, V2 DC, V3 AC and V3 DC.

Ideally, you're expected to connect all the power modules (or feed) to power supply. If you don't connect any one feed, the system raises an alarm or error message.

You can configure the single-feed power mode to suppress the error message or an alarm for any missing feeds.

Configuration Example

The following example enables the single power feed mode for the 0/PS2/M0/SP power module:

```
Router#admin
Router(admin)#config
Router(admin-config)#power single-feed location 0/PS2/M0/SP
```

Additional References

The following sections provide references related to hardware management on Cisco IOS XR software.

Related Topic	Document Title
Cisco IOS XR hardware commands	Hardware Redundancy and Node Administration Commands on the Cisco ASR 9000 Series Router module of System Management Command Reference for Cisco ASR 9000 Series Routers
Cisco IOS XR hardware documentation	See Cisco Carrier Routing System Install and Upgrade Guides at: http://www.cisco.com/en/US/products/ ps5763/prod_installation_guides_list.html
Information about getting started with Cisco IOS XR software	Cisco ASR 9000 Series Aggregation Services Router Getting Started Guide
ROM Monitor	<i>ROM Monitor Configuration Guide for Cisco ASR</i> 9000 <i>Routers</i>
Cisco IOS XR command master list	Cisco ASR 9000 Series Aggregation Services Router Commands Master List
Information about user groups and task IDs	Configuring AAA Services on the Cisco ASR 9000 Series Router module of System Security Configuration Guide for Cisco ASR 9000 Series Routers

Related Documents

Standards

litte
-

I

MIBs

MBs	MIBs Link
	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

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

Technical Assistance

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


A

admin configure command **458–459**, **470** admin-config submode, *See* admin configure command

C

config-register command 458–459 controllers command 460 CPU controller bits 470

D

drives, formatting 460

F

format command **460** formatting drives **460**

Η

hardware **432** displaying status **432**

I

interface preconfigure command 461–462

L

```
line card 461–463
removal 461
replacement 462–463
```

Ν

```
node 457
power cycle 457
reload 457
shutdown 457
```

0

```
OIR 461
```

line cards, MSCs and PLIMs **461** online insertion and removal, *See* OIR

Ρ

PLIM 462–463 replacement 462–463 different media type 463 same media type and port count 462 same media type, different port count 462 power cycle 457

R

redundancy 449–451 commands 450 manual switchover 451 primary RP 449 standby RP 450 redundancy switchover command 451 reload command 451, 458 reloading software 457 RP reload 451

S

show controllers command 460 show diag command 432, 435 show environment command 442 show interfaces command 461 show platform command 441 administration EXEC mode 441 show redundancy command 446, 458 show running-config command 461 show variables boot command 458 show version command 439 shutting down a node 457

U

upgrade cpuctribits command 470

INDEX