MPLS Layer 2 VPNs Configuration Guide, Cisco IOS Release 12.2SR

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

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.

CONTENTS

Any Transport over MPLS 1

- Finding Feature Information 1
- Prerequisites for Any Transport over MPLS 2
- Restrictions for Any Transport over MPLS 3
- Information About Any Transport over MPLS 4
  - How AToM Transports Layer 2 Packets 5
  - AToM Configuration Commands Prior to Cisco IOS Release 12.0(25)S 5
  - Benefits of AToM 6
  - MPLS Traffic Engineering Fast Reroute 6
  - Maximum Transmission Unit Guidelines for Estimating Packet Size 7
    - Example Estimating Packet Size 8
    - mpls mtu Command Changes 9
  - Frame Relay over MPLS and DTE DCE and NNI Connections 9
    - Local Management Interface and Frame Relay over MPLS 10
      - How LMI Works 10
  - QoS Features Supported with AToM 11
- How to Configure Any Transport over MPLS 14
  - Configuring the Pseudowire Class 15
  - Configuring ATM AAL5 over MPLS on PVCs 16
  - Configuring ATM AAL5 over MPLS in VC Class Configuration Mode 18
  - Configuring OAM Cell Emulation for ATM AAL5 over MPLS 21
    - Configuring OAM Cell Emulation for ATM AAL5 over MPLS on PVCs 22
    - Configuring OAM Cell Emulation for ATM AAL5 over MPLS in VC Class Configuration Mode 25
  - Configuring ATM Cell Relay over MPLS in VC Mode 27
  - Configuring ATM Cell Relay over MPLS in VC Mode Using VC Class Configuration Mode 29
  - Configuring ATM Cell Relay over MPLS in PVP Mode 31
  - Configuring ATM Cell Relay over MPLS in Port Mode 34
    - Troubleshooting Tips 36

Configuring ATM Single Cell Relay over MPLS 36

Configuring ATM Packed Cell Relay over MPLS 38

Restrictions 38

Configuring ATM Packed Cell Relay over MPLS in VC Mode 38

Configuring ATM Packed Cell Relay over MPLS in VC Mode Using VC Class Configuration Mode 41

Configuring ATM Packed Cell Relay over MPLS in VP Mode 45

Configuring ATM Packed Cell Relay over MPLS in Port Mode 47

Troubleshooting Tips 51

Configuring Ethernet over MPLS in VLAN Mode 51

Configuring Ethernet over MPLS in Port Mode 52

Configuring Ethernet over MPLS with VLAN ID Rewrite 54

Configuring Ethernet over MPLS with VLAN ID Rewrite for Cisco 12k Routers for 12.0(29)S and Earlier Releases 54

Configuring Ethernet over MPLS with VLAN ID Rewrite for Cisco 12k Routers for 12.0(30)S and Later Releases 55

Configuring per-Subinterface MTU for Ethernet over MPLS 58

Configuring Frame Relay over MPLS with DLCI-to-DLCI Connections 61

Configuring Frame Relay over MPLS with Port-to-Port Connections 63

Configuring HDLC and PPP over MPLS 64

Configuring Tunnel Selection 66

Troubleshooting Tips 69

Setting Experimental Bits with AToM 70

Setting the Frame Relay Discard Eligibility Bit on the Cisco 7200 and 7500 Series Routers 75

Matching the Frame Relay DE Bit on the Cisco 7200 and 7500 Series Routers 77

Enabling the Control Word 78

Configuration Examples for Any Transport over MPLS 79

Example ATM AAL5 over MPLS 80

Example OAM Cell Emulation for ATM AAL5 over MPLS 80

Example ATM Cell Relay over MPLS 81

Example ATM Single Cell Relay over MPLS 82

Example Ethernet over MPLS 84

Example Tunnel Selection 84

Example Setting Frame Relay Discard Eligibility Bit on the Cisco 7200 and 7500 Series Routers 86

Example Matching Frame Relay DE Bit on the Cisco 7200 and 7500 Series Routers 87

Example ATM over MPLS 87

Example Ethernet over MPLS with MPLS Traffic Engineering Fast Reroute 88

Example Configuring per-Subinterface MTU for Ethernet over MPLS 91

Example Configuring MTU Values in xconnect Configuration Mode for L2VPN Interworking 93

Example Removing a Pseudowire 95

Additional References 97

Feature Information for Any Transport over MPLS 99

MPLS MTU Command Changes 109

Finding Feature Information 109

Information About MPLS MTU Command Changes 110

MPLS MTU Values During Upgrade 110

Guidelines for Setting MPLS MTU and Interface MTU Values 110

MPLS MTU Values for Ethernet Interfaces 111

How to Configure MPLS MTU Values 111

Setting the Interface MTU and MPLS MTU Values 112

Setting the MPLS MTU Value on an Ethernet Interface 113

Setting the MPLS MTU Value to the Maximum on L3VPN Profiles 114

Configuration Examples for Setting the MPLS MTU Values 115

Example Setting the Interface MTU and MPLS MTU 115

Example Setting the MPLS MTU Value on an Ethernet Interface 116

Example Setting the MPLS MTU Value to the Maximum MTU on L3VPN profiles 117

Additional References 117

Feature Information for MPLS MTU Command Changes 118

AToM Static Pseudowire Provisioning 121

Finding Feature Information 121

Restrictions for AToM Static Pseudowire Provisioning 121

Information About AToM Static Pseudowire Provisioning 122

Pseudowire Provisioning 122

Benefits of Statically Provisioned Pseudowires 122

How to Provision an AToM Static Pseudowire 122

Provisioning an AToM Static Pseudowire 122

Verifying the AToM Static Pseudowire Configuration 124

Configuration Examples for AToM Static Pseudowire Provisioning 125

Provisioning an AToM Pseudowire Example 125

Additional References 126

Feature Information for AToM Static Pseudowire Provisioning 127

L2VPN Interworking 129

Finding Feature Information 129

Prerequisites for L2VPN Interworking 129

Restrictions for L2VPN Interworking 130

General Restrictions 130

Cisco 7600 Series Routers Restrictions 130

Cisco 12000 Series Router Restrictions 132

ATM AAL5 Interworking Restrictions 135

Ethernet VLAN Interworking Restrictions 135

Restrictions 136

Frame Relay Interworking Restrictions 138

PPP Interworking Restrictions 138

Information About L2VPN Interworking 139

Overview of L2VPN Interworking 139

L2VPN Interworking Modes 139

Ethernet (Bridged) Interworking 140

IP (Routed) Interworking 140

VLAN Interworking 141

L2VPN Interworking Support Matrix 141

Static IP Addresses for L2VPN Interworking for PPP 142

How to Configure L2VPN Interworking 142

Configuring L2VPN Interworking 142

Verifying the L2VPN Interworking Configuration 143

Configuring L2VPN Interworking: VLAN Enable-Disable Option for AToM 147

Configuration Examples for L2VPN Interworking 149

Ethernet to VLAN over L2TPV3 (Bridged) Example 150

Ethernet to VLAN over AToM (Bridged) Example 151

Frame Relay to VLAN over L2TPV3 (Routed) Example 152

Frame Relay to VLAN over AToM (Routed) Example 154

Frame Relay to ATM AAL5 over AToM (Routed) Example 155

VLAN to ATM AAL5 over AToM (Bridged) Example 156

Frame Relay to PPP over L2TPv3 (Routed) Example 157

Frame Relay to PPP over AToM (Routed) Example 159

Ethernet VLAN to PPP over AToM (Routed) Example 161

Additional References 163

Feature Information for L2VPN Interworking 164

L2VPN Pseudowire Switching 167

Finding Feature Information 167

Prerequisites for L2VPN Pseudowire Switching 167

Restrictions for L2VPN Pseudowire Switching 168

Information About L2VPN Pseudowire Switching 168

How L2VPN Pseudowire Switching Works 168

How Packets Are Manipulated at the L2VPN Pseudowire Switching Aggregation Point 169

How to Configure L2VPN Pseudowire Switching 169

Examples 171

Configuration Examples for L2VPN Pseudowire Switching 172

L2VPN Pseudowire Switching in an Inter-AS Configuration Example 172

Additional References 177

Feature Information for L2VPN Pseudowire Switching 178

L2VPN Pseudowire Redundancy 181

Finding Feature Information 181

Prerequisites for L2VPN Pseudowire Redundancy 181

Restrictions for L2VPN Pseudowire Redundancy 182

Information About L2VPN Pseudowire Redundancy 182

Introduction to L2VPN Pseudowire Redundancy 182

How to Configure L2VPN Pseudowire Redundancy 184

Configuring the Pseudowire 184

Configuring L2VPN Pseudowire Redundancy 185

Forcing a Manual Switchover to the Backup Pseudowire VC 187

Verifying the L2VPN Pseudowire Redundancy Configuration 188

Configuration Examples for L2VPN Pseudowire Redundancy 189

L2VPN Pseudowire Redundancy and AToM Like to Like Examples 190

L2VPN Pseudowire Redundancy and L2VPN Interworking Examples 190

L2VPN Pseudowire Redundancy with Layer 2 Local Switching Examples 191

Additional References 191

Feature Information for L2VPN Pseudowire Redundancy 192

VPLS Autodiscovery BGP Based 195

Feature Information For 195

Prerequisites for VPLS Autodiscovery BGP Based 195

Restrictions for VPLS Autodiscovery BGP Based 195

Information About VPLS Autodiscovery BGP Based 196

How the VPLS Feature Works 197

How the VPLS Autodiscovery BGP Based Feature Works 197

How Enabling VPLS Autodiscovery Differs from Manually Configuring VPLS 197

Show Commands Affected by VPLS Autodiscovery BGP Based 198

BGP VPLS Autodiscovery Support on a Route Reflector 198

How to Configure VPLS Autodiscovery BGP Based 199

Enabling VPLS Autodiscovery BGP Based 199

Configuring BGP to Enable VPLS Autodiscovery 200

Customizing the VPLS Autodiscovery Settings 204

Configuration Examples for VPLS Autodiscovery BGP Based 206

VPLS Autodiscovery BGP Based Basic Example 206

BGP VPLS Autodiscovery Support on Route Reflector Example 208

Additional References 208

Feature Information for VPLS Autodiscovery BGP Based 210

H-VPLS N-PE Redundancy for QinQ and MPLS Access 213

Finding Feature Information 213

Prerequisites for H-VPLS N-PE Redundancy for QinQ and MPLS Access 213

Restrictions for H-VPLS N-PE Redundancy for QinQ and MPLS Access 214

Information About H-VPLS N-PE Redundancy for QinQ and MPLS Access 215

How H-VPLS N-PE Redundancy for QinQ and MPLS Access Works 215

H-VPLS N-PE Redundancy with QinQ Access Based on MSTP 215

H-VPLS N-PE Redundancy with MPLS Access Based on Pseudowire Redundancy 216

VPLS MAC Address Withdrawal 216

How MAC Address Withdrawal Works with H-VPLS N-PE Redundancy with QinQ Access 217

How MAC Address Withdrawal Works with H-VPLS N-PE Redundancy with MPLS Access 217

How to Configure H-VPLS N-PE Redundancy for QinQ and MPLS Access 217

Configuring the VPLS Pseudowire Between the N-PE Routers 217

Configuring the SVI for the Native VLAN 219

Configuration Examples for H-VPLS N-PE Redundancy for QinQ and MPLS Access 220

Example H-VPLS N-PE Redundancy for QinQ Access 221

Additional References 221

Feature Information for H-VPLS N-PE Redundancy for QinQ and MPLS Access 223
Glossary 225
L2VPN Multisegment Pseudowires 227
Finding Feature Information 227
Prerequisites for L2VPN Multisegment Pseudowires 227
Restrictions for L2VPN Multisegment Pseudowires 228
Information About L2VPN Multisegment Pseudowires 228
L2VPN Pseudowire Defined 228
L2VPN Multisegment Pseudowire Defined 229
MPLS OAM Support for Multisegment Pseudowires 229
MPLS OAM Support for L2VPN VPLS Inter-AS Option B 230
How to Configure L2VPN Multisegment Pseudowires 230
Configuring L2VPN Multisegment Pseudowires 230
Cisco 7600 Router-Specific Instructions 230
Displaying Information About the L2VPN Multisegment Pseudowires 233
Verifying Multisegment Pseudowires with ping mpls and trace mpls Commands 234
Verifying L2VPN VPLS Inter-AS Option B with ping mpls and trace mpls Commands 236
Configuration Examples for L2VPN Multisegment Pseudowires 238
Example Configuring an L2VPN Multisegment Pseudowire 238
Additional References 241
Feature Information for L2VPN Multisegment Pseudowires 243
QoS Policy Support for L2VPN ATM PVPs 245
Finding Feature Information 245
Prerequisites for QoS Policy Support for L2VPN ATM PVPs 245
Restrictions for QoS Policy Support for L2VPN ATM PVPs 245
Information About QoS Policy Support for L2VPN ATM PVPs 246
MQC Structure 246
Elements of a Traffic Class 246
Elements of a Traffic Policy 246
How to Configure QoS Policy Support for L2VPN ATM PVPs 247
Enabling a Service Policy in ATM PVP Mode 247
Enabling Traffic Shaping in ATM PVP Mode 248
Enabling Matching of ATM VCIs 251
Configuration Examples for QoS Policy Support for L2VPN ATM PVPs 252
Enabling Traffic Shaping in ATM PVP Mode Example 252

Additional References 252

Feature Information for QoS Policy Support for L2VPN ATM PVPs 254

L2VPN Pseudowire Preferential Forwarding 255

Finding Feature Information 255

Prerequisites for L2VPN--Pseudowire Preferential Forwarding 255

Restrictions for L2VPN--Pseudowire Preferential Forwarding 256

Information About L2VPN--Pseudowire Preferential Forwarding 256

Overview of L2VPN--Pseudowire Preferential Forwarding 256

How to Configure L2VPN--Pseudowire Preferential Forwarding 257

Configuring the Pseudowire Connection Between PE Routers 257

Configuration Examples for L2VPN--Pseudowire Preferential Forwarding 258

L2VPN--Pseudowire Preferential Forwarding Configuration Example 259

Displaying the Status of the Pseudowires Example 259

Additional References 260

Feature Information for L2VPN: Pseudowire Preferential Forwarding 262

MPLS Pseudowire Status Signaling 263

Finding Feature Information 263

Prerequisites for MPLS Pseudowire Status Signaling 263

Restrictions for MPLS Pseudowire Status Signaling 263

Information About MPLS Pseudowire Status Signaling 264

How MPLS Pseudowire Status Signaling Works 264

When One Router Does Not Support MPLS Pseudowire Status Signaling 264

Status Messages Indicating That the Attachment Circuit Is Down 265

Message Codes in the Pseudowire Status Messages 265

How to Configure MPLS Pseudowire Status Signaling 266

Enabling MPLS Pseudowire Status Signaling 266

Configuration Examples for MPLS Pseudowire Status Signaling 268

MPLS Pseudowire Status Signaling Example 268

Verifying That Both Routers Support Pseudowire Status Messages Example 269

Additional References 269

Any Transport over MPLS

This document describes the Any Transport over MPLS (AToM) feature, which provides the following capabilities:

- Transport data link layer (Layer2) packets over a Multiprotocol Label Switching (MPLS) backbone.
- Enable service providers to connect customer sites with existing Layer 2 networks by using a single, integrated, packet-based network infrastructure--a Cisco MPLS network. Instead of using separate networks with network management environments, service providers can deliver Layer 2 connections over an MPLS backbone.
- Provide a common framework to encapsulate and transport supported Layer 2 traffic types over an MPLS network core.

AToM supports the following like-to-like transport types:

- ATM Adaptation Layer Type-5 (AAL5) over MPLS
- ATM Cell Relay over MPLS
- Ethernet over MPLS (VLAN and port modes)
- Frame Relay over MPLS
- PPP over MPLS
- High-Level Data Link Control (HDLC) over MPLS

- Finding Feature Information, page 1
- Prerequisites for Any Transport over MPLS, page 2
- Restrictions for Any Transport over MPLS, page 3
- Information About Any Transport over MPLS, page 4
- How to Configure Any Transport over MPLS, page 14
- Configuration Examples for Any Transport over MPLS, page 79
- Additional References, page 97
- Feature Information for Any Transport over MPLS, page 99

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Any Transport over MPLS

Before configuring AToM, ensure that the network is configured as follows:

- Configure IP routing in the core so that the provider edge (PE) routers can reach each other via IP.
- Configure MPLS in the core so that a label-switched path (LSP) exists between the PE routers.
- Enable Cisco Express Forwarding or distributed Cisco Express Forwarding before configuring any Layer 2 circuits.
- Configure a loopback interface for originating and terminating Layer 2 traffic. Make sure the PE routers can access the other router’s loopback interface. Note that the loopback interface is not needed in all cases. For example, tunnel selection does not need a loopback interface when AToM is directly mapped to a traffic engineering (TE) tunnel.
- AToM is supported on the Cisco 7200 and 7500 series routers. For details on supported hardware, see the following documents:
    - Cross-Platform Release Notes for Cisco IOS Release 12.0S
    - Cross-Platform Release Notes for Cisco IOS Release 12.4T, Part 2: Platform-Specific Information
- AToM is supported on the Cisco 7600 routers. For details on supported shared port adapters and line cards, see the following documents:
    - Guide to Supported Hardware for Cisco 7600 Series Routers with Release 12.2SR
    - Cross-Platform Release Notes for Cisco IOS Release 12.2SR for the Cisco 7600 Series Routers
- The Cisco 7600 router has platform-specific instructions for configuring some AToM features. Platform-specific configuration information is included in the following documents:
    - The “Configuring PFC3BXL and PFC3B Mode Multiprotocol Label Switching” module of the Cisco 7600 Series Cisco IOS Software Configuration Guide, Release 12.2SR
    - The “Configuring Multiprotocol Label Switching on the Optical Services Modules” module of the OSM Configuration Note , Release 12.2SR
    - The “Configuring Multiprotocol Label Switching on FlexWAN and Enhanced FlexWAN Modules” module of the FlexWAN and Enhanced FlexWAN Modules Installation and Configuration Guides of Cisco 7600 Series Routers
    - The “Configuring Any Transport over MPLS on a SIP” section of the Cisco 7600 Series Router SIP, SSC, and SPA Software Configuration Guide
    - The “Configuring AToM VP Cell Mode Relay Support” section of the Cisco 7600 Series Router SIP, SSC, and SPA Software Configuration Guide
    - The Cross-Platform Release Notes for Cisco IOS Release 12.2SR
- AToM is supported on the Cisco 10000 series routers. For details on supported hardware, see the “Configuring Any Transport over MPLS” section of the Cisco 10000 Series Router Software Configuration Guide.
- The Cisco 10000 series router has platform-specific instructions for configuring some AToM features. Platform-specific configuration information is contained in the “Configuring Any Transport over MPLS” section of the Cisco 10000 Series Router Broadband Aggregation, Leased-Line, and MPLS Configuration Guide.
- AToM is supported on the Cisco12000 series routers. For information about hardware requirements, see the Cross-Platform Release Notes for Cisco IOS Release 12.0S.

Restrictions for Any Transport over MPLS

General Restrictions

The following general restrictions pertain to all transport types under AToM:

- Address format: Configure the Label Distribution Protocol (LDP) router ID on all PE routers to be a loopback address with a /32 mask. Otherwise, some configurations might not function properly.
- Layer 2 virtual private networks (L2VPN) features (AToM and Layer 2 Tunnel Protocol Version 3 (L2TPv3)) are not supported on an ATM interface.
- Distributed Cisco Express Forwarding is the only forwarding model supported on the Cisco 12000 series routers and is enabled by default. Disabling distributed Cisco Express Forwarding on the Cisco 12000 series routers disables forwarding.
- Distributed Cisco Express Forwarding mode is supported on the Cisco 7500 series routers for Frame Relay, HDLC, and PPP. In distributed Cisco Express Forwarding mode, the switching process occurs on the Versatile Interface Processors (VIPs) that support switching. When distributed Cisco Express Forwarding is enabled, VIP port adapters maintain identical copies of the Forwarding Information Base (FIB) and adjacency tables. The port adapters perform the express forwarding between port adapters, relieving the Route Switch Processor (RSP) from performing the switching. Distributed Cisco Express Forwarding uses an interprocess communications (IPC) mechanism to ensure synchronization of FIBs and adjacency tables between the RSP and port adapters.
- To convert an interface with L2TPv3 xconnect to AToM xconnect, remove the L2TPv3 configuration from the interface and then configure AToM. Some features may not work if AToM is configured when L2TPv3 configuration is not removed properly.

ATM Cell Relay over MPLS Restrictions

The following restrictions pertain to ATM Cell Relay over MPLS:

- For ATM Cell Relay over MPLS,if you have TE tunnels running between the PE routers, you must enable LDP on the tunnel interfaces.
- Configuring ATM Relay over MPLS with the Cisco 12000 Series Router engine 2 8-port OC-3 STM-1 ATM line card: In Cisco IOS Release 12.0(25)S, there were special instructions for configuring ATM cell relay on the Cisco 12000 series router with an engine 2 8-port OC-3 STM-1 ATM line card. The special configuration instructions do not apply to releases later than Cisco IOS Release 12.0(25)S and you do not need to use the **atm mode cell-relay** command.

In Cisco IOS Release 12.0(25)S, when you configured the Cisco 12000 series 8-port OC-3 STM-1 ATM line card for ATM Cell Relay over MPLS, two ports were reserved. In releases later than Cisco IOS Release 12.0(25)S, only one port is reserved.

In addition, in Cisco IOS Release 12.0(25)S, if you configured an 8-port OC-3 STM-1 ATM port for ATM Adaptation Layer 5 (AAL5) over MPLS and then configured ATM single cell relay over MPLS on that port, the Virtual Circuits (VCs) and Virtual Paths (VPs) for AAL5 on the port and its corresponding port were removed. Starting in Cisco IOS Release 12.0(26)S, this behavior no longer occurs. ATM AAL5 over MPLS and ATM single cell relay over MPLS are supported on the same port. The Cisco 12000 series 8-port OC-3 STM-1 ATM line cards now support, by default, the ATM single cell relay over MPLS feature in both VP and VC modes and ATM AAL5 over MPLS on the same port.

- The F4 end-to-end Operation, Administration, and Maintenance (OAM) cells are transparently transported along with the ATM cells. When a permanent virtual path (PVP) or Permanent Virtual Circuit (PVC) is down on one PE router, the label associated with that PVP or PVC is withdrawn.

Subsequently, the peer PE router detects the label withdrawal and sends an F4 AIS/RDI signal to its corresponding customer edge (CE) router. The PVP or PVC on the peer PE router remains in the up state.

Ethernet over MPLS (EoMPLS) Restrictions

The following restrictions pertain to the Ethernet over MPLS feature:

- Ethernet over MPLS supports VLAN packets that conform to the IEEE 802.1Q standard. The 802.1Q specification establishes a standard method for inserting VLAN membership information into Ethernet frames. The Inter-Switch Link (ISL) protocol is not supported between the PE and CE routers.
- The AToM control word is supported. However, if the peer PE does not support a control word, the control word is disabled. This negotiation is done by LDP label binding.
- Ethernet packets with hardware-level cyclic redundancy check (CRC) errors, framing errors, and runt packets are discarded on input.
- In Cisco IOS Release 12.2(25)S, the behavior of the **mpls mtu** command changed. If the interface MTU is less than 1524 bytes, you can set the maximum MPLS MTU to 24 bytes more than the interface MTU. For example, if the interface MTU is set to 1510 bytes, then you can set the maximum MPLS MTU to 1534 bytes (1510 + 24).

Caution Although you can set the MPLS MTU to a value greater than the interface MTU, you must set the MPLS MTU to a value less than or equal to the interface MTU to prevent data corruption, dropped packets, and high CPU rates.

If the interface MTU is greater than or equal to 1524 bytes, then you can set the maximum MPLS MTU as high as the interface MTU. For example, if the interface MTU is set to 1600 bytes, then you can set the MPLS MTU to a maximum of 1600 bytes. If you set the MPLS MTU to a value higher than the interface MTU, traffic is dropped.

For interfaces that do not allow you to configure the interface MTU value and for interfaces where the interface MTU is 1500 bytes, the MPLS MTU range is 64 to 1524 bytes.

If you upgrade to Cisco IOS Release 12.2(25)S from an earlier release and you have an MPLS MTU setting that does not conform to these guidelines, the command is rejected. See the Maximum Transmission Unit Guidelines for Estimating Packet Size, page 7 for more information.

Frame Relay over MPLS Restrictions

The following restrictions pertain to the Frame Relay over MPLS feature:

- Frame Relay traffic shaping is not supported with AToM switched VCs.
- If you configure Frame Relay over MPLS on the Cisco 12000 series router and the core-facing interface is an engine 4 or 4+ line card and the edge-facing interface is an engine 0 or 2 line card, then the BECN, FECN, control word (CW), and DE bit information is stripped from the PVC.

Information About Any Transport over MPLS

- How AToM Transports Layer 2 Packets, page 5
- AToM Configuration Commands Prior to Cisco IOS Release 12.0(25)S, page 5
- Benefits of AToM, page 6
- MPLS Traffic Engineering Fast Reroute, page 6

- Maximum Transmission Unit Guidelines for Estimating Packet Size, page 7
- Frame Relay over MPLS and DTE DCE and NNI Connections, page 9
- QoS Features Supported with AToM, page 11

How AToM Transports Layer 2 Packets

AToM encapsulates Layer 2 frames at the ingress PE and sends them to a corresponding PE at the other end of a pseudowire, which is a connection between the two PE routers. The egress PE removes the encapsulation and sends out the Layer 2 frame.

The successful transmission of the Layer 2 frames between PE routers is due to the configuration of the PE routers. You can set up the connection, called a pseudowire, between the routers and specify the following information on each PE router:

- The type of Layer 2 data that will be transported across the pseudowire such as Ethernet, Frame Relay, or ATM
- The IP address of the loopback interface of the peer PE router, which enables the PE routers to communicate
- A unique combination of peer PE IP address and VC ID that identifies the pseudowire

The following example shows the basic configuration steps on a PE router that enable the transport of Layer 2 packets. Each transport type has slightly different steps.

Step 1 defines the interface or subinterface on the PE router:

```
Router# interface
interface-type interface-number
```

Step 2 specifies the encapsulation type for the interface, such as dot1q:

```
Router(config-if)# encapsulation encapsulation-type
```

Step 3 does the following:

- Makes a connection to the peer PE router by specifying the LDP router ID of the peer PE router.
- Specifies a 32-bit unique identifier, called the VC ID, which is shared between the two PE routers.

The combination of the peer router ID and the VC ID must be unique on the router. Two circuits cannot use the same combination of the peer router ID and VC ID.

- Specifies the tunneling method used to encapsulate data in the pseudowire. AToM uses MPLS as the tunneling method.

```
Router(config-if)# xconnect peer-router-id vcid encapsulation mpls
```

As an alternative, you can set up a pseudowire class to specify the tunneling method and other characteristics. For more information, see the Configuring the Pseudowire Class, page 15.

AToM Configuration Commands Prior to Cisco IOS Release 12.0(25)S

In releases of AToM before Cisco IOS 12.0(25)S, the **mpls l2 transport route**command was used to configure AToM circuits.This command has been replaced with the **xconnect**command.

No enhancements will be made to the **mpls l2transport route**command. Enhancements will be made to either the **xconnect**command or the **pseudowire-class**command. Therefore, Cisco recommends that you use the **xconnect** command to configure AToM circuits.

Configurations from releases before Cisco IOS 12.0(25)S that use the **mpls l2transport route**command are still supported.

Benefits of AToM

The following list explains some of the benefits of enabling Layer 2 packets to be sent in the MPLS network:

- The AToM product set accommodates many types of Layer 2 packets, including Ethernet and Frame Relay, across multiple Cisco router platforms, such as the Cisco 7200 and Cisco 7500 series routers. This enables the service provider to transport all types of traffic over the backbone and accommodate all types of customers.
- AToM adheres to the standards developed for transporting Layer 2 packets over MPLS. (See the "Standards" section for the specific standards that AToM follows.) This benefits the service provider that wants to incorporate industry-standard methodologies in the network. Other Layer 2 solutions are proprietary, which can limit the service provider's ability to expand the network and can force the service provider to use only one vendor's equipment.
- Upgrading to AToM is transparent to the customer. Because the service provider network is separate from the customer network, the service provider can upgrade to AToM without disruption of service to the customer. The customers assume that they are using a traditional Layer 2 backbone.

MPLS Traffic Engineering Fast Reroute

AToM can use MPLS traffic engineering (TE) tunnels with fast reroute (FRR) support. AToM VCs can be rerouted around a failed link or node at the same time as MPLS and IP prefixes.

Enabling fast reroute on AToM does not require any special commands; you can use the standard fast reroute (FRR) commands. At the ingress PE, an AToM tunnel is protected by fast reroute when it is routed to an FRR-protected TE tunnel. Both link and node protection are supported for AToM VCs at the ingress PE. For more information on configuring MPLS TE fast reroute, see the following document:

MPLS Traffic Engineering (TE)--Link and Node Protection, with RSVP Hellos Support

Note The AToM VC independence feature was introduced in Cisco IOS Release 12.0(31)S. This feature enables the Cisco 12000 series router to perform fast reroute in fewer than 50 milliseconds, regardless of the number of VCs configured. In previous releases, the fast reroute time depended on the number of VCs inside the protected TE tunnel.

For the Cisco 12000 series routers, fast reroute uses three or more labels, depending on where the TE tunnel ends:

- If the TE tunnel is from a PE router to a PE router, three labels are used.
- If the TE tunnel is from a PE router to the core router, four labels are used.

Engine 0 ATM line cards support three or more labels, but the performance degrades. Engine 2 Gigabit Ethernet line cards and engine 3 line cards support three or more labels and can work with the fast reroute feature.

You can issue the **debug mpls l2transport fast-reroute**command to debug fast reroute with AToM.

Note This command does not display output on platforms where AToM fast reroute is implemented in the forwarding code. The command does display output on Cisco 10720 Internet router line cards and Cisco 12000 series line cards. This command does not display output for the Cisco 7500 (both Route Processor (RP) and Versatile Interface Processor (VIP)) series routers, Cisco 7200 series routers, and Cisco 12000 series RP.

In the following example, the primary link is disabled, which causes the backup tunnel (Tunnel 1) to become the primary path. In the following example, bolded output shows the status of the tunnel:

```
Router# execute-on slot 3 debug mpls l2transport fast-reroute
========= Line Card (Slot 3) =========
AToM fast reroute debugging is on
SLOT 3:Sep 16 17:58:56.346: AToM SMGR: Processing TFIB FRR event for 10.4.0.1
SLOT 3:Sep 16 17:58:56.346: AToM SMGR: Finished processing TFIB FRR event for 10.4.0.1
SLOT 3:Sep 16 17:58:56.346: AToM SMGR: Processing TFIB FRR event for Tunnel41
SLOT 3:Sep 16 17:58:56.346: AToM SMGR: Finished processing TFIB FRR event for Tunnel41
Sep 16 17:58:58.342: %LINK-3-UPDOWN: Interface POS0/0, changed state to down
Sep 16 17:58:58.342: %OSPF-5-ADJCHG: Process 1, Nbr 10.0.0.1 on POS0/0 from FULL to DOWN,
Neighbor Down: Interface down or detached
Sep 16 17:58:59.342: %LINEPROTO-5-UPDOWN: Line protocol on Interface POS0/0, changed
state to down
```

Maximum Transmission Unit Guidelines for Estimating Packet Size

The following calculation helps you determine the size of the packets traveling through the core network. You set the maximum transmission unit (MTU) on the core-facing interfaces of the P and PE routers to accommodate packets of this size. The MTU should be greater than or equal to the total bytes of the items in the following equation:

Core MTU >= (Edge MTU + Transport header + AToM header + (MPLS label stack * MPLS label size))

The following sections describe the variables used in the equation:

Edge MTU

The edge MTU is the MTU for customer-facing interfaces.

Transport Header

The Transport header depends on the transport type. The table below lists the specific sizes of the headers.

Table 1 Header Size of Packets

Transport Type	Packet Size
AAL5	0-32 bytes
Ethernet VLAN	18 bytes
Ethernet Port	14 bytes
Frame Relay DLCI	2 bytes for Cisco encapsulation, 8 bytes for Internet Engineering Task Force (IETF) encapsulation
HDLC	4 bytes

Transport Type	Packet Size
PPP	4 bytes

AToM Header

The AToM header is 4 bytes (control word). The control word is optional for Ethernet, PPP, HDLC, and cell relay transport types. However, the control word is required for Frame Relay and ATM AAL5 transport types.

MPLS Label Stack

The MPLS label stack size depends on the configuration of the core MPLS network:

- AToM uses one MPLS label to identify the AToM VCs (VC label). Therefore, the minimum MPLS label stack is one for directly connected AToM PEs, which are PE routers that do not have a P router between them.
- If LDP is used in the MPLS network, the label stack size is two (the LDP label and the VC label).
- If a TE tunnel is used instead of LDP between PE routers in the MPLS network, the label stack size is two (the TE label and the VC label).
- If a TE tunnel and LDP are used in the MPLS network (for example, a TE tunnel between P routers or between P and PE routers, with LDP on the tunnel), the label stack is three (the TE label, LDP label, and VC label).
- If you use MPLS fast reroute in the MPLS network, you add a label to the stack. The maximum MPLS label stack in this case is four (the FRR label, TE label, LDP label, and VC label).
- If AToM is used by the customer carrier in an MPLS VPN Carrier Supporting Carrier environment, you add a label to the stack. The maximum MPLS label stack in the provider carrier network is five (the FRR label, TE label, LDP label, VPN label, and VC label).
- If an AToM tunnel spans different service providers that exchange MPLS labels using IPv4 Border Gateway Protocol (BGP) (RFC 3107), you add a label to the stack. The maximum MPLS label stack is five (the FRR label, TE label, Border Gateway Protocol (BGP) label, LDP label, and VC label).

Other circumstances can increase the MPLS label stack size. Therefore, analyze the complete data path between the AToM tunnel endpoints, determine the maximum MPLS label stack size for your network, and then multiply the label stack size by the size of the MPLS label.

- Example Estimating Packet Size, page 8
- mpls mtu Command Changes, page 9

Example Estimating Packet Size

The size of packets is estimated in the following example, which uses the following assumptions:

- The edge MTU is 1500 bytes.
- The transport type is Ethernet VLAN, which designates 18 bytes for the transport header.
- The AToM header is 0, because the control word is not used.
- The MPLS label stack is 2, because LDP is used. The MPLS label is 4 bytes.

```
Edge MTU + Transport header + AToM header + (MPLS label stack * MPLS label) = Core MTU
1500     + 18                    + 0      + (2                * 4         ) = 1526
```

You must configure the P and PE routers in the core to accept packets of 1526 bytes.

Once you determine the MTU size to set on your P and PE routers, you can issue the **mtu**command on the routers to set the MTU size. The following example specifies an MTU of 1526 bytes:

```
Router(config-if)# mtu 1526
```

mpls mtu Command Changes

Some interfaces (such as FastEthernet) require the **mpls mtu** command to change the MTU size. In Cisco IOS Release 12.2(25)S, the behavior of the **mpls mtu** command changed.

If the interface MTU is fewer than 1524 bytes, you can set the maximum MPLS MTU to 24 bytes more than the interface MTU. For example, if the interface MTU is set to 1510 bytes, then you can set the maximum MPLS MTU to 1534 bytes (1510 + 24).

Caution Although you can set the MPLS MTU to a value greater than the interface MTU, you must set the MPLS MTU value to less than or equal to the interface MTU to prevent data corruption, dropped packets, and high CPU rates.

If the interface MTU is greater than or equal to 1524 bytes, then you can set the maximum MPLS MTU value to as high as the interface MTU value. For example, if the interface MTU is set to 1600 bytes, then you can set the MPLS MTU to a maximum of 1600 bytes. If you set the MPLS MTU value to higher than the interface MTU, traffic is dropped.

For interfaces that do not allow you to configure the interface MTU value and for interfaces where the interface MTU is 1500 bytes, the MPLS MTU range is 64 to 1524 bytes.

For GRE tunnel interfaces you can set the MPLS MTU value to either the default value or the maximum value that is supported by the platform for the interface.

You can set the MPLS MTU value to the maximum value by using the **max** keyword along with the **mpls mtu** command. The **mpls mtu max** command allows the previously dropped packets to pass through the GRE tunnel by fragmentation on the underlying physical interface.

Note that the MPLS MTU value cannot be greater than the interface MTU value for non-GRE tunnels.

If you upgrade to Cisco IOS Release 12.2(25)S and you have an MPLS MTU setting that does not conform to these guidelines, the command is rejected.

For Cisco IOS Release 12.2(27)SBC, 12.2(33)SRA, 12.4(11)T, 12.2(33)SXH, and later releases, you cannot set the MPLS MTU to a value greater than the interface MTU. This eliminates problems, such as dropped packets, data corruption, and high CPU rates. See the MPLS MTU Command Changes document for more information.

Frame Relay over MPLS and DTE DCE and NNI Connections

You can configure an interface as a DTE device or a DCE switch, or as a switch connected to a switch with network-to-network interface (NNI) connections. Use the following command in interface configuration mode:

frame-relay intf-type [**dce** | **dte** | **nni**]

The keywords are explained in the table below.

Table 2 ***frame-relay intf-type Command Keywords***

Keyword	Description
dce	Enables the router or access server to function as a switch connected to a router.
dte	Enables the router or access server to function as a DTE device. DTE is the default.
nni	Enables the router or access server to function as a switch connected to a switch.

- Local Management Interface and Frame Relay over MPLS, page 10

Local Management Interface and Frame Relay over MPLS

Local Management Interface (LMI) is a protocol that communicates status information about PVCs. When a PVC is added, deleted, or changed, the LMI notifies the endpoint of the status change. LMI also provides a polling mechanism that verifies that a link is up.

- How LMI Works, page 10

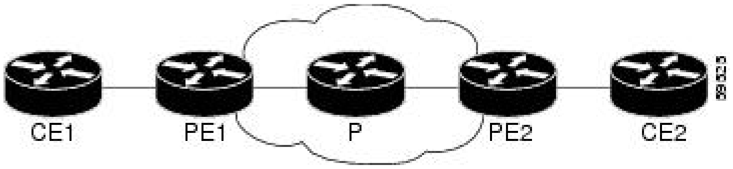
How LMI Works

To determine the PVC status, LMI checks that a PVC is available from the reporting device to the Frame Relay end-user device. If a PVC is available, LMI reports that the status is “Active,” which means that all interfaces, line protocols, and core segments are operational between the reporting device and the Frame Relay end-user device. If any of those components is not available, the LMI reports a status of “Inactive.”

Note Only the DCE and NNI interface types can report the LMI status.

The figure below is a sample topology that helps illustrate how LMI works.

Figure 1 ***Sample Topology***



In the figure above, note the following:

- CE1 and PE1 and PE2 and CE2 are Frame Relay LMI peers.
- CE1 and CE2 can be Frame Relay switches or end-user devices.
- Each Frame Relay PVC comprises multiple segments.
- The DLCI value is local to each segment and is changed as traffic is switched from segment to segment. Two Frame Relay PVC segments exist in the figure; one is between PE1 and CE1 and the other is between PE2 and CE2.

The LMI protocol behavior depends on whether you have DLCI-to-DLCI or port-to-port connections.

DLCI-to-DLCI Connections

If you have DLCI-to-DLCI connections, LMI runs locally on the Frame Relay ports between the PE and CE devices:

- CE1 sends an active status to PE1 if the PVC for CE1 is available. If CE1 is a switch, LMI checks that the PVC is available from CE1 to the user device attached to CE1.
- PE1 sends an active status to CE1 if the following conditions are met:
    - A PVC for PE1 is available.
    - PE1 received an MPLS label from the remote PE router.
    - An MPLS tunnel label exists between PE1 and the remote PE.

For DTE or DCE configurations, the following LMI behavior exists: The Frame Relay device accessing the network (DTE) does not report the PVC status. Only the network device (DCE) or NNI can report the status. Therefore, if a problem exists on the DTE side, the DCE is not aware of the problem.

Port-to-Port Connections

If you have port-to-port connections, the PE routers do not participate in the LMI status-checking procedures. LMI operates only between the CE routers. The CE routers must be configured as DCE-DTE or NNI-NNI.

For information about LMI, including configuration instructions, see the “Configuring the LMI” section of the Configuring Frame Relay document.

QoS Features Supported with AToM

For information about configuring QoS features on Cisco 12000 series routers, see the following feature module:

Any Transport over MPLS (AToM): Layer 2 QoS for the Cisco 12000 Series Router (Quality of Service)

The tables below list the QoS features supported by AToM on the Cisco 7200 and 7500 series routers.

Table 3 QoS Features Supported with Ethernet over MPLS on the Cisco 7200 and 7500 Series Routers

QoS Feature	Ethernet over MPLS
Service policy	Can be applied to: • Interface (input and output) • Subinterface (input and output)
Classification	Supports the following commands: • **match cos** (on interfaces and subinterfaces) • **match mpls experimental** (on interfaces and subinterfaces) • **match qos-group** (on interfaces) (output policy)

QoS Feature	Ethernet over MPLS
Marking	Supports the following commands: • **set cos** (output policy) • **set discard-class** (input policy) • **set mpls experimental** (input policy) (on interfaces and subinterfaces) • **set qos-group** (input policy)
Policing	Supports the following: • Single-rate policing • Two-rate policing • Color-aware policing • Multiple-action policing
Queueing and shaping	Supports the following: • Distributed Low Latency Queueing (dLLQ) • Distributed Weighted Random Early Detection (dWRED) • Byte-based WRED

Table 4 QoS Features Supported with Frame Relay over MPLS on the Cisco 7200 and 7500 Series Routers

QoS Feature	Frame Relay over MPLS
Service policy	Can be applied to: • Interface (input and output) • PVC (input and output)
Classification	Supports the following commands: • **match fr-de** (on interfaces and VCs) • **match fr-dlci** (on interfaces) • **match qos-group**
Marking	Supports the following commands: • **frame-relay congestion management** (output) • **set discard-class** • **set fr-de** (output policy) • **set fr-fecn-becn** (output) • **set mpls experimental** • **set qos-group** • **threshold ecn** (output)

QoS Feature	Frame Relay over MPLS
Policing	Supports the following: • Single-rate policing • Two-rate policing • Color-aware policing • Multiple-action policing
Queueing and shaping	Supports the following: • dLLQ • dWRED • Distributed traffic shaping • Distributed class-based weighted fair queueing (dCBWFQ) • Byte-based WRED • **random-detect discard-class-based** command

Table 5 QoS Features Supported with ATM Cell Relay and AAL5 over MPLS on the Cisco 7200 and 7500 Series Routers

QoS Feature	ATM Cell Relay and AAL5 over MPLS
Service policy	Can be applied to: • Interface (input and output) • Subinterface (input and output) • PVC (input and output)
Classification	Supports the following commands: • **match mpls experimental** (on VCs) • **match qos-group** (output)
Marking	Supports the following commands: • **random-detect discard-class-based** (input) • **set clp** (output) (on interfaces, subinterfaces, and VCs) • **set discard-class** (input) • **set mpls experimental** (input) (on interfaces, subinterfaces, and VCs) • **set qos-group** (input)

QoS Feature	ATM Cell Relay and AAL5 over MPLS
Policing	Supports the following: • Single-rate policing • Two-rate policing • Color-aware policing • Multiple-action policing
Queueing and shaping	Supports the following: • dLLQ • dWRED • dCBWFQ • Byte-based WRED • random-detect discard-class-based command • Class-based shaping support on ATM PVCs

How to Configure Any Transport over MPLS

This section explains how to perform a basic AToM configuration and includes the following procedures:

- Configuring the Pseudowire Class, page 15
- Configuring ATM AAL5 over MPLS on PVCs, page 16
- Configuring ATM AAL5 over MPLS in VC Class Configuration Mode, page 18
- Configuring OAM Cell Emulation for ATM AAL5 over MPLS, page 21
- Configuring ATM Cell Relay over MPLS in VC Mode, page 27
- Configuring ATM Cell Relay over MPLS in VC Mode Using VC Class Configuration Mode, page 29
- Configuring ATM Cell Relay over MPLS in PVP Mode, page 31
- Configuring ATM Cell Relay over MPLS in Port Mode, page 34
- Configuring ATM Single Cell Relay over MPLS, page 36
- Configuring ATM Packed Cell Relay over MPLS, page 38
- Configuring Ethernet over MPLS in VLAN Mode, page 51
- Configuring Ethernet over MPLS in Port Mode, page 52
- Configuring Ethernet over MPLS with VLAN ID Rewrite, page 54
- Configuring Ethernet over MPLS with VLAN ID Rewrite for Cisco 12k Routers for 12.0(29)S and Earlier Releases, page 54
- Configuring Ethernet over MPLS with VLAN ID Rewrite for Cisco 12k Routers for 12.0(30)S and Later Releases, page 55
- Configuring per-Subinterface MTU for Ethernet over MPLS, page 58
- Configuring Frame Relay over MPLS with DLCI-to-DLCI Connections, page 61
- Configuring Frame Relay over MPLS with Port-to-Port Connections, page 63
- Configuring HDLC and PPP over MPLS, page 64
- Configuring Tunnel Selection, page 66
- Setting Experimental Bits with AToM, page 70
- Setting the Frame Relay Discard Eligibility Bit on the Cisco 7200 and 7500 Series Routers, page 75

- Matching the Frame Relay DE Bit on the Cisco 7200 and 7500 Series Routers, page 77
- Enabling the Control Word, page 78

Configuring the Pseudowire Class

The successful transmission of the Layer 2 frames between PE routers is due to the configuration of the PE routers. You set up the connection, called a pseudowire, between the routers.

Note In simple configurations, this task is optional. You do not need to specify a pseudowire class if you specify the tunneling method as part of the **xconnect**command.

The pseudowire-class configuration group specifies the following characteristics of the tunneling mechanism:

- Encapsulation type
- Control protocol
- Payload-specific options

For more information about the **pseudowire-class**command, see the following feature module: Layer 2 Tunnel Protocol Version 3.

You must specify the **encapsulation mpls**commandas part of the pseudowire class or as part of the **xconnect** command for the AToM VCs to work properly. If you omit the **encapsulation mpls**commandas part of the **xconnect**command, you will receive the following error:

```
% Incomplete command.
```

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **pseudowire-class** *name*
4. **encapsulation mpls**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**pseudowire-class** *name* **Example:** `Router(config)# pseudowire-class atom`	Establishes a pseudowire class with a name that you specify and enters pseudowire class configuration mode.
Step 4	**encapsulation mpls** **Example:** `Router(config-pw-class)# encapsulation mpls`	Specifies the tunneling encapsulation.

Step 5	**end**
	Example:
	`Router(config-pw-class)# end`
		Exits pseudowire class configuration mode and returns to privileged EXEC mode.

To change the type of encapsulation, remove the pseudowire with the **no pseudowire-class** command, reestablish the pseudowire, and specify the new encapsulation type.

Once you specify the **encapsulation mpls** command, you can neither remove it using the **no encapsulation mpls**command nor change the command setting using the **encapsulation l2tpv3** command. If you try to remove or change the encapsulation type using the above-mentioned commands, you will get the following error message:

```
Encapsulation changes are not allowed on an existing pw-class.
```

To remove a pseudowire, use the **clear xconnect** command in privileged EXEC mode. You can remove all pseudowires or specific pseudowires on an interface or peer router.

Configuring ATM AAL5 over MPLS on PVCs

ATM AAL5 over MPLS for PVCs encapsulates ATM AAL5 service data unit (SDUs) in MPLS packets and forwards them across the MPLS network. Each ATM AAL5 SDU is transported as a single packet.

Note AAL5 over MPLS is supported only in SDU mode.

>

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *typeslot/port*
4. **pvc** [*name*] *vpi/vci* **l2transport**
5. **encapsulation aal5**
6. **xconnect** *peer-router-id vcid* **encapsulation mpls**
7. **exit**
8. **exit**
9. **exit**
10. **show mpls l2transport vc**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *typeslot/port* **Example:** `Router(config)# interface atm1/0`	Specifies the interface by type, slot, and port number, and enters interface configuration mode.
Step 4	**pvc** [*name*] *vpi/vci* **l2transport** **Example:** `Router(config-if)# pvc 1/200 l2transport`	Creates or assigns a name to an ATM PVC and enters L2transport PVC configuration mode. • The **l2transport** keyword indicates that the PVC is a switched PVC instead of a terminated PVC.
Step 5	**encapsulation aal5** **Example:** `Router(config-if-atm-l2trans-pvc)# encapsulation aal5`	Specifies the ATM ALL5 encapsulation for the PVC. • Make sure that you specify the same encapsulation type on the PE and CE routers.

	Command or Action	Purpose
Step 6	**xconnect** *peer-router-id vcid* **encapsulation mpls** **Example:** `Router(config-if-atm-l2trans-pvc)# xconnect 10.13.13.13 100 encapsulation mpls`	Binds the attachment circuit to a pseudowire VC.
Step 7	**exit** **Example:** `Router(config-if-atm-l2trans-pvc)# exit`	Exits L2transport PVC configuration mode.
Step 8	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode.
Step 9	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode.
Step 10	**show mpls l2transport vc** **Example:** `Router# show mpls l2transport vc`	Displays output that shows ATM AAL5 over MPLS is configured on a PVC.

Examples

The following is sample output from the **show mpls l2transport vc**command, which shows that ATM AAL5 over MPLS is configured on a PVC:

```
Router# show mpls l2transport vc
Local intf   Local circuit         Dest address      VC ID      Status
---------    -------------         ------------      -----      ------
ATM1/0       ATM AAL5 1/100        10.4.4.4           100        UP
```

Configuring ATM AAL5 over MPLS in VC Class Configuration Mode

You can create a VC class that specifies the AAL5 encapsulation and then attach the encapsulation type to an interface, subinterface, or PVC. The following task creates a VC class and attaches it to a main interface.

Note AAL5 over MPLS is supported only in SDU mode.

>

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vc-class atm** *vc-class-name*
4. **encapsulation** *layer-type*
5. **exit**
6. **interface** *typeslot/port*
7. **class-int** *vc-class-name*
8. **pvc** [*name*] *vpi/vci* **l2transport**
9. **xconnect** *peer-router-id vcid* **encapsulation mpls**
10. **exit**
11. **exit**
12. **exit**
13. **show atm class-links**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**vc-class atm** *vc-class-name* **Example:** `Router(config)# vc-class atm aal5class`	Creates a VC class and enters VC class configuration mode.

	Command or Action	Purpose
Step 4	**encapsulation** *layer-type* **Example:** `Router(config-vc-class)# encapsulation aal5`	Configures AAL and the encapsulation type.
Step 5	**exit** **Example:** `Router(config-vc-class)# exit`	Exits VC class configuration mode.
Step 6	**interface** *typeslot/port* **Example:** `Router(config)# interface atm1/0`	Specifies the interface by type, slot, and port number, and enters interface configuration mode.

Step 7	**class-int** *vc-class-name*
	Example:
	`Router(config-if)# class-int aal5class`
		Applies a VC class to the ATM main interface or subinterface.
		Note You can also apply a VC class to a PVC.

Step 8	**pvc** [*name*] *vpi/vci* **l2transport** **Example:** `Router(config-if)# pvc 1/200 l2transport`	Creates or assigns a name to an ATM PVC and enters L2transport PVC configuration mode. • The **l2transport** keyword indicates that the PVC is a switched PVC instead of a terminated PVC.
Step 9	**xconnect** *peer-router-id vcid* **encapsulation mpls** **Example:** `Router(config-if-atm-l2trans-pvc)# xconnect 10.13.13.13 100 encapsulation mpls`	Binds the attachment circuit to a pseudowire VC.
Step 10	**exit** **Example:** `Router(config-if-atm-l2trans-pvc)# exit`	Exits L2transport PVC configuration mode.

	Command or Action	Purpose
Step 11	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode.
Step 12	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode.
Step 13	**show atm class-links** **Example:** `Router# show atm class-links`	Shows the type of encapsulation and that the VC class was applied to an interface.

Examples

In the following example, the command output of the **show atm class-links**command verifies that ATM AAL5 over MPLS is configured as part of a VC class. The command output shows the type of encapsulation and that the VC class was applied to an interface.

```
Router# show atm class-links 1/100
Displaying vc-class inheritance for ATM1/
0.0, vc 1/
100:
no broadcast - Not configured - using default
encapsulation aal5 - VC-class configured on main interface
```

Configuring OAM Cell Emulation for ATM AAL5 over MPLS

If a PE router does not support the transport of Operation, Administration, and Maintenance (OAM) cells across a label switched path (LSP), you can use OAM cell emulation to locally terminate or loop back the OAM cells. You configure OAM cell emulation on both PE routers, which emulates a VC by forming two unidirectional LSPs. You use the **oam-ac emulation-enable**and **oam-pvc manage**commands on both PE routers to enable OAM cell emulation.

After you enable OAM cell emulation on a router, you can configure and manage the ATM VC in the same manner as you would a terminated VC. A VC that has been configured with OAM cell emulation can send loopback cells at configured intervals toward the local CE router. The endpoint can be either of the following:

- End-to-end loopback, which sends OAM cells to the local CE router.
- Segment loopback, which responds to OAM cells to a device along the path between the PE and CE routers.

The OAM cells include the following cells:

- Alarm indication signal (AIS)

- Remote defect indication (RDI)

These cells identify and report defects along a VC. When a physical link or interface failure occurs, intermediate nodes insert OAM AIS cells into all the downstream devices affected by the failure. When a router receives an AIS cell, it marks the ATM VC down and sends an RDI cell to let the remote end know about the failure.

This section contains two tasks:

- Configuring OAM Cell Emulation for ATM AAL5 over MPLS on PVCs, page 22
- Configuring OAM Cell Emulation for ATM AAL5 over MPLS in VC Class Configuration Mode, page 25

Configuring OAM Cell Emulation for ATM AAL5 over MPLS on PVCs

Perform this task to configure OAM cell emulation for ATM AAL5 over MPLS on a PVC.

Note For AAL5 over MPLS, you can configure the **oam-pvc manage**commandonly after you issue the **oam-ac emulation-enable** command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *typeslot* /port
4. **pvc** [*name*] *vpi/vci* **l2transport**
5. **encapsulation aal5**
6. **xconnect** *peer-router-id vcid* **encapsulation mpls**
7. **oam-ac emulation-enable** [*ais-rate*]
8. **oam-pvc manage** [*frequency*]
9. **exit**
10. **exit**
11. **exit**
12. **show atm pvc**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *typeslot* /port **Example:** `Router(config)# interface atm1/0`	Specifies the interface by type, slot, and port number, and enters interface configuration mode.
Step 4	**pvc** [*name*] *vpi/vci* **l2transport** **Example:** `Router(config-if)# pvc 1/200 l2transport`	Creates or assigns a name to an ATM PVC and enters L2transport PVC configuration mode. • The **l2transport** keyword indicates that the PVC is a switched PVC instead of a terminated PVC.
Step 5	**encapsulation aal5** **Example:** `Router(config-if-atm-l2trans-pvc)# encapsulation aal5`	Specifies ATM AAL5 encapsulation for the PVC. • Make sure you specify the same encapsulation type on the PE and CE routers.
Step 6	**xconnect** *peer-router-id vcid* **encapsulation mpls** **Example:** `Router(config-if-atm-l2trans-pvc)# xconnect 10.13.13.13 100 encapsulation mpls`	Binds the attachment circuit to a pseudowire VC.
Step 7	**oam-ac emulation-enable** [*ais-rate*] **Example:** `Router(config-if-atm-l2trans-pvc)# oam-ac emulation-enable 30`	Enables OAM cell emulation for AAL5 over MPLS. • The *ais-rate* argument lets you specify the rate at which AIS cells are sent. The default is one cell every second. The range is 0 to 60 seconds.
Step 8	**oam-pvc manage** [*frequency*] **Example:** `Router(config-if-atm-l2trans-pvc)# oam-pvc manage`	Enables the PVC to generate end-to-end OAM loopback cells that verify connectivity on the virtual circuit. • The optional *frequency* argument is the interval between transmission of loopback cells and ranges from 0 to 600 seconds. The default value is 10 seconds.

	Command or Action	Purpose
Step 9	**exit** **Example:** `Router(config-if-atm-l2trans-pvc)# exit`	Exits L2transport PVC configuration mode.
Step 10	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode.
Step 11	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode.
Step 12	**show atm pvc** **Example:** `Router# show atm pvc`	Displays output that shows OAM cell emulation is enabled on the ATM PVC.

Examples

The output of the **show atm pvc** command in the following example shows that OAM cell emulation is enabled on the ATM PVC:

```
Router# show atm pvc 5/500
ATM4/1/0.200: VCD: 6, VPI: 5, VCI: 500
UBR, PeakRate: 1
AAL5-LLC/SNAP, etype:0x0, Flags: 0x34000C20, VCmode: 0x0
OAM Cell Emulation: enabled, F5 End2end AIS Xmit frequency: 1 second(s)
OAM frequency: 0 second(s), OAM retry frequency: 1 second(s)
OAM up retry count: 3, OAM down retry count: 5
OAM Loopback status: OAM Disabled
OAM VC state: Not ManagedVerified
ILMI VC state: Not Managed
InPkts: 564, OutPkts: 560, InBytes: 19792, OutBytes: 19680
InPRoc: 0, OutPRoc: 0
InFast: 4, OutFast: 0, InAS: 560, OutAS: 560
InPktDrops: 0, OutPktDrops: 0
CrcErrors: 0, SarTimeOuts: 0, OverSizedSDUs: 0
Out CLP=1 Pkts: 0
OAM cells received: 26
F5 InEndloop: 0, F5 InSegloop: 0, F5 InAIS: 0, F5 InRDI: 26
OAM cells sent: 77
F5 OutEndloop: 0, F5 OutSegloop: 0, F5 OutAIS: 77, F5 OutRDI: 0
OAM cell drops: 0
Status: UP
```

Configuring OAM Cell Emulation for ATM AAL5 over MPLS in VC Class Configuration Mode

The following steps explain how to configure OAM cell emulation as part of a VC class. You can then apply the VC class to an interface, a subinterface, or a VC. When you configure OAM cell emulation in VC class configuration mode and then apply the VC class to an interface, the settings in the VC class apply to all the VCs on the interface, unless you specify a different OAM cell emulation value at a lower level, such as the subinterface or VC level. For example, you can create a VC class that specifies OAM cell emulation and sets the rate of AIS cells to every 30 seconds. You can apply the VC class to an interface. Then, for one PVC, you can enable OAM cell emulation and set the rate of AIS cells to every 15 seconds. All the PVCs on the interface use the cell rate of 30 seconds, except for the one PVC that was set to 15 seconds.

Perform this task to enable OAM cell emulation as part of a VC class and apply it to an interface.

Note For AAL5 over MPLS, you can configure the **oam-pvc manage**command only after you issue the **oam-ac emulation-enable** command**.**

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vc-class atm** *name*
4. **encapsulation** *layer-type*
5. **oam-ac emulation-enable** [*ais-rate*]
6. **oam-pvc manage** [*frequency*]
7. **exit**
8. **interface** *typeslot/port*
9. **class-int** *vc-class-name*
10. **pvc** [*name*] *vpi/vci* **l2transport**
11. **xconnect** *peer-router-id vcid* **encapsulation mpls**
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**vc-class atm** *name* **Example:** `Router(config)# vc-class atm oamclass`	Creates a VC class and enters VC class configuration mode.
Step 4	**encapsulation** *layer-type* **Example:** `Router(config-vc-class)# encapsulation aal5`	Configures the AAL and encapsulation type.
Step 5	**oam-ac emulation-enable** [*ais-rate*] **Example:** `Router(config-vc-class)# oam-ac emulation-enable 30`	Enables OAM cell emulation for AAL5 over MPLS. • The *ais-rate* argument lets you specify the rate at which AIS cells are sent. The default is one cell every second. The range is 0 to 60 seconds.
Step 6	**oam-pvc manage** [*frequency*] **Example:** `Router(config-vc-class)# oam-pvc manage`	Enables the PVC to generate end-to-end OAM loopback cells that verify connectivity on the virtual circuit. • The optional *frequency* argument is the interval between transmission of loopback cells and ranges from 0 to 600 seconds. The default value is 10 seconds.
Step 7	**exit** **Example:** `Router(config-vc-class)# exit`	Exits VC class configuration mode.
Step 8	**interface** *typeslot/port* **Example:** `Router(config)# interface atm1/0`	Specifies the interface by type, slot, and port number, and enters interface configuration mode.

Step 9	**class-int** *vc-class-name*
	Example:
	`Router(config-if)# class-int oamclass`
		Applies a VC class to the ATM main interface or subinterface.
		Note You can also apply a VC class to a PVC.

	Command or Action	Purpose
Step 10	**pvc** [*name*] *vpi/vci* **l2transport** **Example:** `Router(config-if)# pvc 1/200 l2transport`	Creates or assigns a name to an ATM PVC and enters L2transport PVC configuration mode. • The **l2transport** keyword indicates that the PVC is a switched PVC instead of a terminated PVC.
Step 11	**xconnect** *peer-router-id vcid* **encapsulation mpls** **Example:** `Router(config-if-atm-l2trans-pvc)# xconnect 10.13.13.13 100 encapsulation mpls`	Binds the attachment circuit to a pseudowire VC.

Step 12	**end**
	Example:
	`Router(config-if-atm-l2trans-pvc)# end`
		Exits L2transport PVC configuration mode and returns to privileged EXEC mode.

Configuring ATM Cell Relay over MPLS in VC Mode

Perform this task to configure ATM cell relay on the permanent virtual circuits.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface atm** *slot* /*port*
4. **pvc** *vpi/vci* **l2transport**
5. **encapsulation aal0**
6. **xconnect** *peer-router-id vcid* **encapsulation mpls**
7. **exit**
8. **exit**
9. **exit**
10. **show atm vc**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface atm** *slot* /*port* **Example:** `Router(config)# interface atm1/0`	Specifies an ATM interface and enters interface configuration mode.
Step 4	**pvc** *vpi/vci* **l2transport** **Example:** `Router(config-if)# pvc 0/100 l2transport`	Assigns a virtual path identifier (VPI) and virtual circuit identifier (VCI) and enters L2transport PVC configuration mode. • The **l2transport**keyword indicates that the PVC is a switched PVC instead of a terminated PVC.
Step 5	**encapsulation aal0** **Example:** `Router(config-if-atm-l2trans-pvc)# encapsulation aal0`	For ATM cell relay, specifies raw cell encapsulation for the interface. • Make sure you specify the same encapsulation type on the PE and CE routers.
Step 6	**xconnect** *peer-router-id vcid* **encapsulation mpls** **Example:** `Router(config-if-atm-l2trans-pvc)# xconnect 10.13.13.13 100 encapsulation mpls`	Binds the attachment circuit to a pseudowire VC.
Step 7	**exit** **Example:** `Router(config-if-atm-l2trans-pvc)# exit`	Exits L2transport PVC configuration mode.

	Command or Action	Purpose
Step 8	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode.
Step 9	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode.
Step 10	**show atm vc** **Example:** `Router# show atm vc`	Verifies that OAM cell emulation is enabled on the ATM VC.

Examples

The output of the **show atm vc** command shows that the interface is configured for VC mode cell relay:

```
Router# show atm vc 7
ATM3/0: VCD: 7, VPI: 23, VCI: 100
UBR, PeakRate: 149760
AAL0-Cell Relay, etype:0x10, Flags: 0x10000C2D, VCmode: 0x0
OAM Cell Emulation: not configured
InBytes: 0, OutBytes: 0
Status: UP
```

Configuring ATM Cell Relay over MPLS in VC Mode Using VC Class Configuration Mode

You can create a VC class that specifies the ATM cell relay encapsulation and then attach the VC class to an interface, subinterface, or VC. The following task creates a VC class that specifies the ATM cell relay encapsulation and attaches it to a main interface.

Note You can configure VC class configuration mode only in VC mode. VC class configuration mode is not supported on VP or port mode.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vc-class atm** *name*
4. **encapsulation** *layer-type*
5. **exit**
6. **interface** *typeslot* /*port*
7. **class-int** *vc-class-name*
8. **pvc** [*name*] *vpi/vci* **l2transport**
9. **xconnect** *peer-router-id vcid* **encapsulation mpls**
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**vc-class atm** *name* **Example:** `Router(config)# vc-class atm cellrelay`	Creates a VC class and enters VC class configuration mode.
Step 4	**encapsulation** *layer-type* **Example:** `Router(config-vc-class)# encapsulation aal0`	Configures the AAL and encapsulation type.
Step 5	**exit** **Example:** `Router(config-vc-class)# exit`	Exits VC class configuration mode.

	Command or Action	Purpose
Step 6	**interface** *typeslot* /*port* **Example:** `Router(config)# interface atm1/0`	Specifies the interface by type, slot, and port number, and enters interface configuration mode.

Step 7	**class-int** *vc-class-name*
	Example:
	`Router(config-if)# class-int cellrelay`
		Applies a VC class to the ATM main interface or subinterface.
		Note You can also apply a VC class to a PVC.

Step 8	**pvc** [*name*] *vpi*/*vci* **l2transport** **Example:** `Router(config-if)# pvc 1/200 l2transport`	Creates or assigns a name to an ATM PVC and enters L2transport PVC configuration mode. • The **l2transport** keyword indicates that the PVC is a switched PVC instead of a terminated PVC.
Step 9	**xconnect** *peer-router-id vcid* **encapsulation mpls** **Example:** `Router(config-if-atm-l2trans-pvc)# xconnect 10.13.13.13 100 encapsulation mpls`	Binds the attachment circuit to a pseudowire VC.

Step 10	**end**
	Example:
	`Router(config-if-atm-l2trans-pvc)# end`
		Exits L2transport PVC configuration mode and returns to privileged EXEC mode.

Configuring ATM Cell Relay over MPLS in PVP Mode

VP mode allows cells coming into a predefined PVP on the ATM interface to be transported over the MPLS backbone to a predefined PVP on the egress ATM interface. You can use VP mode to send single cells or packed cells over the MPLS backbone.

To configure VP mode, you must specify the following:

- The VP for transporting cell relay cells.
- The IP address of the peer PE router and the VC ID.

When configuring ATM cell relay over MPLS in VP mode, use the following guidelines:

- You do not need to enter the **encapsulation aal0** command in VP mode.
- One ATM interface can accommodate multiple types of ATM connections. VP cell relay, VC cell relay, and ATM AAL5 over MPLS can coexist on one ATM interface. On the Cisco 12000 series router, this is true only on the engine 0 ATM line cards.

- If a VPI is configured for VP cell relay, you cannot configure a PVC using the same VPI.
- VP trunking (mapping multiple VPs to one emulated VC label) is not supported. Each VP is mapped to one emulated VC.
- Each VP is associated with one unique emulated VC ID. The AToM emulated VC type is ATM VP cell transport.
- The AToM control word is supported. However, if a peer PE does not support the control word, it is disabled. This negotiation is done by LDP label binding.
- VP mode (and VC mode) drop idle cells.

Perform this task to configure ATM cell relay in PVP mode.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface atm** *slot* /*port*
4. **atm pvp** *vpi* **l2transport**
5. **xconnect** *peer-router-id vcid* **encapsulation mpls**
6. **exit**
7. **exit**
8. **exit**
9. **show atm vp**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface atm** *slot* /*port* **Example:** `Router(config)# interface atm1/0`	Defines the interface and enters interface configuration mode.

	Command or Action	Purpose
Step 4	**atm pvp** *vpi* **l2transport** **Example:** `Router(config-if)# atm pvp 1 l2transport`	Specifies that the PVP is dedicated to transporting ATM cells and enters L2transport PVP configuration mode. • The **l2transport** keyword indicates that the PVP is for cell relay. This mode is for Layer 2 transport only; it is not for regular PVPs.
Step 5	**xconnect** *peer-router-id vcid* **encapsulation mpls** **Example:** `Router(config-if-atm-l2trans-pvp)# xconnect 10.0.0.1 123 encapsulation mpls`	Binds the attachment circuit to a pseudowire VC. • The syntax for this command is the same as for all other Layer 2 transports.
Step 6	**exit** **Example:** `Router(config-if-atm-l2trans-pvP)# exit`	Exits L2 transport PVP configuration mode.
Step 7	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode.
Step 8	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode.
Step 9	**show atm vp** **Example:** `Router# show atm vp`	Displays output that shows OAM cell emulation is enabled on the ATM VP.

Examples

The following **show atm vp** command in the following example shows that the interface is configured for VP mode cell relay:

```
Router# show atm vp 1
ATM5/0  VPI: 1, Cell Relay, PeakRate: 149760, CesRate: 0, DataVCs: 1, CesVCs: 0, Status:
ACTIVE
  VCD    VCI   Type   InPkts   OutPkts   AAL/Encap     Status
  6      3     PVC    0        0         F4 OAM        ACTIVE
  7      4     PVC    0        0         F4 OAM        ACTIVE
```

```
TotalInPkts: 0, TotalOutPkts: 0, TotalInFast: 0, TotalOutFast: 0,
TotalBroadcasts: 0 TotalInPktDrops: 0, TotalOutPktDrops: 0
```

Configuring ATM Cell Relay over MPLS in Port Mode

Port mode cell relay allows cells coming into an ATM interface to be packed into an MPLS packet and transported over the MPLS backbone to an egress ATM interface.

To configure port mode, issue the **xconnect** command from an ATM main interface and specify the destination address and the VC ID. The syntax of the **xconnect** command is the same as for all other transport types. Each ATM port is associated with one unique pseudowire VC label.

When configuring ATM cell relay over MPLS in port mode, use the following guidelines:

- The pseudowire VC type is set to ATM transparent cell transport (AAL0).
- The AToM control word is supported. However, if the peer PE does not support a control word, the control word is disabled. This negotiation is done by LDP label binding.

Note The AToM control word is not supported for port mode cell relay on Cisco 7600 series routers.

- Port mode and VP and VC mode are mutually exclusive. If you enable an ATM main interface for cell relay, you cannot enter any PVP or PVC commands.
- If the pseudowire VC label is withdrawn due to an MPLS core network failure, the PE router sends a line AIS to the CE router.
- For the Cisco 7600 series routers, you must specify the interface ATM slot, bay, and port for the SIP400 or SIP200.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface atm** *slot* /*port*
4. **xconnect** *peer-router-id vcid* **encapsulation mpls**
5. **exit**
6. **exit**
7. **show atm route**
8. **show mpls l2transport vc**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

Step 3	**interface atm** *slot* /*port*
	Example:
	or **interface atm** *slot/bay/port*
	Example:
	`Router(config)# interface atm1/0`
	Example:
	`or`
	Example:
	`Router(config)# interface atm4/3/0`
		Specifies an ATM interface and enters interface configuration mode.
		• For the Cisco 7600 series routers, you must specify the interface ATM slot, bay, and port for the SIP400 or SIP200. In the example the slot is 4, the bay is 3, and the port is 0.

Step 4	**xconnect** *peer-router-id vcid* **encapsulation mpls** **Example:** `Router(config-if)# xconnect 10.0.0.1 123 encapsulation mpls`	Binds the attachment circuit to the interface.
Step 5	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode.
Step 6	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode.

	Command or Action	Purpose
Step 7	**show atm route** **Example:** `Router# show atm route`	Displays output that shows ATM cell relay in port mode has been enabled.
Step 8	**show mpls l2transport vc** **Example:** `Router# show mpls l2transport vc`	Displays the attachment circuit and the interface.

Examples

The **show atm route** command in the following example displays port mode cell relay state. The following example shows that atm interface 1/0 is for cell relay, the VC ID is 123 and the tunnel is down.

```
Router# show atm route
Input Intf        Output Intf     Output VC       Status
ATM1/0            ATOM Tunnel     123             DOWN
```

The **show mpls l2transport vc** command in the following example also shows configuration information:

```
Router# show mpls l2transport vc
Local intf     Local circuit        Dest address    VC ID      Status
-------------  -------------------- --------------- ---------- ----------
AT1/0          ATM CELL ATM1/0      10.1.1.121     1121       UP
```

- Troubleshooting Tips, page 36

Troubleshooting Tips

The **debug atm l2transport** and **debug mpls l2transport vc**display troubleshooting information.

Configuring ATM Single Cell Relay over MPLS

The single cell relay feature allows you to insert one ATM cell in each MPLS packet. You can use single cell relay in both VP and VC mode. The configuration steps show how to configure single cell relay in VC mode. For VP mode, see the Configuring ATM Cell Relay over MPLS in PVP Mode, page 31.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface atm** *slot/port*
4. **pvc** *vpi/vci* **l2transport**
5. **encapsulation aal0**
6. **xconnect** *peer-router-id vcid* **encapsulation mpls**
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface atm** *slot/port* **Example:** `Router(config)# interface atm1/0`	Specifies an ATM interface and enters interface configuration mode.
Step 4	**pvc** *vpi/vci* **l2transport** **Example:** `Router(config-if)# pvc 1/100 l2transport`	Assigns a VPI and VCI and enters L2transport PVC configuration mode. • The **l2transport** keyword indicates that the PVC is a switched PVC instead of a terminated PVC.
Step 5	**encapsulation aal0** **Example:** `Router(config-if-atm-l2trans-pvc)# encapsulation aal0`	Specifies raw cell encapsulation for the interface. • Make sure you specify the same encapsulation type on the PE and CE routers.
Step 6	**xconnect** *peer-router-id vcid* **encapsulation mpls** **Example:** `Router(config-if-atm-l2trans-pvc)# xconnect 10.0.0.1 123 encapsulation mpls`	Binds the attachment circuit to a pseudowire VC.

Step 7	**end**
	Example:
	`Router(config-if-atm-l2trans-pvc)# end`
		Exits L2transport PVC configuration mode and returns to privileged EXEC mode.

Configuring ATM Packed Cell Relay over MPLS

The packed cell relay feature allows you to insert multiple concatenated ATM cells in an MPLS packet. The packed cell relay feature is more efficient than single cell relay, because each ATM cell is 52 bytes, and each AToM packet is at least 64 bytes.

At a high level, packed cell relay configuration consists of the following steps:

1. You specify the amount of time a PE router can wait for cells to be packed into an MPLS packet. You can set up three timers by default with different amounts of time attributed to each timer.
2. You enable packed cell relay, specify how many cells should be packed into each MPLS packet, and choose which timer to use during the cell packing process.

- Restrictions, page 38
- Configuring ATM Packed Cell Relay over MPLS in VC Mode, page 38
- Configuring ATM Packed Cell Relay over MPLS in VC Mode Using VC Class Configuration Mode, page 41
- Configuring ATM Packed Cell Relay over MPLS in VP Mode, page 45
- Configuring ATM Packed Cell Relay over MPLS in Port Mode, page 47
- Troubleshooting Tips, page 51

Restrictions

- The **cell-packing**command is available only if you use AAL0 encapsulation in VC mode. If the command is configured with ATM AAL5 encapsulation, the command is not valid.
- Only cells from the same VC, VP, or port can be packed into one MPLS packet. Cells from different connections cannot be concatenated into the same MPLS packet.
- When you change, enable, or disable the cell-packing attributes, the ATM VC, VP, or port and the MPLS emulated VC are reestablished.
- If a PE router does not support packed cell relay, the PE router sends only one cell per MPLS packet.
- The number of packed cells does not need to match between the PE routers. The two PE routers agree on the lower of the two values. For example, if PE1 is allowed to pack 10 cells per MPLS packet and PE2 is allowed to pack 20 cells per MPLS packet, the two PE routers would agree to send no more than 10 cells per packet.
- If the number of cells packed by the peer PE router exceeds the limit, the packet is dropped.
- Issue the **atm mcpt-timers**command on an ATM interface before issuing the **cell-packing**command.

See the following sections for configuration information:

Configuring ATM Packed Cell Relay over MPLS in VC Mode

Perform this task to configure the ATM packed cell relay over MPLS feature in VC mode.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface atm** *slot* /*port*
4. **shutdown**
5. **atm mcpt-timers** [*timer1-timeout timer2-timeout timer3-timeout*]
6. **no shutdown**
7. **pvc** *vpi/vci* **l2transport**
8. **encapsulation aal0**
9. **xconnect** *peer-router-id vcid* **encapsulation mpls**
10. **cell-packing** *cells* **mcpt-timer** *timer*
11. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface atm** *slot* /*port* **Example:** `Router(config)# interface atm1/0`	Defines the interface and enters interface configuration mode.
Step 4	**shutdown** **Example:** `Router(config-if)# shutdown`	Shuts down the interface.

	Command or Action	Purpose
Step 5	**atm mcpt-timers** [*timer1-timeout timer2-timeout timer3-timeout*] **Example:** `Router(config-if)# atm mcpt-timers 100 200 250` **Example:**	Sets up the cell-packing timers, which specify how long the PE router can wait for cells to be packed into an MPLS packet. • You can set up to three timers. For each timer, you specify the maximum cell-packing timeout (MCPT). This value gives the cell-packing function a limited amount of time to complete. If the timer expires before the maximum number of cells are packed into an AToM packet, the packet is sent anyway. The timeout's default and range of acceptable values depends on the ATM link speed. • The respective default values for the PA-A3 port adapters are: ◦ OC-3: 30, 60, and 90 microseconds ◦ T3: 100, 200, and 300 microseconds ◦ E3: 130, 260, and 390 microseconds • You can specify either the number of microseconds or use the default. • The respective range of values for the PA-A3 port adapters are: ◦ OC-3: 10 to 4095 microseconds ◦ T3: 30 to 4095 microseconds ◦ E3: 40 to 4095 microseconds
Step 6	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Enables the interface.
Step 7	**pvc** *vpi/vci* **l2transport** **Example:** `Router(config-if)# pvc 1/100 l2transport`	Assigns a VPI and VCI and enters L2transport PVC configuration mode. • The **l2transport** keyword indicates that the PVC is a switched PVC instead of a terminated PVC.
Step 8	**encapsulation aal0** **Example:** `Router(config-if-atm-l2trans-pvc)# encapsulation aal0`	Specifies raw cell encapsulation for the interface. • Make sure you specify the same encapsulation type on the PE routers.

	Command or Action	Purpose
Step 9	**xconnect** *peer-router-id vcid* **encapsulation mpls** **Example:** `Router(config-if-atm-l2trans-pvc)# xconnect 10.0.0.1 123 encapsulation mpls`	Binds the attachment circuit to a pseudowire VC.
Step 10	**cell-packing** *cells* **mcpt-timer** *timer* **Example:** `Router(config-if-atm-l2trans-pvc)# cell-packing 10 mcpt-timer 1` **Example:**	Enables cell packing and specifies the cell-packing parameters. • The *cells*argument represents the maximum number of cells to be packed into an MPLS packet. The range is from 2 to the MTU of the interface divided by 52. The default is MTU/52. • The *timer*argument allows you to specify which timer to use. The default is timer 1. • See the **cell-packing** command page for more information.

Step 11	**end**
	Example:
	`Router(config-if-atm-l2trans-pvc)# end`
		Exits L2transport PVC configuration mode and returns to privileged EXEC mode.

Configuring ATM Packed Cell Relay over MPLS in VC Mode Using VC Class Configuration Mode

You can create a VC class that specifies the ATM cell relay encapsulation and the cell packing parameters and then attach the VC class to an interface, subinterface, or VC. The following task creates a VC class that specifies the ATM cell relay encapsulation and cell packing and attaches it to a main interface.

Note You can configure VC class configuration mode only in VC mode. VC class configuration mode is not supported on VP or port mode.

When you configure cell packing in VC class configuration mode and then apply the VC class to an interface, the settings in the VC class apply to all the VCs on the interface, unless you specify a different cell packing value at a lower level, such as the subinterface or VC level. For example, you can create a VC class that specifies three cells to be packed. You can apply the VC class to an interface. Then, for one PVC, you can specify two cells to be packed. All the PVCs on the interface pack three cells, except for the one PVC that was set to set two cells.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vc-class atm** *name*
4. **encapsulation** *layer-type*
5. **cell-packing** *cells* **mcpt-timer** *timer*
6. **exit**
7. **interface** *typeslot* /*port*
8. **shutdown**
9. **atm mcpt-timers** [*timer1-timeout timer2-timeout timer3-timeout*]
10. **no shutdown**
11. **class-int** *vc-class-name*
12. **pvc** [*name*] *vpi*/*vci* **l2transport**
13. **xconnect** *peer-router-id vcid* **encapsulation mpls**
14. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**vc-class atm** *name* **Example:** `Router(config)# vc-class atm cellpacking`	Creates a VC class and enters VC class configuration mode.
Step 4	**encapsulation** *layer-type* **Example:** `Router(config-vc-class)# encapsulation aal0`	Configures the AAL and encapsulation type.

	Command or Action	Purpose
Step 5	**cell-packing** *cells* **mcpt-timer** *timer* **Example:** `Router(config-vc-class)# cell-packing 10 mcpt-timer 1` **Example:**	Enables cell packing and specifies the cell-packing parameters. • The *cells*argument represents the maximum number of cells to be packed into an MPLS packet. The range is from 2 to the MTU of the interface divided by 52. The default is MTU/52. • The *timer*argument allows you to specify which timer to use. The default is timer 1. • See the **cell-packing** command page for more information.
Step 6	**exit** **Example:** `Router(config-vc-class)# exit`	Exits VC class configuration mode.
Step 7	**interface** *typeslot* /*port* **Example:** `Router(config)# interface atm1/0`	Specifies the interface by type, slot, and port number, and enters interface configuration mode.
Step 8	**shutdown** **Example:** `Router(config-if)# shutdown`	Shuts down the interface.

	Command or Action	Purpose
Step 9	**atm mcpt-timers** [*timer1-timeout timer2-timeout timer3-timeout*] **Example:** `Router(config-if)# atm mcpt-timers 100 200 250` **Example:**	Sets up the cell-packing timers, which specify how long the PE router can wait for cells to be packed into an MPLS packet. • You can set up to three timers. For each timer, you specify the MCPT. This value gives the cell-packing function a limited amount of time to complete. If the timer expires before the maximum number of cells are packed into an AToM packet, the packet is sent anyway. The timeout's default and range of acceptable values depends on the ATM link speed. • The respective default values for the PA-A3 port adapters are: ◦ OC-3: 30, 60, and 90 microseconds ◦ T3: 100, 200, and 300 microseconds ◦ E3: 130, 260, and 390 microseconds • You can specify either the number of microseconds or use the default. • The respective range of values for the PA-A3 port adapters are: ◦ OC-3: 10 to 4095 microseconds ◦ T3: 30 to 4095 microseconds ◦ E3: 40 to 4095 microseconds
Step 10	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Enables the interface.

Step 11	**class-int** *vc-class-name*
	Example:
	`Router(config-if)# class-int cellpacking`
		Applies a VC class to the ATM main interface or subinterface.
		Note You can also apply a VC class to a PVC.

Step 12	**pvc** [*name*] *vpi/vci* **l2transport** **Example:** `Router(config-if)# pvc 1/200 l2transport`	Creates or assigns a name to an ATM PVC and enters L2transport PVC configuration mode. • The **l2transport** keyword indicates that the PVC is a switched PVC instead of a terminated PVC.

	Command or Action	Purpose
Step 13	**xconnect** *peer-router-id vcid* **encapsulation mpls** **Example:** `Router(config-if-atm-l2trans-pvc)# xconnect 10.13.13.13 100 encapsulation mpls`	Binds the attachment circuit to a pseudowire VC.

Step 14	**end**
	Example:
	`Router(config-if-atm-l2trans-pvc)# end`
		Exits L2transport PVC configuration mode and returns to privileged EXEC mode.

Configuring ATM Packed Cell Relay over MPLS in VP Mode

Perform this task to configure the ATM cell-packing feature in VP mode.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface atm** *slot* /*port*
4. **shutdown**
5. **atm mcpt-timers** [*timer1-timeout timer2-timeout timer3-timeout*]
6. **no shutdown**
7. **atm pvp** *vpi* **l2transport**
8. **xconnect** *peer-router-id vcid* **encapsulation mpls**
9. **cell-packing** *cells* **mcpt-timer** *timer*
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface atm** *slot* /*port* **Example:** `Router(config)# interface atm1/0`	Defines the interface and enters interface configuration mode.
Step 4	**shutdown** **Example:** `Router(config-if)# shutdown`	Shuts down the interface.
Step 5	**atm mcpt-timers** [*timer1-timeout timer2-timeout timer3-timeout*] **Example:** `Router(config-if)# atm mcpt-timers 100 200 250` **Example:**	Sets up the cell-packing timers, which specify how long the PE router can wait for cells to be packed into an MPLS packet. • You can set up to three timers. For each timer, you specify the MCPT. This value gives the cell-packing function a limited amount of time to complete. If the timer expires before the maximum number of cells are packed into an AToM packet, the packet is sent anyway. The timeout's default and range of acceptable values depends on the ATM link speed. • The respective default values for the PA-A3 port adapters are: ◦ OC-3: 30, 60, and 90 microseconds ◦ T3: 100, 200, and 300 microseconds ◦ E3: 130, 260, and 390 microseconds • You can specify either the number of microseconds or use the default. • The respective range of values for the PA-A3 port adapters are: ◦ OC-3: 10 to 4095 microseconds ◦ T3: 30 to 4095 microseconds ◦ E3: 40 to 4095 microseconds
Step 6	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Enables the interface.

	Command or Action	Purpose
Step 7	**atm pvp** *vpi* **l2transport** **Example:** `Router(config-if)# atm pvp 1 l2transport`	Specifies that the PVP is dedicated to transporting ATM cells and enters L2transport PVP configuration mode. • The **l2transport** keyword indicates that the PVP is for cell relay. This mode is for Layer 2 transport only; it is not for regular PVPs.
Step 8	**xconnect** *peer-router-id vcid* **encapsulation mpls** **Example:** `Router(cfg-if-atm-l2trans-pvp)# xconnect 10.0.0.1 123 encapsulation mpls`	Binds the attachment circuit to a pseudowire VC. • The syntax for this command is the same as for all other Layer 2 transports.
Step 9	**cell-packing** *cells* **mcpt-timer** *timer* **Example:** `Router(cfg-if-atm-l2trans-pvp)# cell-packing 10 mcpt-timer 1` **Example:**	Enables cell packing and specifies the cell-packing parameters. • The *cells*argument represents the maximum number of cells to be packed into an MPLS packet. The range is from 2 to the MTU of the interface divided by 52. The default is MTU/52. • The *timer*argument allows you to specify which timer to use. The default is timer 1. • See the **cell-packing** command page for more information.

Step 10	**end**
	Example:
	`Router(config-if-atm-l2trans-pvc)# end`
		Exits L2transport PVC configuration mode and returns to privileged EXEC mode.

Configuring ATM Packed Cell Relay over MPLS in Port Mode

Perform this task to configure ATM packed cell relay over MPLS in port mode.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface atm** *slot* /*port*
4. **shutdown**
5. **atm mcpt-timers** [*timer1-timeout timer2-timeout timer3-timeout*]
6. **no shutdown**
7. **cell-packing** *cells* **mcpt-timer** *timer*
8. **xconnect** *peer-router-id vcid* **encapsulation mpls**
9. **exit**
10. **exit**
11. **show atm cell-packing**
12. **show atm vp**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface atm** *slot* /*port* **Example:** `Router(config)# interface atm1/0`	Specifies an ATM interface and enters interface configuration mode.
Step 4	**shutdown** **Example:** `Router(config-if)# shutdown`	Shuts down the interface.

	Command or Action	Purpose
Step 5	**atm mcpt-timers** [*timer1-timeout timer2-timeout timer3-timeout*] **Example:** `Router(config-if)# atm mcpt-timers 100 200 250`	Sets up the cell-packing timers, which specify how long the PE router can wait for cells to be packed into an MPLS packet. • You can set up to three timers. For each timer, you specify the MCPT. This value gives the cell-packing function a limited amount of time to complete. If the timer expires before the maximum number of cells are packed into an AToM packet, the packet is sent anyway. The timeout's default and range of acceptable values depends on the ATM link speed. • The respective default values for the PA-A3 port adapters are: ◦ OC-3: 30, 60, and 90 microseconds ◦ T3: 100, 200, and 300 microseconds ◦ E3: 130, 260, and 390 microseconds • You can specify either the number of microseconds or use the default. • The respective range of values for the PA-A3 port adapters are: ◦ OC-3: 10 to 4095 microseconds ◦ T3: 30 to 4095 microseconds ◦ E3: 40 to 4095 microseconds
Step 6	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Enables the interface.
Step 7	**cell-packing** *cells* **mcpt-timer** *timer* **Example:** `Router(config-if)# cell-packing 10 mcpt-timer 1` **Example:**	Enables cell packing and specifies the cell-packing parameters. • The *cells*argument represents the maximum number of cells to be packed into an MPLS packet. The range is from 2 to the MTU of the interface divided by 52. The default is MTU/52. • The *timer*argument allows you to specify which timer to use. The default is timer 1. • See the cell-packing command page for more information.
Step 8	**xconnect** *peer-router-id vcid* **encapsulation mpls** **Example:** `Router(config-if)# xconnect 10.0.0.1 123 encapsulation mpls`	Binds the attachment circuit to the interface.

	Command or Action	Purpose
Step 9	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode.
Step 10	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode.
Step 11	**show atm cell-packing** **Example:** `Router# show atm cell-packing`	Displays cell-packing statistics.
Step 12	**show atm vp** **Example:** `Router# show atm vp`	Displays cell-packing information.

Examples

The **show atm cell-packing** command in the following example displays the following statistics:

- The number of cells that are to be packed into an MPLS packet on the local and peer routers
- The average number of cells sent and received
- The timer values associated with the local router

```
Router# show atm cell-packing
                         average                 average
        circuit  local  nbr of cells    peer    nbr of cells    MCPT
        type     MNCP   rcvd in one pkt MNCP    sent in one pkt (us)
==============================================================================
atm 1/0 vc 1/200  20    15              30              20       60
atm 1/0 vp 2      25    21              30              24      100
```

The **show atm vp** command in the following example displays the cell packing information at the end of the output:

```
Router# show atm vp 12
ATM5/0  VPI: 12, Cell Relay, PeakRate: 149760, CesRate: 0, DataVCs: 1, CesVCs: 0, Status:
ACTIVE
  VCD    VCI   Type   InPkts   OutPkts   AAL/Encap     Status
  6      3     PVC    0        0         F4 OAM        ACTIVE
  7      4     PVC    0        0         F4 OAM        ACTIVE
TotalInPkts: 0, TotalOutPkts: 0, TotalInFast: 0, TotalOutFast: 0,
TotalBroadcasts: 0 TotalInPktDrops: 0, TotalOutPktDrops: 0
Local MNCP: 5, average number of cells received: 3
```

```
Peer MNCP: 1, average number of cells sent: 1
Local MCPT: 100 us
```

Troubleshooting Tips

To debug ATM cell packing, issue the **debug atm cell-packing** command.

Configuring Ethernet over MPLS in VLAN Mode

A VLAN is a switched network that is logically segmented by functions, project teams, or applications regardless of the physical location of users. Ethernet over MPLS allows you to connect two VLAN networks that are in different locations. You configure the PE routers at each end of the MPLS backbone and add a point-to-point VC. Only the two PE routers at the ingress and egress points of the MPLS backbone know about the VCs dedicated to transporting Layer 2 VLAN traffic. All other routers do not have table entries for those VCs. Ethernet over MPLS in VLAN mode transports Ethernet traffic from a source 802.1Q VLAN to a destination 802.1Q VLAN over a core MPLS network.

Note You must configure Ethernet over MPLS (VLAN mode) on the subinterfaces.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface gigabitethernet** *slot /interface.subinterface*
4. **encapsulation dot1q** *vlan-id*
5. **xconnect** *peer-router-id vcid* **encapsulation mpls**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**interface gigabitethernet** *slot* / *interface.subinterface* **Example:** `Router(config)# interface gigabitethernet4/0.1`	Specifies the Gigabit Ethernet subinterface and enters subinterface configuration mode. • Make sure the subinterface on the adjoining CE router is on the same VLAN as this PE router.
Step 4	**encapsulation dot1q** *vlan-id* **Example:** `Router(config-subif)# encapsulation dot1q 100`	Enables the subinterface to accept 802.1Q VLAN packets. • The subinterfaces between the CE and PE routers that are running Ethernet over MPLS must be in the same subnet. All other subinterfaces and backbone routers do not.
Step 5	**xconnect** *peer-router-id vcid* **encapsulation mpls** **Example:** `Router(config-subif)# xconnect 10.0.0.1 123 encapsulation mpls`	Binds the attachment circuit to a pseudowire VC. • The syntax for this command is the same as for all other Layer 2 transports.

Step 6	**end**
	Example:
	`Router(config-if-atm-l2trans-pvc)# end`
		Exits L2transport PVC configuration mode and returns to privileged EXEC mode.

Configuring Ethernet over MPLS in Port Mode

Port mode allows a frame coming into an interface to be packed into an MPLS packet and transported over the MPLS backbone to an egress interface. The entire Ethernet frame without the preamble or FCS is transported as a single packet. To configure port mode, use the **xconnect** command in interface configuration mode and specify the destination address and the VC ID. The syntax of the **xconnect** command is the same as for all other transport types. Each interface is associated with one unique pseudowire VC label.

When configuring Ethernet over MPLS in port mode, use the following guidelines:

- The pseudowire VC type is set to Ethernet.
- Port mode and Ethernet VLAN mode are mutually exclusive. If you enable a main interface for port-to-port transport, you cannot also enter commands on a subinterface.
- In Cisco IOS Release 12.2(33)SRE and later releases, L2VPN Routed Interworking using Ethernet over MPLS (EOMPLS) is no longer supported. When you configure the **interworking ip** command in pseudowire configuration mode, the **xconnect** command is disabled. To configure L2VPN Routed Interworking, use either Ethernet over MPLS (EOMPLS) or SVI (Switched Virtual Interface) based EOMPLS.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface gigabitethernet** *slot/interface*
4. **xconnect** *peer-router-id vcid* **encapsulation mpls**
5. **exit**
6. **exit**
7. **show mpls l2transport vc**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface gigabitethernet** *slot/interface* **Example:** `Router(config)# interface gigabitethernet4/0`	Specifies the Gigabit Ethernet interface and enters interface configuration mode. • Make sure the interface on the adjoining CE router is on the same VLAN as this PE router.
Step 4	**xconnect** *peer-router-id vcid* **encapsulation mpls** **Example:** `Router(config-if)# xconnect 10.0.0.1 123 encapsulation mpls`	Binds the attachment circuit to a pseudowire VC. • The syntax for this command is the same as for all other Layer 2 transports.
Step 5	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode.

	Command or Action	Purpose
Step 6	**exit** **Example:** `Router(config)# exit`	Exits router configuration mode.
Step 7	**show mpls l2transport vc** **Example:** `Router# show mpls l2transport vc`	Displays information about Ethernet over MPLS port mode.

Examples

In the following example, the output of the **show mpls l2transport vc detail**command is displayed:

```
Router# show mpls l2transport vc detail
Local interface: Gi4/0.1 up, line protocol up, Eth VLAN 2 up
Destination address: 10.1.1.1, VC ID: 2, VC status: up
.
.
.
Local interface: Gi8/0/1 up, line protocol up, Ethernet up
  Destination address: 10.1.1.1, VC ID: 8, VC status: up
```

Configuring Ethernet over MPLS with VLAN ID Rewrite

The VLAN ID rewrite feature enables you to use VLAN interfaces with different VLAN IDs at both ends of the tunnel.

The Cisco 12000 series router requires you to configure VLAN ID rewrite manually, as described in the following sections.

The following routers automatically perform VLAN ID rewrite on the disposition PE router. No configuration is required:

- Cisco 7200 series routers.
- Cisco 7500 series routers.
- Cisco 10720 series routers.
- Routers supported on Cisco IOS Release 12.4(11)T. (Use Cisco Feature Navigator to find information about platform support and Cisco IOS and Catalyst OS software image support.)

The following sections explain how to configure the VLAN ID rewrite feature:

Configuring Ethernet over MPLS with VLAN ID Rewrite for Cisco 12k Routers for 12.0(29)S and Earlier Releases

Use the following guidelines for the VLAN ID rewrite feature for the Cisco 12000 series routers in Cisco IOS releases earlier than 12.0(29)S:

- The IP Service Engine (ISE) 4-port Gigabit Ethernet line card performs the VLAN ID rewrite on the disposition side at the edge-facing line card.

- The engine 2 3-port Gigabit Ethernet line card performs the VLAN ID rewrite on the imposition side at the edge-facing line card.

The VLAN ID rewrite functionality requires that both ends of the Ethernet over MPLS connections be provisioned with the same line cards. Make sure that both edge-facing ends of the virtual circuit use either the engine 2 or ISE Ethernet line card. The following example shows the system flow with the VLAN ID rewrite feature:

- The ISE 4-port Gigabit Ethernet line card:

Traffic flows from VLAN1 on CE1 to VLAN2 on CE2. As the frame reaches the edge-facing line card of the disposition router PE2, the VLAN ID in the dot1Q header changes to the VLAN ID assigned to VLAN2.

- The engine 2 3-port Gigabit Ethernet line card:

Traffic flows from VLAN1 on CE1 to VLAN2 on CE2. As the frame reaches the edge-facing line card of the imposition router PE1, the VLAN ID in the dot1Q header changes to the VLAN ID assigned to VLAN2.

For the Cisco 12000 series router engine 2 3-port Gigabit Ethernet line card, you must issue the **remote circuit id** command as part of the Ethernet over MPLS VLAN ID rewrite configuration.

Configuring Ethernet over MPLS with VLAN ID Rewrite for Cisco 12k Routers for 12.0(30)S and Later Releases

In Cisco IOS Release 12.0(30)S, the following changes to VLAN ID rewrite were implemented:

- The ISE 4-port Gigabit Ethernet line card can perform VLAN ID rewrite at both the imposition and disposition sides of the edge-facing router.
- The **remote circuit id** command is not required as part of the Ethernet over MPLS VLAN ID rewrite configuration, as long as both PE routers are running Cisco IOS Release 12.0(30)S. The VLAN ID rewrite feature is implemented automatically when you configure Ethernet over MPLS.
- The VLAN ID rewrite feature in Cisco IOS Release 12.0(30)S can interoperate with routers that are running earlier releases. If you have a PE router at one end of the circuit that is using an earlier Cisco IOS release and the **remote circuit id**command, the other PE can run Cisco IOS Release 12.0(30)S and still perform VLAN ID rewrite.
- You can mix the line cards on the PE routers, as shown in the following table

Table 6 Supported Line Cards for VLAN ID Rewrite Feature:

If PE1 Has These Line Cards	Then PE2 Can Use These Line Cards
Engine 2 3-port Gigabit Ethernet line card or ISE 4-port Gigabit Ethernet line card	Engine 2 3-port Gigabit Ethernet line card or ISE 4-port Gigabit Ethernet line card
ISE 4-port Gigabit Ethernet line card	Any Cisco 12000 series router line card

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface gigabitethernet** *slot /interface.subinterface*
4. **encapsulation dot1q** *vlan-id*
5. **xconnect** *peer-router-id vcid* **encapsulation mpls**
6. **remote circuit id** *remote-vlan-id*
7. **exit**
8. **exit**
9. **exit**
10. **show controllers eompls forwarding-table**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface gigabitethernet** *slot /interface.subinterface* **Example:** `Router(config)# interface gigabitethernet4/0.1`	Specifies the Gigabit Ethernet subinterface and enters subinterface configuration mode. • Make sure the subinterfaces between the CE and PE routers that are running Ethernet over MPLS are in the same subnet. All other subinterfaces and backbone routers do not need to be in the same subnet.
Step 4	**encapsulation dot1q** *vlan-id* **Example:** `Router(config-subif)# encapsulation dot1q 100`	Enables the subinterface to accept 802.1Q VLAN packets. • Make sure the subinterface on the adjoining CE router is on the same VLAN as this PE router.

	Command or Action	Purpose
Step 5	**xconnect** *peer-router-id vcid* **encapsulation mpls** **Example:** `Router(config-subif)# xconnect 10.0.0.1 123 encapsulation mpls`	Binds the attachment circuit to a pseudowire VC and enters xconnect configuration mode. • The syntax for this command is the same as for all other Layer 2 transports.
Step 6	**remote circuit id** *remote-vlan-id* **Example:** `Router(config-subif-xconn)# remote circuit id 101`	Enables you to use VLAN interfaces with different VLAN IDs at both ends of the tunnel. • This command is required only for the Cisco 12000 series router engine 2 3-port Gigabit Ethernet line card.
Step 7	**exit** **Example:** `Router(config-subif-xconn)# exit`	Exits xconnect configuration mode.
Step 8	**exit** **Example:** `Router(config-subif)# exit`	Exits subinterface configuration mode.
Step 9	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode.
Step 10	**show controllers eompls forwarding-table** **Example:** `Router# execute slot 0 show controllers eompls forwarding-table`	Displays information about VLAN ID rewrite.

Examples

On PE1

On PE2

The command output of the **show controllers eompls forwarding-table**command in the following example shows VLAN ID rewrite configured on the Cisco 12000 series routers with an engine 2 3-port

Gigabit Ethernet line card. In the following example, the bolded command output show the VLAN ID rewrite information.

```
Router# execute slot 0 show controllers eompls forwarding-table 0 2
Port # 0, VLAN-ID # 2, Table-index 2
EoMPLS configured: 1
tag_rew_ptr             = D001BB58
Leaf entry?     = 1
FCR index       = 20
           **tagrew_psa_addr    = 0006ED60
           **tagrew_vir_addr    = 7006ED60
           **tagrew_phy_addr    = F006ED60
        [0-7] loq 8800 mtu 4458  oq 4000 ai 3 oi 04019110 (encaps size 4)
        cw-size 4 vlanid-rew 3
        gather A30 (bufhdr size 32 EoMPLS (Control Word) Imposition profile 81)
        2 tag: 18 18
        counters 1182, 10 reported 1182, 10.
    Local OutputQ (Unicast):    Slot:2  Port:0  RED queue:0  COS queue:0
    Output Q (Unicast):         Port:0          RED queue:0  COS queue:0

Router# execute slot 0 show controllers eompls forwarding-table 0 3

Port # 0, VLAN-ID # 3, Table-index 3
EoMPLS configured: 1
tag_rew_ptr             = D0027B90
Leaf entry?     = 1
FCR index       = 20
           **tagrew_psa_addr    = 0009EE40
           **tagrew_vir_addr    = 7009EE40
           **tagrew_phy_addr    = F009EE40
        [0-7] loq 9400 mtu 4458  oq 4000 ai 8 oi 84000002 (encaps size 4)
        cw-size 4 vlanid-rew 2
        gather A30 (bufhdr size 32 EoMPLS (Control Word) Imposition profile 81)
        2 tag: 17 18
        counters 1182, 10 reported 1182, 10.
    Local OutputQ (Unicast):    Slot:5  Port:0  RED queue:0  COS queue:0
    Output Q (Unicast):         Port:0          RED queue:0  COS queue:0
```

Configuring per-Subinterface MTU for Ethernet over MPLS

Cisco IOS Release 12.2(33)SRC introduces the ability to specify MTU values in xconnect subinterface configuration mode. When you use xconnect subinterface configuration mode to set the MTU value, you establish a pseudowire connection for situations where the interfaces have different MTU values that cannot be changed.

If you specify an MTU value in xconnect subinterface configuration mode that is outside the range of supported MTU values (64 bytes to the maximum number of bytes supported by the interface), the command might be rejected. If you specify an MTU value that is out of range in xconnect subinterface configuration mode, the router enters the command in subinterface configuration mode.

Note Configuring the MTU value in xconnect subinterface configuration mode has the following restrictions:

- The following features do not support MTU values in xconnect subinterface configuration mode:
  - Layer 2 Tunnel Protocol Version 3 (L2TPv3)
  - Virtual Private LAN services (VPLS)
  - L2VPN Pseudowire Switching
- The MTU value can be configured in xconnect subinterface configuration mode only on the following interfaces and subinterfaces:
  - Ethernet
  - FastEthernet
  - Gigabit Ethernet
- The router uses an MTU validation process for remote VCs established through LDP, which compares the MTU value configured in xconnect subinterface configuration mode to the MTU value of the remote customer interface. If an MTU value has not been configured in xconnect subinterface configuration mode, then the validation process compares the MTU value of the local customer interface to the MTU value of the remote xconnect, either explicitly configured or inherited from the underlying interface or subinterface.
- When you configure the MTU value in xconnect subinterface configuration mode, the specified MTU value is not enforced by the dataplane. The dataplane enforces the MTU values of the interface (port mode) or subinterface (VLAN mode).
- Ensure that the interface MTU is larger than the MTU value configured in xconnect subinterface configuration mode. If the MTU value of the customer-facing subinterface is larger than the MTU value of the core-facing interface, traffic may not be able to travel across the pseudowire.

>

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface gigabitethernet** *slot /interface*
4. **mtu** *mtu-value*
5. **interface gigabitethernet** *slot /interface.subinterface*
6. **encapsulation dot1q** *vlan-id*
7. **xconnect** *peer-router-id vcid* **encapsulation mpls**
8. **mtu** *mtu-value*
9. **end**
10. **show mpls l2transport binding**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface gigabitethernet** *slot /interface* **Example:** `Router(config)# interface gigabitethernet4/0`	Specifies the Gigabit Ethernet interface and enters interface configuration mode.
Step 4	**mtu** *mtu-value* **Example:** `Router(config-if)# mtu 2000`	Specifies the MTU value for the interface. • The MTU value specified at the interface level can be inherited by a subinterface.
Step 5	**interface gigabitethernet** *slot /interface.subinterface* **Example:** `Router(config-if)# interface gigabitethernet4/0.1`	Specifies the Gigabit Ethernet subinterface and enters subinterface configuration mode. • Make sure the subinterface on the adjoining CE router is on the same VLAN as this PE router.
Step 6	**encapsulation dot1q** *vlan-id* **Example:** `Router(config-subif)# encapsulation dot1q 100`	Enables the subinterface to accept 802.1Q VLAN packets. • The subinterfaces between the CE and PE routers that are running Ethernet over MPLS must be in the same subnet. All other subinterfaces and backbone routers need not be.
Step 7	**xconnect** *peer-router-id vcid* **encapsulation mpls** **Example:** `Router(config-subif)# xconnect 10.0.0.1 123 encapsulation mpls`	Binds the attachment circuit to a pseudowire VC. • The syntax for this command is the same as for all other Layer 2 transports. Enters xconnect subinterface configuration mode.

	Command or Action	Purpose
Step 8	**mtu** *mtu-value* **Example:** `Router(config-if-xconn)# mtu 1400`	Specifies the MTU for the VC.

Step 9	**end**
	Example:
	`Router(config-if-xconn)# end`
		Exits xconnect subinterface configuration mode and returns to privileged EXEC mode.

Step 10	**show mpls l2transport binding** **Example:** `Router# show mpls l2transport binding`	Displays the MTU values assigned to the local and remote interfaces.

Configuring Frame Relay over MPLS with DLCI-to-DLCI Connections

Frame Relay over MPLS encapsulates Frame Relay PDUs in MPLS packets and forwards them across the MPLS network. For Frame Relay, you can set up data-link connection identifier (DLCI)-to-DLCI connections or port-to-port connections. With DLCI-to-DLCI connections, the PE routers manipulate the packet by removing headers, adding labels, and copying control word elements from the header to the PDU.

Perform this task to configure Frame Relay over MPLS with DLCI-to-DLCI connections.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **frame-relay switching**
4. **interface serial** *slot /port*
5. **encapsulation frame-relay** [**cisco** | **ietf**]
6. **frame-relay intf-type dce**
7. **exit**
8. **connect** *connection-name interface dlci* **l2transport**
9. **xconnect** *peer-router-id vcid* **encapsulation mpls**
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**frame-relay switching** **Example:** `Router(config)# frame-relay switching`	Enables PVC switching on a Frame Relay device.
Step 4	**interface serial** *slot* /*port* **Example:** `Router(config)# interface serial3/1`	Specifies a serial interface and enters interface configuration mode.
Step 5	**encapsulation frame-relay** [**cisco** | **ietf**] **Example:** `Router(config-if)# encapsulation frame-relay ietf`	Specifies Frame Relay encapsulation for the interface. • You can specify different types of encapsulations. You can set one interface to Cisco encapsulation and the other interface to IETF encapsulation.
Step 6	**frame-relay intf-type dce** **Example:** `Router(config-if)# frame-relay intf-type dce`	Specifies that the interface is a DCE switch. • You can also specify the interface to support Network-to-Network Interface (NNI) and DTE connections.
Step 7	**exit** **Example:** `Router(config-if)# exit`	Exits from interface configuration mode.

	Command or Action	Purpose
Step 8	**connect** *connection-name interface dlci* **l2transport** **Example:** `Router(config)# connect fr1 serial5/0 1000 l2transport`	Defines connections between Frame Relay PVCs and enters connect configuration mode. • Using the **l2transport** keyword specifies that the PVC will not be a locally switched PVC, but will be tunneled over the backbone network. • The *connection-name* argument is a text string that you provide. • The *interface* argument is the interface on which a PVC connection will be defined. • The *dlci*argument is the DLCI number of the PVC that will be connected.
Step 9	**xconnect** *peer-router-id vcid* **encapsulation mpls** **Example:** `Router(config-fr-pw-switching)# xconnect 10.0.0.1 123 encapsulation mpls`	Creates the VC to transport the Layer 2 packets. • In a DLCI-to DLCI connection type, Frame Relay over MPLS uses the **xconnect** command in connect configuration mode.

Step 10	**end**
	Example:
	`Router(config-fr-pw-switching)# end`
		Exits connect configuration mode and returns to privileged EXEC mode.

Configuring Frame Relay over MPLS with Port-to-Port Connections

Frame Relay over MPLS encapsulates Frame Relay PDUs in MPLS packets and forwards them across the MPLS network. For Frame Relay, you can set up DLCI-to-DLCI connections or port-to-port connections. With port-to-port connections, you use HDLC mode to transport the Frame Relay encapsulated packets. In HDLC mode, the whole HDLC packet is transported. Only the HDLC flags and FCS bits are removed. The contents of the packet are not used or changed, including the backward explicit congestion notification (BECN), forward explicit congestion notification (FECN) and discard eligibility (DE) bits.

Perform this task to set up Frame Relay port-to-port connections.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface serial** *slot* /*port*
4. **encapsulation hdlc**
5. **xconnect** *peer-router-id vcid* **encapsulation mpls**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface serial** *slot* /*port* **Example:** `Router(config)# interface serial5/0`	Specifies a serial interface and enters interface configuration mode.
Step 4	**encapsulation hdlc** **Example:** `Router(config-if)# encapsulation hdlc`	Specifies that Frame Relay PDUs will be encapsulated in HDLC packets.
Step 5	**xconnect** *peer-router-id vcid* **encapsulation mpls** **Example:** `Router(config-if)# xconnect 10.0.0.1 123 encapsulation mpls`	Creates the VC to transport the Layer 2 packets.

Step 6	**end**
	Example:
	`Router(config-if)# end`
		Exits interface configuration mode and enters privileged EXEC mode.

Configuring HDLC and PPP over MPLS

With HDLC over MPLS, the whole HDLC packet is transported. The ingress PE router removes only the HDLC flags and FCS bits. The contents of the packet are not used or changed.

With PPP over MPLS, the ingress PE router removes the flags, address, control field, and the FCS.

Note

The following restrictions pertain to the HDLC over MPLS feature:

- Asynchronous interfaces are not supported.
- You must configure HDLC over MPLS on router interfaces only. You cannot configure HDLC over MPLS on subinterfaces.

The following restrictions pertain to the PPP over MPLS feature:

- Zero hops on one router is not supported. However, you can have back-to-back PE routers.
- Asynchronous interfaces are not supported. The connections between the CE and PE routers on both ends of the backbone must have similar link layer characteristics. The connections between the CE and PE routers must both be synchronous.
- Multilink PPP (MLP) is not supported.
- You must configure PPP on router interfaces only. You cannot configure PPP on subinterfaces.

>

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface serial** *slot* /*port*
4. Do one of the following:
   - **encapsulation ppp**
   -
   - **encapsulation hdlc**
5. **xconnect** *peer-router-id vcid* **encapsulation mpls**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**interface serial** *slot /port* **Example:** `Router(config)# interface serial5/0`	Specifies a serial interface and enters interface configuration mode. • You must configure HDLC and PPP over MPLS on router interfaces only. You cannot configure HDLC over MPLS on subinterfaces.

Step 4	Do one of the following:
	• **encapsulation ppp**
	•
	• **encapsulation hdlc**
	Example:
	`Router(config-if)# encapsulation ppp`
	Example:
	`or`
	Example:
	Example:
	`Router(config-if)# encapsulation hdlc`
		Specifies HDLC or PPP encapsulation and enters connect configuration mode.

Step 5	**xconnect** *peer-router-id vcid* **encapsulation mpls** **Example:** `Router(config-fr-pw-switching)# xconnect 10.0.0.1 123 encapsulation mpls`	Creates the VC to transport the Layer 2 packets.
Step 6	**end**	Exits connect configuration mode and returns to privileged EXEC mode.

Configuring Tunnel Selection

The tunnel selection feature allows you to specify the path that traffic uses. You can specify either an MPLS TE tunnel or destination IP address or domain name server (DNS) name.

You also have the option of specifying whether the VCs should use the default path (the path LDP uses for signaling) if the preferred path is unreachable. This option is enabled by default; you must explicitly disable it.

You configure tunnel selection when you set up the pseudowire class. You enable tunnel selection with the **preferred-path** command. Then, you apply the pseudowire class to an interface that has been configured to transport AToM packets.

The following guidelines provide more information about configuring tunnel selection:

- The **preferred-path** command is available only if the pseudowire encapsulation type is MPLS.
- This tunnel selection feature is enabled when you exit from pseudowire mode.
- The selected path should be an LSP destined to the peer PE router.
- The selected tunnel must be an MPLS TE tunnel.
- If you select a tunnel, the tunnel tailend must be on the remote PE router.
- If you specify an IP address, that address must be the IP address of the loopback interface on the remote PE router. The address must have a /32 mask. There must be an LSP destined to that selected address. The LSP need not be a TE tunnel.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **pseudowire-class** *name*
4. **encapsulation mpls**
5. **preferred-path** {**interface tunnel** *tunnel-number* | **peer** {*ip-address* | *host-name*}} [**disable-fallback**]
6. **exit**
7. **interface** *slot* /*port*
8. **encapsulation** *encapsulation-type*
9. **xconnect** *peer-router-id vcid* **pw-class name**
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**pseudowire-class** *name* **Example:** `Router(config)# pseudowire-class ts1`	Establishes a pseudowire class with a name that you specify and enters pseudowire configuration mode.
Step 4	**encapsulation mpls** **Example:** `Router(config-pw-class)# encapsulation mpls`	Specifies the tunneling encapsulation. • For AToM, the encapsulation type is **mpls**.
Step 5	**preferred-path** {**interface tunnel** *tunnel-number* | **peer** {*ip-address* | *host-name*}} [**disable-fallback**] **Example:** `Router(config-pw-class)# preferred path peer 10.18.18.18`	Specifies the MPLS traffic engineering tunnel or IP address or hostname to be used as the preferred path.
Step 6	**exit** **Example:** `Router(config-pw-class)# exit`	Exits from pseudowire configuration mode.
Step 7	**interface** *slot* /*port* **Example:** `Router(config)# interface atm1/1`	Specifies an interface and enters interface configuration mode.
Step 8	**encapsulation** *encapsulation-type* **Example:** `Router(config-if)# encapsulation aal5`	Specifies the encapsulation for the interface.
Step 9	**xconnect** *peer-router-id vcid* **pw-class name** **Example:** `Router(config-if)# xconnect 10.0.0.1 123 pw-class ts1`	Binds the attachment circuit to a pseudowire VC.

	Command or Action	Purpose

Step 10	**end**
	Example:
	`Router(config-if)# end`
		Exits interface configuration mode and returns to Privileged EXEC mode.

Examples

In the following example, the **show mpls l2transport vc** command shows the following information about the VCs:

- VC 101 has been assigned a preferred path called Tunnel1. The default path is disabled, because the preferred path specified that the default path should not be used if the preferred path fails.
- VC 150 has been assigned an IP address of a loopback address on PE2. The default path can be used if the preferred path fails.

In the following example, command output that is bolded shows the preferred path information.

```
Router# show mpls l2transport vc detail
Local interface: Gi0/0/0.1 up, line protocol up, Eth VLAN 222 up
  Destination address: 10.16.16.16, VC ID: 101, VC status: up
    Preferred path: Tunnel1,  active
    Default path: disabled
    Tunnel label: 3, next hop point2point
    Output interface: Tu1, imposed label stack {17 16}
  Create time: 00:27:31, last status change time: 00:27:31
  Signaling protocol: LDP, peer 10.16.16.16:0 up
    MPLS VC labels: local 25, remote 16
    Group ID: local 0, remote 6
    MTU: local 1500, remote 1500
    Remote interface description:
  Sequencing: receive disabled, send disabled
  VC statistics:
    packet totals: receive 10, send 10
    byte totals:   receive 1260, send 1300
    packet drops:  receive 0, send 0
Local interface: AT1/0/0 up, line protocol up, ATM AAL5 0/50 up
  Destination address: 10.16.16.16, VC ID: 150, VC status: up
    Preferred path: 10.18.18.18, active
    Default path: ready
    Tunnel label: 3, next hop point2point
    Output interface: Tu2, imposed label stack {18 24}
  Create time: 00:15:08, last status change time: 00:07:37
  Signaling protocol: LDP, peer 10.16.16.16:0 up
    MPLS VC labels: local 26, remote 24
    Group ID: local 2, remote 0
    MTU: local 4470, remote 4470
    Remote interface description:
  Sequencing: receive disabled, send disabled
  VC statistics:
    packet totals: receive 0, send 0
    byte totals:   receive 0, send 0
    packet drops:  receive 0, send 0
```

- Troubleshooting Tips, page 69

Troubleshooting Tips

You can use the **debug mpls l2transport vc event**command to troubleshoot tunnel selection. For example, if the tunnel interface that is used for the preferred path is shut down, the default path is enabled. The **debug mpls l2transport vc event**command provides the following output:

```
AToM SMGR [10.2.2.2, 101]: Processing imposition update, vc_handle 62091860,
update_action 3, remote_vc_label 16
AToM SMGR [10.2.2.2, 101]: selected route no parent rewrite: tunnel not up
AToM SMGR [10.2.2.2, 101]: Imposition Programmed, Output Interface: Et3/2
```

Setting Experimental Bits with AToM

MPLS AToM uses the three experimental bits in a label to determine the queue of packets. You statically set the experimental bits in both the VC label and the LSP tunnel label, because the LSP tunnel label might be removed at the penultimate router. The following sections explain the transport-specific implementations of the EXP bits.

Note For information about setting EXP bits on the Cisco 12000 series router for Cisco IOS Release 12.0(30)S, see the AToM: L2 QoS feature module.

Note The following restrictions apply to ATM AAL5 over MPLS with EXP bits:

- ATM AAL5 over MPLS allows you to statically set the experimental bits.
- If you do not assign values to the experimental bits, the priority bits in the header's "tag control information" field are set to zero.
- On the Cisco 7500 series routers, distributed Cisco Express Forwarding must be enabled before you set the experimental bits.

The following restrictions apply to ATM Cell Relay over MPLS with EXP bits:

- ATM Cell Relay over MPLS allows you to statically set the experimental bits in VC, PVP, and port modes.
- If you do not assign values to the experimental bits, the priority bits in the header's "tag control information" field are set to zero.
- On the Cisco 7500 series routers, distributed Cisco Express Forwarding must be enabled before you set the experimental bits.

The following restrictions apply to Ethernet over MPLS with EXP bits:

On the Cisco 7200 and 7500 Series Routers

- Ethernet over MPLS allows you to set the EXP bits by using either of the following methods:
  - Writing the priority bits into the experimental bit field, which is the default.
  - Using the **match any**command with the **set mpls exp** command.
- If you do not assign values to the experimental bits, the priority bits in the 802.1Q header's "tag control information" field are written into the experimental bit fields.
- On the Cisco 7500 series routers, distributed Cisco Express Forwarding must be enabled before you set the experimental bits.

On the Cisco 10720 Internet Router

The table below lists the commands that are supported on the Cisco 10720 Internet router for Ethernet over MPLS. The letter Y means that the command is supported on that interface. A dash (--) means that command is not supported on that interface.

Note The **match cos**command is supported only on subinterfaces, not main interfaces.

Table 7 Commands Supported on the Cisco 10720 Router for Ethernet over MPLS

Commands	Imposition	Disposition		
Traffic Matching Commands	In	Out	In	Out
match any	Y	Y	Y	Y
match cos	Y	--	--	--

Commands	Imposition	Disposition		
match input-interface	--	--	Y	Y
match mpls exp	--	Y	Y	--
match qos-group	--	Y	--	Y
Traffic Action Commands	In	Out	In	Out
set cos	--	--	--	Y
set mpls exp	Y	--	--	--
set qos-group	Y	--	Y	--
set srp-priority	--	Y	--	--

The following restrictions apply to Frame Relay over MPLS and EXP bits:

- If you do not assign values to the experimental bits, the priority bits in the header's “tag control information” field are set to zero.
- On the Cisco 7500 series routers, distributed Cisco Express Forwarding must be enabled before you set the experimental bits.

The following restrictions apply to HDLC over MPLS and PPP over MPLS and EXP bits:

- If you do not assign values to the experimental bits, zeros are written into the experimental bit fields.
- On the Cisco 7500 series routers, enable distributed Cisco Express Forwarding before setting the experimental bits.

Set the experimental bits in both the VC label and the LSP tunnel label. You set the experimental bits in the VC label, because the LSP tunnel label might be removed at the penultimate router. Perform this task to set the experimental bits.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map** *class-name*
4. **match any**
5. **exit**
6. **policy-map** *policy-name*
7. **class** *class-name*
8. **set mpls experimental** *value*
9. **exit**
10. **exit**
11. **interface** *slot* /*port*
12. **service-policy input** *policy-name*
13. **exit**
14. **exit**
15. **show policy-map interface** *interface-name* [**vc** [*vpi*/] *vci*] [**dlci** *dlci*] [**input** | **output**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

Step 3	**class-map** *class-name*
	Example:
	`Router(config)# class-map class1`
		Specifies the user-defined name of the traffic class and enters class map configuration mode.

Step 4	**match any** **Example:** `Router(config-cmap)# match any`	Specifies that all packets will be matched. • Use only the **any** keyword. Other keywords might cause unexpected results.

	Command or Action	Purpose
Step 5	**exit** **Example:** `Router(config-cmap)# exit`	Exits class map configuration mode.
Step 6	**policy-map** *policy-name* **Example:** `Router(config)# policy-map policy1`	Specifies the name of the traffic policy to configure and enters policy-map configuration mode.

Step 7	**class** *class-name*
	Example:
	`Router(config-pmap)# class class1`
		Specifies the name of the predefined traffic that was configured with the **class-map** command and was used to classify traffic to the traffic policy specified, and enters policy-map class configuration mode.

Step 8	**set mpls experimental** *value* **Example:** `Router(config-pmap-c)# set mpls experimental 7`	Designates the value to which the MPLS bits are set if the packets match the specified policy map.
Step 9	**exit** **Example:** `Router(config-pmap-c)# exit`	Exits policy-map class configuration mode.
Step 10	**exit** **Example:** `Router(config-pmap)# exit`	Exits policy-map configuration mode.
Step 11	**interface** *slot* /*port* **Example:** `Router(config)# interface atm4/0`	Specifies the interface and enters interface configuration mode.

	Command or Action	Purpose
Step 12	**service-policy input** *policy-name* **Example:** `Router(config-if)# service-policy input policy1`	Attaches a traffic policy to an interface.
Step 13	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode.
Step 14	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode.
Step 15	**show policy-map interface** *interface-name* [**vc** [*vpi*/] *vci*] [**dlci** *dlci*] [**input** | **output**] **Example:** `Router# show policy-map interface serial3/0`	Displays the traffic policy attached to an interface.

Setting the Frame Relay Discard Eligibility Bit on the Cisco 7200 and 7500 Series Routers

You can use the DE bit in the address field of a Frame Relay frame to prioritize frames in congested Frame Relay networks. The Frame Relay DE bit has only one bit and can therefore only have two settings, 0 or 1. If congestion occurs in a Frame Relay network, frames with the DE bit set to 1 are discarded before frames with the DE bit set to 0. Therefore, important traffic should have the DE bit set to 0, and less important traffic should be forwarded with the DE bit set at 1. The default DE bit setting is 0. You can change the DE bit setting to 1 with the **set fr-de** command.

Note The **set fr-de** command can be used only in an output service policy.

Perform this task to set the Frame Relay DE bit on the Cisco 7200 and 7500 series routers.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map** *policy-name*
4. **class** *class-name*
5. **set fr-de**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**policy-map** *policy-name* **Example:** `Router(config)# policy-map policy1`	Specifies the name of the traffic policy to configure and enters policy-map configuration mode. • Names can be a maximum of 40 alphanumeric characters.

Step 4	**class** *class-name*
	Example:
	`Router(config-pmap)# class class1`
		Specifies the name of a predefined traffic class and enters policy-map class configuration mode.

Step 5	**set fr-de** **Example:** `Router(config-pmap-c)# set fr-de`	Sets the Frame Relay DE bit setting for all packets that match the specified traffic class from 0 to 1.

	Command or Action	Purpose

Step 6	**end**
	Example:
	`Router(config-pmap-c)# end`
		Exits policy-map class configuration mode and returns to privileged EXEC mode.

Matching the Frame Relay DE Bit on the Cisco 7200 and 7500 Series Routers

You can use the **match fr-de**command to enable frames with a DE bit setting of 1 to be considered a member of a defined class and forwarded according to the specifications set in the service policy.

Perform this task to match frames with the FR DE bit set to 1.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map** *class-map-name*
4. **match fr-de**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

Step 3	**class-map** *class-map-name*
	Example:
	`Router(config)# class-map de-bits`
		Specifies the name of a predefined traffic class and enters class-map configuration mode.

	Command or Action	Purpose
Step 4	**match fr-de** **Example:** `Router(config-cmap)# match fr-de`	Classifies all frames with the DE bit set to 1.

Step 5	**end**
	Example:
	`Router(config-cmap)# end`
		Exits class-map configuration mode and returns to privileged EXEC mode.

Enabling the Control Word

You can enable the control word for dynamic and static pseudowires under a pseudowire class. Use the **control-word** command to enable, disable, or set a control word to autosense mode. If you do not enable a control word, autosense is the default mode for the control word.

Perform this task to enable a control word.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **pseudowire-class cw_enable**
4. **encapsulation mpls**
5. **control-word**
6. **exit**
7. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**pseudowire-class cw_enable** **Example:** `Router(config)# pseudowire-class cw_enable`	Enters pseudowire class configuration mode.
Step 4	**encapsulation mpls** **Example:** `Router(config-pw-class)# encapsulation mpls`	Specifies the tunneling encapsulation. • For AToM, the encapsulation type is mpls.
Step 5	**control-word** **Example:** `Router(config-pw-class)# control-word`	Enables the control word.
Step 6	**exit** **Example:** `Router(config-pw-class)# exit`	Exits pseudowire class configuration mode and returns to global configuration mode.
Step 7	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode.

Configuration Examples for Any Transport over MPLS

- Example ATM AAL5 over MPLS, page 80
- Example OAM Cell Emulation for ATM AAL5 over MPLS, page 80
- Example ATM Cell Relay over MPLS, page 81
- Example ATM Single Cell Relay over MPLS, page 82
- Example Ethernet over MPLS, page 84
- Example Tunnel Selection, page 84
- Example Setting Frame Relay Discard Eligibility Bit on the Cisco 7200 and 7500 Series Routers, page 86
- Example Matching Frame Relay DE Bit on the Cisco 7200 and 7500 Series Routers, page 87
- Example ATM over MPLS, page 87
- Example Ethernet over MPLS with MPLS Traffic Engineering Fast Reroute, page 88

- Example Configuring per-Subinterface MTU for Ethernet over MPLS, page 91
- Example Configuring MTU Values in xconnect Configuration Mode for L2VPN Interworking, page 93
- Example Removing a Pseudowire, page 95

Example ATM AAL5 over MPLS

ATM AAL5 over MPLS on PVCs

The following example shows how to enable ATM AAL5 over MPLS on an ATM PVC:

```
enable
 configure terminal
 interface atm1/
0
 pvc 1/
200 l2transport
 encapsulation aal5
 xconnect 10.13.13.13 100 encapsulation mpls
```

ATM AAL5 over MPLS in VC Class Configuration Mode

The following example shows how to configure ATM AAL5 over MPLS in VC class configuration mode. The VC class is then applied to an interface.

```
enable
configure terminal
vc-class atm aal5class
encapsulation aal5
interface atm1/
0
class-int aal5class
pvc 1/
200 l2transport
xconnect 10.13.13.13 100 encapsulation mpls
```

The following example shows how to configure ATM AAL5 over MPLS in VC class configuration mode. The VC class is then applied to a PVC.

```
enable
configure terminal
vc-class atm aal5class
encapsulation aal5
interface atm1/
0
pvc 1/
200 l2transport
class-vc aal5class
xconnect 10.13.13.13 100 encapsulation mpls
```

Example OAM Cell Emulation for ATM AAL5 over MPLS

OAM Cell Emulation for ATM AAL5 over MPLS on PVCs

The following example shows how to enable OAM cell emulation on an ATM PVC:

```
interface ATM 1/0/0
pvc 1/200 l2transport
encapsulation aal5
```

```
xconnect 10.13.13.13 100 encapsulation mpls
oam-ac emulation-enable
oam-pvc manage
```

The following example shows how to set the rate at which an AIS cell is sent every 30 seconds:

```
interface ATM 1/0/0
pvc 1/200 l2transport
encapsulation aal5
xconnect 10.13.13.13 100 encapsulation mpls
oam-ac emulation-enable 30
oam-pvc manage
```

OAM Cell Emulation for ATM AAL5 over MPLS in VC Class Configuration Mode

The following example shows how to configure OAM cell emulation for ATM AAL5 over MPLS in VC class configuration mode. The VC class is then applied to an interface.

```
enable
configure terminal
vc-class atm oamclass
encapsulation aal5
oam-ac emulation-enable 30
oam-pvc manage
interface atm1/0
class-int oamclass
pvc 1/200 l2transport
xconnect 10.13.13.13 100 encapsulation mpls
```

The following example shows how to configure OAM cell emulation for ATM AAL5 over MPLS in VC class configuration mode. The VC class is then applied to a PVC.

```
enable
configure terminal
vc-class atm oamclass
encapsulation aal5
oam-ac emulation-enable 30
oam-pvc manage
interface atm1/0
pvc 1/200 l2transport
class-vc oamclass
xconnect 10.13.13.13 100 encapsulation mpls
```

The following example shows how to configure OAM cell emulation for ATM AAL5 over MPLS in VC class configuration mode. The VC class is then applied to an interface. One PVC is configured with OAM cell emulation at an AIS rate of 10. That PVC uses the AIS rate of 10 instead of 30.

```
enable
configure terminal
vc-class atm oamclass
encapsulation aal5
oam-ac emulation-enable 30
oam-pvc manage
interface atm1/0
class-int oamclass
pvc 1/200 l2transport
oam-ac emulation-enable 10
xconnect 10.13.13.13 100 encapsulation mpls
```

Example ATM Cell Relay over MPLS

ATM Cell Relay over MPLS in VC Mode Using VC Class Configuration Mode

The following example shows how to configure ATM cell relay over MPLS in VC class configuration mode. The VC class is then applied to an interface.

```
enable
configure terminal
vc-class atm cellrelay
encapsulation aal0
interface atm1/0
class-int cellrelay
pvc 1/200 l2transport
xconnect 10.13.13.13 100 encapsulation mpls
```

The following example shows how to configure ATM cell relay over MPLS in VC class configuration mode. The VC class is then applied to a PVC.

```
enable
configure terminal
vc-class atm cellrelay
encapsulation aal0
interface atm1/0
pvc 1/200 l2transport
class-vc cellrelay
```

xconnect 10.13.13.13 100 encapsulation mpls

ATM Cell Relay over MPLS in PVP Mode

The following example shows how to transport single ATM cells over a virtual path:

```
pseudowire-class vp-cell-relay
encapsulation mpls
interface atm 5/0
atm pvp 1 l2transport
xconnect 10.0.0.1 123 pw-class vp-cell-relay
```

ATM Cell Relay over MPLS in Port Mode

The following example shows how to configure interface ATM 5/0 to transport ATM cell relay packets:

```
pseudowire-class atm-cell-relay
encapsulation mpls
interface atm 5/0
xconnect 10.0.0.1 123 pw-class atm-cell-relay
```

The following example shows how to configure interface ATM 9/0/0 to transport ATM cell relay packets on a Cisco 7600 series router, where you must specify the interface ATM slot, bay, and port:

```
pseudowire-class atm-cell-relay
encapsulation mpls
interface atm 9/0/0
xconnect 10.0.0.1 500 pw-class atm-cell-relay
```

Example ATM Single Cell Relay over MPLS

ATM Packed Cell Relay over MPLS in VC Mode

The following example shows that ATM PVC 1/100 is an AToM cell relay PVC. There are three timers set up, with values of 1000 milliseconds, 800 milliseconds, and 500 milliseconds, respectively. The **cell-**

packing command specifies that five ATM cells are to be packed into an MPLS packet. The **cell-packing** command also specifies that timer 1 is to be used.

```
interface atm 1/0
shutdown
atm mcpt-timer 1000 800 500
no shutdown
pvc 1/100 l2transport
encapsulation aal0
xconnect 10.0.0.1 123 encapsulation mpls
```

cell-packing 5 mcpt-timer 1

ATM Packed Cell Relay over MPLS in VC Mode Using VC Class Configuration Mode

The following example shows how to configure ATM cell relay over MPLS with cell packing in VC class configuration mode. The VC class is then applied to an interface.

```
enable
configure terminal
vc-class atm cellpacking
encapsulation aal0
cell-packing 10 mcpt-timer 1
interface atm1/0
shutdown
atm mcpt-timers 100 200 250
no shutdown
class-int cellpacking
pvc 1/200 l2transport
xconnect 10.13.13.13 100 encapsulation mpls
```

The following example shows how to configure ATM cell relay over MPLS in VC class configuration mode. The VC class is then applied to a PVC.

```
enable
configure terminal
vc-class atm cellpacking
encapsulation aal0
cell-packing 10 mcpt-timer 1
interface atm1/0
shutdown
atm mcpt-timers 100 200 250
no shutdown
pvc 1/200 l2transport
class-vc cellpacking
xconnect 10.13.13.13 100 encapsulation mpls
```

ATM Packed Cell Relay over MPLS in VP Mode

The following example shows packed cell relay enabled on an interface configured for PVP mode. The **cell-packing** command specifies that 10 ATM cells are to be packed into an MPLS packet. The **cell-packing** command also specifies that timer 2 is to be used.

```
interface atm 1/0
shutdown
atm mcpt-timer 1000 800 500
no shutdown
atm pvp 100 l2transport
xconnect 10.0.0.1 234 encapsulation mpls
cell-packing 10 mcpt-timer 2
```

ATM Packed Cell Relay over MPLS in Port Mode

The following example shows packed cell relay enabled on an interface set up for port mode. The **cell-packing** command specifies that 10 ATM cells are to be packed into an MPLS packet. The **cell-packing** command also specifies that timer 2 is to be used.

```
interface atm 5/0
shutdown
atm mcpt-timer 1000 800 500
no shutdown
cell-packing 10 mcpt-timer 2
xconnect 10.0.0.1 123 encapsulation mpls
```

Example Ethernet over MPLS

Ethernet over MPLS in Port Mode

The following example shows how to configure VC 123 in Ethernet port mode:

```
pseudowire-class ethernet-port
encapsulation mpls

int gigabitethernet1/0
xconnect 10.0.0.1 123 pw-class ethernet-port
```

Ethernet over MPLS with VLAN ID Rewrite

The following example shows how to configure VLAN ID rewrite on peer PE routers with Cisco 12000 series router engine 2 3-port Gigabit Ethernet line cards.

PE1	PE2

`interface GigabitEthernet0/0.2`
`encapsulation dot1Q 2`
`no ip directed-broadcast`
`no cdp enable`
`xconnect 10.5.5.5 2 encapsulation mpls`
`remote circuit id 3`
	`interface GigabitEthernet3/0.2`
	`encapsulation dot1Q 3`
	`no ip directed-broadcast`
	`no cdp enable`
	`xconnect 10.3.3.3 2 encapsulation mpls`
	`remote circuit id 2`

Example Tunnel Selection

The following example shows how to set up two preferred paths for PE1. One preferred path specifies an MPLS traffic engineering tunnel. The other preferred path specifies an IP address of a loopback address on PE2. There is a static route configured on PE1 that uses a TE tunnel to reach the IP address on PE2.

PE1 Configuration

```
mpls label protocol ldp
mpls traffic-eng tunnels
tag-switching tdp router-id Loopback0
pseudowire-class pw1
 encapsulation mpls
 preferred-path interface Tunnel1 disable-fallback
!
pseudowire-class pw2
 encapsulation mpls
 preferred-path peer 10.18.18.18
!
```

```
interface Loopback0
 ip address 10.2.2.2 255.255.255.255
 no ip directed-broadcast
 no ip mroute-cache
!
interface Tunnel1
 ip unnumbered Loopback0
 no ip directed-broadcast
 tunnel destination 10.16.16.16
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng priority 7 7
 tunnel mpls traffic-eng bandwidth 1500
 tunnel mpls traffic-eng path-option 1 explicit name path-tu1
!
interface Tunnel2
 ip unnumbered Loopback0
 no ip directed-broadcast
 tunnel destination 10.16.16.16
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng priority 7 7
 tunnel mpls traffic-eng bandwidth 1500
 tunnel mpls traffic-eng path-option 1 dynamic
!
interface gigabitethernet0/0/0
 no ip address
 no ip directed-broadcast
 no negotiation auto
!
interface gigabitethernet0/0/0.1
 encapsulation dot1Q 222
 no ip directed-broadcast
 xconnect 10.16.16.16 101 pw-class pw1
!
interface ATM1/0/0
 no ip address
 no ip directed-broadcast
 no atm enable-ilmi-trap
 no atm ilmi-keepalive
 pvc 0/50 l2transport
  encapsulation aal5
  xconnect 10.16.16.16 150 pw-class pw2
!
interface Ethernet2/0/1
 ip address 10.0.0.1 255.255.255.0
 no ip directed-broadcast
 tag-switching ip
 mpls traffic-eng tunnels
 ip rsvp bandwidth 15000 15000
!
router ospf 1
 log-adjacency-changes
 network 10.0.0.0 0.0.0.255 area 0
 network 10.2.2.2 0.0.0.0 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
ip route 10.18.18.18 255.255.255.255 Tunnel2
!
ip explicit-path name path-tu1 enable
 next-address 10.0.0.1
 index 3 next-address 10.0.0.1
```

PE2 Configuration

```
mpls label protocol ldp
mpls traffic-eng tunnels
mpls ldp router-id Loopback0
interface Loopback0
 ip address 10.16.16.16 255.255.255.255
 no ip directed-broadcast
 no ip mroute-cache
!
```

```
interface Loopback2
 ip address 10.18.18.18 255.255.255.255
 no ip directed-broadcast
!
interface Ethernet3/1
 ip address 10.0.0.2 255.255.255.0
 no ip directed-broadcast
 mpls traffic-eng tunnels
 mpls ip
 no cdp enable
 ip rsvp bandwidth 15000 15000
!
interface Ethernet3/3
 no ip address
 no ip directed-broadcast
 no cdp enable
!
interface Ethernet3/3.1
 encapsulation dot1Q 222
 no ip directed-broadcast
 no cdp enable
 mpls l2transport route 10.2.2.2 101
!
interface ATM5/0
 no ip address
 no ip directed-broadcast
 no atm enable-ilmi-trap
 no atm ilmi-keepalive
 pvc 0/50 l2transport
  encapsulation aal5
  xconnect 10.2.2.2 150 encapsulation mpls
!
router ospf 1
 log-adjacency-changes
 network 10.0.0.0 0.0.0.255 area 0
 network 10.16.16.16 0.0.0.0 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
```

Example Setting Frame Relay Discard Eligibility Bit on the Cisco 7200 and 7500 Series Routers

The following example shows how to configure the service policy called set-de and attach it to an interface. In this example, the class map called data evaluates all packets exiting the interface for an IP precedence value of 1. If the exiting packet has been marked with the IP precedence value of 1, the packet’s DE bit is set to 1.

```
class-map data
match ip precedence 1
policy-map set-de
class data
set fr-de
interface Serial0/0/0
encapsulation frame-relay
interface Serial0/0/0.1 point-to-point
ip address 192.168.249.194 255.255.255.252
frame-relay interface-dlci 100
service output set-de
```

Example Matching Frame Relay DE Bit on the Cisco 7200 and 7500 Series Routers

The following example shows how to configure the service policy called match-de and attach it to an interface. In this example, the class map called data evaluates all packets entering the interface for a DE bit setting of 1. If the entering packet has been a DE bit value of 1, the packet’s EXP bit setting is set to 3.

```
class-map data
match fr-de
policy-map match-de
class data
set mpls exp 3
ip routing
ip cef distributed
mpls label protocol ldp
interface Loopback0
 ip address 10.20.20.20 255.255.255.255
interface Ethernet1/0/0
 ip address 10.0.0.2 255.255.255.0
 mpls ip
interface Serial4/0/0
 encapsulation frame-relay
service input match-de
connect 100 Serial4/0/0 100 l2transport
```

xconnect 10.10.10.10 100 encapsulation mpls

Example ATM over MPLS

The table below shows the configuration of ATM over MPLS on two PE routers.

Table 8 ATM over MPLS Configuration Example

PE1	PE2
`mpls label protocol ldp`	`mpls label protocol ldp`
` mpls ldp router-id Loopback0 force`	` mpls ldp router-id Loopback0 force`
`!`	`!`
`interface Loopback0`	`interface Loopback0`
` ip address 10.16.12.12 255.255.255.255`	` ip address 10.13.13.13 255.255.255.255`
`!`	
`interface ATM4/0`	`interface ATM4/0`
` pvc 0/100 l2transport`	`  pvc 0/100 l2transport`
`   encapsulation aal0`	`    encapsulation aal0`
`   xconnect 10.13.13.13 100 encapsulation mpls`	`    xconnect 10.16.12.12 100 encapsulation mpls`
`!`	`!`
`interface ATM4/0.300 point-to-point`	`interface ATM4/0.300 point-to-point`
` no ip directed-broadcast`	` no ip directed-broadcast`
` no atm enable-ilmi-trap`	` no atm enable-ilmi-trap`
` pvc 0/300 l2transport`	` pvc 0/300 l2transport`
`   encapsulation aal0`	`   encapsulation aal0`
`   xconnect 10.13.13.13 300 encapsulation mpls`	`   xconnect 10.16.12.12 300 encapsulation mpls`

Example Ethernet over MPLS with MPLS Traffic Engineering Fast Reroute

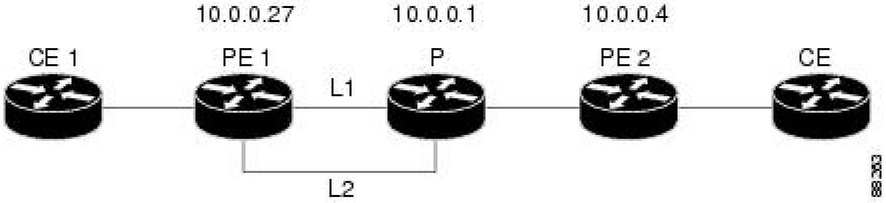
The following configuration example and the figure below show the configuration of Ethernet over MPLS with fast reroute on AToM PE routers.

Routers PE1 and PE2 have the following characteristics:

- A TE tunnel called Tunnel41 is configured between PE1and PE2, using an explicit path through a link called L1. AToM VCs are configured to travel through the FRR-protected tunnel Tunnel41.
- The link L1 is protected by FRR, the backup tunnel is Tunnel1.

- PE2 is configured to forward the AToM traffic back to PE1 through the L2 link.

Figure 2 Fast Reroute Configuration



PE1 Configuration

```
mpls label protocol ldp
mpls traffic-eng tunnels
mpls ldp router-id Loopback1 force
!
pseudowire-class T41
 encapsulation mpls
 preferred-path interface Tunnel41 disable-fallback
!
pseudowire-class IP1
 encapsulation mpls
 preferred-path peer 10.4.0.1 disable-fallback
!
interface Loopback1
 ip address 10.0.0.27 255.255.255.255
!
interface Tunnel1
 ip unnumbered Loopback1
 tunnel destination 10.0.0.1
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 10000
 tunnel mpls traffic-eng path-option 1 explicit name FRR
!
interface Tunnel41
 ip unnumbered Loopback1
 tunnel destination 10.0.0.4
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 1000
 tunnel mpls traffic-eng path-option 1 explicit name name-1
 tunnel mpls traffic-eng fast-reroute
!
interface POS0/0
 description pe1name POS8/0/0
 ip address 10.1.0.2 255.255.255.252
 mpls traffic-eng tunnels
 mpls traffic-eng backup-path Tunnel1
 crc 16
 clock source internal
 pos ais-shut
 pos report lrdi
 ip rsvp bandwidth 155000 155000
!
interface POS0/3
 description pe1name POS10/1/0
 ip address 10.1.0.14 255.255.255.252
 mpls traffic-eng tunnels
 crc 16
 clock source internal
 ip rsvp bandwidth 155000 155000
!
interface gigabitethernet3/0.1
 encapsulation dot1Q 203
 xconnect 10.0.0.4 2 pw-class IP1
!
```

```
interface gigabitethernet3/0.2
 encapsulation dot1Q 204
 xconnect 10.0.0.4 4 pw-class T41
!
router ospf 1
 network 10.0.0.0 0.255.255.255 area 0
 mpls traffic-eng router-id Loopback1
 mpls traffic-eng area 0
!
ip classless
ip route 10.4.0.1 255.255.255.255 Tunnel41
!
ip explicit-path name xxxx-1 enable
 next-address 10.4.1.2
 next-address 10.1.0.10
```

P Configuration

```
ip cef
mpls traffic-eng tunnels
!
interface Loopback1
 ip address 10.0.0.1 255.255.255.255
!
interface FastEthernet1/0/0
 ip address 10.4.1.2 255.255.255.0
 mpls traffic-eng tunnels
 ip rsvp bandwidth 10000 10000
!
interface POS8/0/0
 description xxxx POS0/0
 ip address 10.1.0.1 255.255.255.252
 mpls traffic-eng tunnels
 pos ais-shut
 pos report lrdi
 ip rsvp bandwidth 155000 155000
!
interface POS10/1/0
 description xxxx POS0/3
 ip address 10.1.0.13 255.255.255.252
 mpls traffic-eng tunnels
 ip rsvp bandwidth 155000 155000
!
router ospf 1
 network 10.0.0.0 0.255.255.255 area 0
 mpls traffic-eng router-id Loopback1
 mpls traffic-eng area 0
```

PE2 Configuration

```
ip cef
mpls label protocol ldp
mpls traffic-eng tunnels
mpls ldp router-id Loopback1 force
!
interface Loopback1
 ip address 10.0.0.4 255.255.255.255
!
interface loopback 2
ip address 10.4.0.1 255.255.255.255
!
interface Tunnel27
 ip unnumbered Loopback1
 tunnel destination 10.0.0.27
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 1000
 tunnel mpls traffic-eng path-option 1 explicit name xxxx-1
!
```

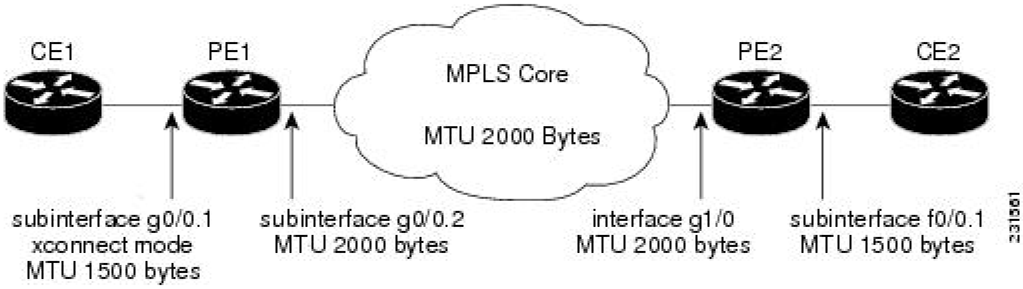
```
interface FastEthernet0/0.2
 encapsulation dot1Q 203
 xconnect 10.0.0.27 2 encapsulation mpls
!
interface FastEthernet0/0.3
 encapsulation dot1Q 204
 xconnect 10.0.0.27 4 encapsulation mpls
!
interface FastEthernet1/1
 ip address 10.4.1.1 255.255.255.0
 mpls traffic-eng tunnels
 ip rsvp bandwidth 10000 10000
!
router ospf 1
 network 10.0.0.0 0.255.255.255 area 0
 mpls traffic-eng router-id Loopback1
 mpls traffic-eng area 0
!
ip explicit-path name xxxx-1 enable
 next-address 10.4.1.2
 next-address 10.1.0.10
```

Example Configuring per-Subinterface MTU for Ethernet over MPLS

The figure below shows a configuration that enables matching MTU values between VC endpoints.

As shown in the figure below, PE1 is configured in xconnect subinterface configuration mode with an MTU value of 1500 bytes in order to establish an end-to-end VC with PE2, which also has an MTU value of 1500 bytes. If PE1 was not set with an MTU value of 1500 bytes, in xconnect subinterface configuration mode, the subinterface would inherit the MTU value of 2000 bytes set on the interface. This would cause a mismatch in MTU values between the VC endpoints, and the VC would not come up.

Figure 3 Configuring MTU Values in xconnect Subinterface Configuration Mode



The following examples show the router configurations in the figure above:

CE1 Configuration

```
interface gigabitethernet0/0
 mtu 1500
 no ip address
!
interface gigabitethernet0/0.1
 encapsulation dot1Q 100
 ip address 10.181.182.1 255.255.255.0
```

PE1 Configuration

```
interface gigabitethernet0/0
 mtu 2000
 no ip address
```

```
!
interface gigabitethernet0/0.1
 encapsulation dot1Q 100
 xconnect 10.1.1.152 100 encapsulation mpls
  mtu 1500
!
interface gigabitethernet0/0.2
 encapsulation dot1Q 200
 ip address 10.151.100.1 255.255.255.0
 mpls ip
```

PE2 Configuration

```
interface gigabitethernet1/0
 mtu 2000
 no ip address
!
interface gigabitethernet1/0.2
 encapsulation dot1Q 200
 ip address 10.100.152.2 255.255.255.0
 mpls ip
!
interface fastethernet0/0
 no ip address
!
interface fastethernet0/0.1
 description default MTU of 1500 for FastEthernet
 encapsulation dot1Q 100
 xconnect 10.1.1.151 100 encapsulation mpls
```

CE2 Configuration

```
interface fastethernet0/0
 no ip address
interface fastethernet0/0.1
 encapsulation dot1Q 100
 ip address 10.181.182.2 255.255.255.0
```

The **show mpls l2transport binding**command, issued from router PE1, shows a matching MTU value of 1500 bytes on both the local and remote routers:

```
Router# show mpls l2transport binding
Destination Address: 10.1.1.152,  VC ID: 100
    Local Label: 100
        Cbit: 1,    VC Type: Ethernet,    GroupID: 0
        MTU: 1500,   Interface Desc: n/a
        VCCV: CC Type: CW [1], RA [2]
              CV Type: LSPV [2]
    Remote Label: 202
        Cbit: 1,    VC Type: Ethernet,    GroupID: 0
        MTU: 1500,   Interface Desc: n/a
        VCCV: CC Type: RA [2]
              CV Type: LSPV [2]


Router# show mpls l2transport vc detail
Local interface: Gi0/0.1 up, line protocol up, Eth VLAN 100 up
  Destination address: 10.1.1.152, VC ID: 100, VC status: up
    Output interface: Gi0/0.2, imposed label stack {202}
    Preferred path: not configured
    Default path: active
    Next hop: 10.151.152.2
  Create time: 1d11h, last status change time: 1d11h
  Signaling protocol: LDP, peer 10.1.1.152:0 up
    Targeted Hello: 10.1.1.151(LDP Id) -> 10.1.1.152
    MPLS VC labels: local 100, remote 202
    Group ID: local 0, remote 0
    MTU: local 1500, remote 1500
    Remote interface description:
```

```
  Sequencing: receive disabled, send disabled
  VC statistics:
    packet totals: receive 41, send 39
    byte totals:   receive 4460, send 5346
    packet drops:  receive 0, send 0
```

In the following example, you are specifying an MTU of 1501 in xconnect subinterface configuration mode, and that value is out of range, the router enters the command in subinterface configuration mode, where it is accepted:

```
Router# configure terminal
router(config)# interface gigabitethernet0/2.1
router(config-subif)# xconnect 10.10.10.1 100 encapsulation mpls
router(config-subif-xconn)# mtu ?
<64 - 1500> MTU size in bytes
router(config-subif-xconn)# mtu 1501
router(config-subif)# mtu ?
<64 - 17940> MTU size in bytes
```

If the MTU value is not accepted in either xconnect subinterface configuration mode or subinterface configuration mode, then the command is rejected, as shown in the following example:

```
Router# configure terminal
router(config)# interface gigabitethernet0/2.1
router(config-subif)# xconnect 10.10.10.1 100 encapsulation mpls
router(config-subif-xconn)# mtu ?
<64 - 1500> MTU size in bytes
router(config-subif-xconn)# mtu 63
% Invalid input detected at ^ marker
```

Example Configuring MTU Values in xconnect Configuration Mode for L2VPN Interworking

The following example shows an L2VPN Interworking example. The PE1 router has a serial interface configured with an MTU value of 1492 bytes. The PE2 router uses xconnect configuration mode to set a matching MTU of 1492 bytes, which allows the two routers to form an interworking VC. If the PE2 router did not set the MTU value in xconnect configuration mode, the interface would be set to 1500 bytes by default and the VC would not come up.

PE1 Configuration

```
pseudowire-class atom-ipiw
 encapsulation mpls
 interworking ip
!
interface Loopback0
 ip address 10.1.1.151 255.255.255.255
!
interface Serial2/0
 mtu 1492
 no ip address
 encapsulation ppp
 no fair-queue
 serial restart-delay 0
 xconnect 10.1.1.152 123 pw-class atom-ipiw
!
interface Serial4/0
 ip address 10.151.100.1 255.255.255.252
 encapsulation ppp
 mpls ip
 serial restart-delay 0
!
router ospf 1
 log-adjacency-changes
 network 10.1.1.151 0.0.0.0 area 0
 network 10.151.100.0 0.0.0.3 area 0
!
mpls ldp router-id Loopback0
```

PE2 Configuration

```
pseudowire-class atom-ipiw
 encapsulation mpls
 interworking ip
!
interface Loopback0
 ip address 10.1.1.152 255.255.255.255
!
interface Ethernet0/0
 no ip address
 xconnect 10.1.1.151 123 pw-class atom-ipiw
  mtu 1492
!
interface Serial4/0
 ip address 10.100.152.2 255.255.255.252
 encapsulation ppp
 mpls ip
 serial restart-delay 0
!
router ospf 1
 log-adjacency-changes
 network 10.1.1.152 0.0.0.0 area 0
 network 10.100.152.0 0.0.0.3 area 0
!
mpls ldp router-id Loopback0
```

The **show mpls l2transport binding**command shows that the MTU value for the local and remote routers is 1492 bytes.

PE1 Configuration

```
Router# show mpls l2transport binding

Destination Address: 10.1.1.152,  VC ID: 123
    Local Label: 105
        Cbit: 1,    VC Type: PPP,    GroupID: 0
        MTU: 1492,   Interface Desc: n/a
        VCCV: CC Type: CW [1], RA [2]
              CV Type: LSPV [2]
    Remote Label: 205
        Cbit: 1,    VC Type: Ethernet,    GroupID: 0
        MTU: 1492,   Interface Desc: n/a
        VCCV: CC Type: RA [2]
              CV Type: LSPV [2]
Router# show mpls l2transport vc detail
Local interface: Se2/0 up, line protocol up, PPP up
  MPLS VC type is PPP, interworking type is IP
  Destination address: 10.1.1.152, VC ID: 123, VC status: up
    Output interface: Se4/0, imposed label stack {1003 205}
    Preferred path: not configured
    Default path: active
    Next hop: point2point
  Create time: 00:25:29, last status change time: 00:24:54
  Signaling protocol: LDP, peer 10.1.1.152:0 up
    Targeted Hello: 10.1.1.151(LDP Id) -> 10.1.1.152
    Status TLV support (local/remote)   : enabled/supported
      Label/status state machine        : established, LruRru
      Last local dataplane   status rcvd: no fault
      Last local SSS circuit status rcvd: no fault
      Last local SSS circuit status sent: no fault
      Last local  LDP TLV    status sent: no fault
      Last remote LDP TLV    status rcvd: no fault
    MPLS VC labels: local 105, remote 205
    Group ID: local n/a, remote 0
    MTU: local 1492, remote 1492
    Remote interface description:
  Sequencing: receive disabled, send disabled
  VC statistics:
    packet totals: receive 30, send 29
    byte totals:   receive 2946, send 3364
    packet drops:  receive 0, send 0
```

PE2 Configuration

```
Router# show mpls l2transport binding

Destination Address: 10.1.1.151,  VC ID: 123
    Local Label: 205
        Cbit: 1,    VC Type: Ethernet,    GroupID: 0
        MTU: 1492,   Interface Desc: n/a
        VCCV: CC Type: RA [2]
              CV Type: LSPV [2]
    Remote Label: 105
        Cbit: 1,    VC Type: Ethernet,    GroupID: 0
        MTU: 1492,   Interface Desc: n/a
        VCCV: CC Type: CW [1], RA [2]
              CV Type: LSPV [2]
Router# show mpls l2transport vc detail
Local interface: Et0/0 up, line protocol up, Ethernet up
  MPLS VC type is Ethernet, interworking type is IP
  Destination address: 10.1.1.151, VC ID: 123, VC status: up
    Output interface: Se4/0, imposed label stack {1002 105}
    Preferred path: not configured
    Default path: active
    Next hop: point2point
  Create time: 00:25:19, last status change time: 00:25:19
  Signaling protocol: LDP, peer 10.1.1.151:0 up
    Targeted Hello: 10.1.1.152(LDP Id) -> 10.1.1.151
    Status TLV support (local/remote)   : enabled/supported
      Label/status state machine        : established, LruRru
      Last local dataplane   status rcvd: no fault
      Last local SSS circuit status rcvd: no fault
      Last local SSS circuit status sent: no fault
      Last local  LDP TLV    status sent: no fault
      Last remote LDP TLV    status rcvd: no fault
    MPLS VC labels: local 205, remote 105
    Group ID: local n/a, remote 0
    MTU: local 1492, remote 1492
    Remote interface description:
  Sequencing: receive disabled, send disabled
  VC statistics:
    packet totals: receive 29, send 30
    byte totals:   receive 2900, send 3426
    packet drops:  receive 0, send 0
```

Example Removing a Pseudowire

The following example shows how to remove all xconnects:

```
Router# clear xconnect all
02:13:56: Xconnect[ac:Et1/0.1(Eth VLAN)]: provisioning fwder with fwd_type=1, sss_role=1
02:13:56: Xconnect[mpls:10.1.1.2:1234000]: provisioning fwder with fwd_type=2, sss_role=2
02:13:56: Xconnect[ac:Et1/0.2(Eth VLAN)]: provisioning fwder with fwd_type=1, sss_role=2
02:13:56: Xconnect[mpls:10.1.1.2:1234001]: provisioning fwder with fwd_type=2, sss_role=1
02:13:56: Xconnect[ac:Et1/0.3(Eth VLAN)]: provisioning fwder with fwd_type=1, sss_role=1
02:13:56: Xconnect[mpls:10.1.2.2:1234002]: provisioning fwder with fwd_type=2, sss_role=2
02:13:56: Xconnect[ac:Et1/0.4(Eth VLAN)]: provisioning fwder with fwd_type=1, sss_role=2
02:13:56: Xconnect[mpls:10.1.2.2:1234003]: provisioning fwder with fwd_type=2, sss_role=1
02:13:56: MPLS peer 10.1.1.2 vcid 1234000, VC DOWN, VC state DOWN
02:13:56: MPLS peer 10.1.1.2 vcid 1234001, VC DOWN, VC state DOWN
02:13:56: MPLS peer 10.1.2.2 vcid 1234002, VC DOWN, VC state DOWN
02:13:56: MPLS peer 10.1.2.2 vcid 1234003, VC DOWN, VC state DOWN
02:13:56: XC AUTH [Et1/0.1, 1001]: Event: start xconnect authorization, state changed
```

```
from IDLE to AUTHORIZING
02:13:56: XC AUTH [Et1/0.1, 1001]: Event: found xconnect authorization, state changed
from AUTHORIZING to DONE
02:13:56: XC AUTH [Et1/0.3, 1003]: Event: start xconnect authorization, state changed
from IDLE to AUTHORIZING
02:13:56: XC AUTH [Et1/0.3, 1003]: Event: found xconnect authorization, state changed
from AUTHORIZING to DONE
02:13:56: XC AUTH [10.1.1.2, 1234001]: Event: start xconnect authorization, state changed
from IDLE to AUTHORIZING
02:13:56: XC AUTH [10.1.1.2, 1234001]: Event: found xconnect authorization, state changed
from AUTHORIZING to DONE
02:13:56: XC AUTH [10.1.2.2, 1234003]: Event: start xconnect authorization, state changed
from IDLE to AUTHORIZING
02:13:56: XC AUTH [10.1.2.2, 1234003]: Event: found xconnect authorization, state changed
from AUTHORIZING to DONE
02:13:56: XC AUTH [Et1/0.1, 1001]: Event: free xconnect authorization request, state
changed from DONE to END
02:13:56: XC AUTH [Et1/0.3, 1003]: Event: free xconnect authorization request, state
changed from DONE to END
02:13:56: XC AUTH [10.1.1.2, 1234001]: Event: free xconnect authorization request, state
changed from DONE to END
02:13:56: XC AUTH [10.1.2.2, 1234003]: Event: free xconnect authorization request, state
changed from DONE to END
02:13:56: MPLS peer 10.1.1.2 vcid 1234001, VC UP, VC state UP
02:13:56: MPLS peer 10.1.2.2 vcid 1234003, VC UP, VC state UP
02:13:56: MPLS peer 10.1.1.2 vcid 1234000, VC UP, VC state UP
02:13:56: MPLS peer 10.1.2.2 vcid 1234002, VC UP, VC state UP
```

The following example shows how to remove all the xconnects associated with peer router 10.1.1.2:

```
Router# clear xconnect peer 10.1.1.2 all
02:14:08: Xconnect[ac:Et1/0.1(Eth VLAN)]: provisioning fwder with fwd_type=1, sss_role=1
02:14:08: Xconnect[mpls:10.1.1.2:1234000]: provisioning fwder with fwd_type=2, sss_role=2
02:14:08: Xconnect[ac:Et1/0.2(Eth VLAN)]: provisioning fwder with fwd_type=1, sss_role=2
02:14:08: Xconnect[mpls:10.1.1.2:1234001]: provisioning fwder with fwd_type=2, sss_role=1
02:14:08: MPLS peer 10.1.1.2 vcid 1234000, VC DOWN, VC state DOWN
02:14:08: MPLS peer 10.1.1.2 vcid 1234001, VC DOWN, VC state DOWN
02:14:08: XC AUTH [Et1/0.1, 1001]: Event: start xconnect authorization, state changed
from IDLE to AUTHORIZING
02:14:08: XC AUTH [Et1/0.1, 1001]: Event: found xconnect authorization, state changed
from AUTHORIZING to DONE
02:14:08: XC AUTH [10.1.1.2, 1234001]: Event: start xconnect authorization, state changed
from IDLE to AUTHORIZING
02:14:08: XC AUTH [10.1.1.2, 1234001]: Event: found xconnect authorization, state changed
from AUTHORIZING to DONE
02:14:08: XC AUTH [Et1/0.1, 1001]: Event: free xconnect authorization request, state
changed from DONE to END
02:14:08: XC AUTH [10.1.1.2, 1234001]: Event: free xconnect authorization request, state
changed from DONE to END
02:14:08: MPLS peer 10.1.1.2 vcid 1234001, VC UP, VC state UP
02:14:08: MPLS peer 10.1.1.2 vcid 1234000, VC UP, VC state UP
```

The following example shows how to remove the xconnects associated with peer router 10.1.1.2 and VC ID 1234001:

```
Router# clear xconnect peer 10.1.1.2 vcid 1234001
02:14:23: Xconnect[ac:Et1/0.2(Eth VLAN)]: provisioning fwder with fwd_type=1, sss_role=1
02:14:23: Xconnect[mpls:10.1.1.2:1234001]: provisioning fwder with fwd_type=2, sss_role=2
02:14:23: MPLS peer 10.1.1.2 vcid 1234001, VC DOWN, VC state DOWN
02:14:23: XC AUTH [Et1/0.2, 1002]: Event: start xconnect authorization, state changed
from IDLE to AUTHORIZING
02:14:23: XC AUTH [Et1/0.2, 1002]: Event: found xconnect authorization, state changed
from AUTHORIZING to DONE
02:14:23: XC AUTH [Et1/0.2, 1002]: Event: free xconnect authorization request, state
changed from DONE to END
02:14:23: MPLS peer 10.1.1.2 vcid 1234001, VC UP, VC state UP
```

The following example shows how to remove the xconnects associated with interface Ethernet 1/0.1:

```
Router# clear xconnect interface eth1/0.1

02:14:48: Xconnect[ac:Et1/0.1(Eth VLAN)]: provisioning fwder with fwd_type=1, sss_role=2
```

```
02:14:48: Xconnect[mpls:10.1.1.2:1234000]: provisioning fwder with fwd_type=2, sss_role=1
02:14:48: MPLS peer 10.1.1.2 vcid 1234000, VC DOWN, VC state DOWN
02:14:48: XC AUTH [10.1.1.2, 1234000]: Event: start xconnect authorization, state changed
from IDLE to AUTHORIZING
02:14:48: XC AUTH [10.1.1.2, 1234000]: Event: found xconnect authorization, state changed
from AUTHORIZING to DONE
02:14:48: XC AUTH [10.1.1.2, 1234000]: Event: free xconnect authorization request, state
changed from DONE to END
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
MPLS commands	*Cisco IOS Multiprotocol Label Switching Command Reference*
Any Transport over MPLS	"Overview" section of Cisco Any Transport over MPLS
Any Transport over MPLS for the Cisco 10000 series router	Cisco 10000 Series Router Broadband Aggregation, Leased-Line, and MPLS Configuration Guide
Layer 2 Tunnel Protocol Version 3 (L2TPv3)	Layer 2 Tunnel Protocol Version 3 (L2TPv3)
L2VPN interworking	L2VPN Interworking

Standards

Standard	Title
draft-martini-l2circuit-trans-mpls-08.txt	*Transport of Layer 2 Frames Over MPLS*
draft-martini-l2circuit-encap-mpls-04.txt	*Encapsulation Methods for Transport of Layer 2 Frames Over MPLS*

MIBs

MIB	MIBs Link
ATM AAL5 over MPLS and ATM Cell Relay over MPLS: • MPLS LDP MIB (MPLS-LDP-MIB.my) • ATM MIB (ATM-MIB.my) • CISCO AAL5 MIB (CISCO-AAL5-MIB.my) • Cisco Enterprise ATM Extension MIB (CISCO-ATM-EXT-MIB.my) • Supplemental ATM Management Objects (CISCO-IETF-ATM2-PVCTRAP-MIB.my) • Interfaces MIB (IF-MIB.my) Ethernet over MPLS: • CISCO-ETHERLIKE-CAPABILITIES.my • Ethernet MIB (ETHERLIKE-MIB.my) • Interfaces MIB (IF-MIB.my) • MPLS LDP MIB (MPLS-LDP-MIB.my) Frame Relay over MPLS: • Cisco Frame Relay MIB (CISCO-FRAME-RELAY-MIB.my) • Interfaces MIB (IF-MIB.my) • MPLS LDP MIB (MPLS-LDP-MIB.my) HDLC and PPP over MPLS: • MPLS LDP MIB (MPLS-LDP-MIB.my) • Interfaces MIB (IF-MIB.my)	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
RFC 3032	*MPLS Label Stack Encoding*
RFC 3036	*LDP Specification*

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Any Transport over MPLS

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 9 Feature Information for Any Transport over MPLS

Feature Name	Releases	Feature Information
Any Transport over MPLS	12.0(10)ST 12.0(21)ST 12.0(22)S 12.0(23)S 12.0(25)S 12.0(26)S 12.0(27)S 12.0(29)S 12.0(30)S 12.0(31)S 12.0(32)S 12.1(8a)E 12.2(14)S 12.2(15)T 12.2(28)SB 12.2(33)SRB 12.2(33)SXH 12.2(33)SRC 12.2(33)SRD 12.2(1)SRE 12.4(11)T 15.0(1)S 15.1(3)S	In Cisco IOS Release 12.0(10)ST, Any Transport over MPLS: ATM AAL5 over MPLS was introduced on the Cisco 12000 series routers. In Cisco IOS Release 12.1(8a)E, Ethernet over MPLS was introduced on the Cisco 7600 series Internet router. In Cisco IOS Release 12.0(21)ST, Any Transport over MPLS: Ethernet over MPLS was introduced on the Cisco 12000 series routers. ATM AAL5 over MPLS was updated. In Cisco IOS Release 12.0(22)S, Ethernet over MPLS was integrated into this release. Support for the Cisco 10720 Internet router was added. ATM AAL5 over MPLS was integrated into this release for the Cisco 12000 series routers. In Cisco IOS Release 12.0(23)S, the following new features were introduced and support was added for them on the Cisco 7200 and 7500 series routers: • ATM Cell Relay over MPLS (single cell relay, VC mode) • Frame Relay over MPLS • HDLC over MPLS • PPP over MPLS Cisco IOS Release 12.0(23)S also added support on the Cisco 12000, 7200, and 7500 series routers for the following features: • ATM AAL5 over MPLS • Ethernet over MPLS (VLAN mode) The AToM features were integrated into Cisco IOS Release 12.2(14)S.

Feature Name	Releases	Feature Information
		The AToM features were integrated into Cisco IOS Release 12.2(15)T. In Cisco IOS Release 12.0(25)S, the following new features were introduced: • New commands for configuring AToM • Ethernet over MPLS: port mode • ATM Cell Relay over MPLS: packed cell relay • ATM Cell Relay over MPLS: VP mode • ATM Cell Relay over MPLS: port mode • Distributed Cisco Express Forwarding mode for Frame Relay, PPP, and HDLC over MPLS • Fast reroute with AToM • Tunnel selection • Traffic policing • QoS support

Feature Name	Releases	Feature Information
		In Cisco IOS Release 12.0(26)S, the following new features were introduced: • Support for connecting disparate attachment circuits. See L2VPN Interworking for more information. • QoS functionality with AToM for the Cisco 7200 series routers. Support for FECN and BECN marking with Frame Relay over MPLS. (See BECN and FECN Marking for Frame Relay over MPLS for more information.) In Cisco IOS Release 12.0(27)S, the following new features were introduced: • ATM Cell Relay over MPLS: Packed Cell Relay for VC, PVP, and port mode for the Cisco 12000 series router. • Support for ATM over MPLS on the Cisco 12000 series 4-port OC-12X/ STM-4 ATM ISE line card. This feature was integrated into Cisco IOS Release 12.2(25)S for the Cisco 7200 and 7500 series routers. In Cisco IOS Release 12.0(29)S, the “Any Transport over MPLS Sequencing Support” feature was added for the Cisco 7200 and 7500 series routers. In Cisco IOS Release 12.0(30)S, the following new features were introduced: In Cisco IOS Release 12.0(31)S, the Cisco 12000 series router introduced the following enhancements: • AToM VC Independence-- With this enhancement, fast

Feature Name	Releases	Feature Information
		reroute is accomplished in less than 50 milliseconds, regardless of the number of VCs configured. • Support for ISE line cards on the 2.5G ISE SPA Interface Processor (SIP). In Cisco IOS Release 12.0(32)S, the Cisco 12000 series router added engine 5 line card support for the following transport types: • Ethernet over MPLS • Frame Relay over MPLS • HDLC over MPLS • PPP over MPLS

Feature Name	Releases	Feature Information
		This feature was integrated into Cisco IOS Release 12.2(28)SB on the Cisco 10000 series routers. Platform-specific configuration information is contained in the "Configuring Any Transport over MPLS" section of the Cisco 10000 Series Router Broadband Aggregation, Leased-Line, and MPLS Configuration Guide. Any Transport over MPLS was integrated into Cisco IOS Release 12.4(11)T with support for the following features: • Any Transport over MPLS: Ethernet over MPLS: Port Mode • Any Transport over MPLS: Ethernet over MPLS: VLAN Mode • Any Transport over MPLS: Ethernet over MPLS: VLAN ID Rewrite • Any Transport over MPLS: Frame Relay over MPLS • Any Transport over MPLS: AAL5 over MPLS • Any Transport over MPLS: ATM OAM Emulation This feature was integrated into Cisco IOS Release 12.2(33)SRB to support the following features on the Cisco 7600 router: • Any Transport over MPLS: Frame Relay over MPLS • Any Transport over MPLS: ATM Cell Relay over MPLS: Packed Cell Relay • Any Transport over MPLS: Ethernet over MPLS • AToM Static Pseudowire Provisioning Platform-specific configuration information is contained in the following documents:

Feature Name	Releases	Feature Information
		• The "Configuring PFC3BXL and PFC3B Mode Multiprotocol Label Switching" module of the Cisco 7600 Series Cisco IOS Software Configuration Guide, Release 12.2SR • The "Configuring Multiprotocol Label Switching on the Optical Services Modules" module of the OSM Configuration Note, Release 12.2SR • The "Configuring Multiprotocol Label Switching on FlexWAN and Enhanced FlexWAN Modules" module of the FlexWAN and Enhanced FlexWAN Modules Configuration Guide • The "Configuring Any Transport over MPLS on a SIP" section of the Cisco 7600 Series Router SIP, SSC, and SPA Software Configuration Guide • The "Configuring AToM VP Cell Mode Relay Support" section of the Cisco 7600 Series Router SIP, SSC, and SPA Software Configuration Guide • The *Cross-Platform Release Notes for Cisco IOS Release 12.2SR for the Cisco 7600 Series Routers*

Feature Name	Releases	Feature Information
		This feature was integrated into Cisco IOS Release 12.2(33)SXH and supports the following features: • Any Transport over MPLS: Ethernet over MPLS: Port Mode • Any Transport over MPLS: AAL5 over MPLS • Any Transport over MPLS: ATM OAM Emulation • Any Transport over MPLS: Single Cell Relay--VC Mode • Any Transport over MPLS: ATM Cell Relay over MPLS--VP Mode • Any Transport over MPLS: Packed Cell Relay--VC/VP Mode • Any Transport over MPLS: Ethernet over MPLS • ATM Port Mode Packed Cell Relay over AToM • AToM Tunnel Selection The following features were integrated into Cisco IOS Release 12.2(33)SRC: • AToM Tunnel Selection for the Cisco 7200 and Cisco 7300 routers • Per-Subinterface MTU for Ethernet over MPLS (EoMPLS) In Cisco IOS Release 12.2(33)SRD, support for ATM Cell Relay over MPLS in port mode on Cisco 7600 series routers was added. Per Subinterface MTU for Ethernet over MPLS (EoMPLS) was integrated into Cisco IOS Release 15.1(3)S.

Feature Name	Releases	Feature Information
MPLS L2VPN Clear Xconnect Command	12.2(1)SRE 15.0(1)S	These features are supported on Cisco 7600 routers in Cisco IOS Release 12.2(1)SRE and Cisco IOS Release 15.0(1)S. These features enable you to: • Reset a VC associated with an interface, a peer address, or on all the configured xconnect circuit attachments • Set the control word on dynamic pseudowires. • Enable ATM cell packing for static pseudowires. The following commands were introduced or modified by these features: **cell-packing**, **clear xconnect**, **control-word**, **encapsulation (Any Transport over MPLS)**, **oam-ac emulation-enable**.
MPLS MTU Command for GRE Tunnels	15.1(1)T 15.1(2)S	This feature allows you to reset the MPLS MTU size in GRE tunnels from default to the maximum. The **maximum** keyword was replaced with the **max** keyword. The following command was modified by this feature: **mpls mtu**.
ATM Port mode Packed Cell Relay over MPLS	15.2(1)S	This feature was integrated into Cisco IOS Release 12.2(1)S.
Any Transport over MPLS (AToM): ATM Cell Relay over MPLS: Packed Cell Relay	15.2(1)S	This feature was integrated into Cisco IOS Release 12.2(1)S.

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.

MPLS MTU Command Changes

This document explains the change in the behavior of the **mplsmtu** command for the following Cisco IOS releases:

- 12.2(27)SBC and later
- 12.2(33)SRA and later
- 12.2(33)SXH and later
- 12.4(11)T and later
- 15.0(1)M1
- 15.1(2)S

You cannot set the Multiprotocol Label Switching (MPLS) maximum transmission unit (MTU) to a value larger than the interface MTU value. This eliminates problems such as dropped packets, data corruption, and high CPU rates from occurring when the MPLS MTU value settings are larger than the interface MTU values. Cisco IOS software allows the MPLS MTU value to be higher than the interface MTU value only for interfaces that have a default interface MTU value of 1580 or less.

Note In Cisco IOS Release 15.0(1)M1, you can configure the interface MTU on PA-1FE and PA-2FE port adapters (PAs). The range of values is from 1500 to 1530. Before this enhancement, the MTU of those interfaces was not configurable, and any attempt to configure the interface MTU displayed the following message: *%Interface{InterfaceName}doesnotsupportusersettablemtu*.

- Finding Feature Information, page 109
- Information About MPLS MTU Command Changes, page 110
- How to Configure MPLS MTU Values, page 111
- Configuration Examples for Setting the MPLS MTU Values, page 115
- Additional References, page 117
- Feature Information for MPLS MTU Command Changes, page 118

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About MPLS MTU Command Changes

- MPLS MTU Values During Upgrade, page 110
- Guidelines for Setting MPLS MTU and Interface MTU Values, page 110
- MPLS MTU Values for Ethernet Interfaces, page 111

MPLS MTU Values During Upgrade

If you have configuration files with MPLS MTU values that are larger than the interface MTU values and you upgrade to Cisco IOS Release 12.2(27)SBC, 12.2(33)SRA, 12.4(11)T, 12.2(33)SXH, or later releases, the software does not change the MPLS MTU value. When you reboot the router, the software accepts the values that are set for the MPLS MTU and the interface MTU. The following error message is displayed during system initialization:

```
Setting the mpls mtu to xxxx on interface x/x, which is higher than the interface MTU
xxxx. This could lead to packet forwarding problems including packet drops.
You must set the MPLS MTU values equal to or lower than the interface MTU values.
```

Caution If you do not set the MPLS MTU less than or equal to the interface MTU, data corruption, dropped packets, and high CPU conditions can occur.

Guidelines for Setting MPLS MTU and Interface MTU Values

When configuring the network to use MPLS, set the core-facing interface MTU values greater than the edge-facing interface MTU values using one of the following methods:

- Set the interface MTU values on the core-facing interfaces to a higher value than the interface MTU values on the customer-facing interfaces to accommodate any packet labels, such as MPLS labels, that an interface might encounter. Make sure that the interface MTUs on the remote end interfaces have the same interface MTU values. The interface MTU values on both ends of the link must match.
- Set the interface MTU values on the customer-facing interfaces to a lower value than the interface MTU on the core-facing interfaces to accommodate any packet labels, such as MPLS labels, that an interface might encounter. When you set the interface MTU on the edge interfaces, ensure that the interface MTUs on the remote end interfaces have the same values. The interface MTU values on both ends of the link must match.

Changing the interface MTU can also modify the IP MTU, Connectionless Network Service (CLNS) MTU, and other MTU values because they depend on the value of the interface MTU. The Open Shortest Path First (OSPF) routing protocol requires that the IP MTU values match on both ends of the link. Similarly, the Intermediate System-to-Intermediate System (IS-IS) routing protocol requires that the CLNS MTU values match on both ends of the link. If the values on both ends of the link do not match, IS-IS or OSPF cannot complete initialization.

If the configuration of the adjacent router does not include the **mplsmtu**and **mtu** commands, add these commands to the router.

Note The MPLS MTU setting is displayed only in the show running-config output if the MPLS MTU value is different from the interface MTU value. If the values match, only the interface MTU value is displayed.

If you attempt to set the MPLS MTU value higher than the interface MTU value, the software displays the following error message, which prompts you to set the interface MTU to a higher value before you set the MPLS MTU value:

```
% Please increase interface mtu to xxxx and then set mpls mtu
```

Note In Cisco IOS Release 15.1(2)S, the **mplsmtu** command was modified. This command was made available in L3VPN encapsulation configuration mode. The **maximum** keyword was replaced with the **max** keyword. The **override** keyword and the *bytes* argument were removed from the GRE tunnel interface. To set MPLS MTU to the maximum MTU on L3VPN profiles, use the **mplsmtu** command in L3VPN encapsulation configuration mode.

MPLS MTU Values for Ethernet Interfaces

If you have an interface with a default interface MTU value of 1580 or less (such as an Ethernet interface), the **mplsmtu** command provides an **override**keyword, which allows you to set the MPLS MTU to a value higher than the interface MTU value. The **override** keyword is not available for interface types that do not have a default interface MTU value of 1580 or less. For configuration details, see the Setting the MPLS MTU Value on an Ethernet Interface, page 113.

Setting the MPLS MTU value higher than the Ethernet interface MTU value can lead to dropped packets, data corruption, or high CPU rates. When you set the MPLS MTU value higher than the Ethernet interface MTU value, the software displays the following message:

```
%MFI-30-MPLS_MTU_SET: Setting the mpls mtu to xxxx on Ethernet x/x, which is higher than
the interface MTU xxxx. This could lead to packet forwarding problems including packet
drops.
Most drivers will be able to support baby giants and will gracefully drop packets that
are too large. Certain drivers will have packet forwarding problems including data
corruption.
Setting the mpls mtu higher than the interface mtu can lead to packet forwarding problems
and may be blocked in a future release.
```

Note The**override** keyword is supported in Cisco IOS Release 12.2(27)SBC, 12.2(33)SRA, 12.4(11)T, 12.2(33)SXH, and later releases, but may not be supported in a future release.

How to Configure MPLS MTU Values

The following sections explain how to configure MPLS MTU and interface MTU values:

- Setting the Interface MTU and MPLS MTU Values, page 112
- Setting the MPLS MTU Value on an Ethernet Interface, page 113
- Setting the MPLS MTU Value to the Maximum on L3VPN Profiles, page 114

Setting the Interface MTU and MPLS MTU Values

Use the following steps to set the interface MTU and the MPLS MTU.

Note In Cisco IOS Release 15.0(1)M1, you can configure the interface MTU on PA-1FE and PA-2FE port adapters. The range of values is from 1500 to 1530. Before this enhancement, the MTU of those interfaces was not configurable.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type slot* / *port*
4. **mtu** *bytes*
5. **mpls mtu** *bytes*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type slot* / *port* **Example:** `Router(config)# interface Serial 1/0`	Enters interface configuration mode to configure the interface.
Step 4	**mtu** *bytes* **Example:** `Router(config-if)# mtu 1520`	Sets the interface MTU size.

	Command or Action	Purpose
Step 5	**mpls mtu** *bytes* **Example:** `Router(config-if)# mpls mtu 1520`	Sets the MPLS MTU to match the interface MTU.

Step 6	**end**
	Example:
	`Router(config-if)# end`
		Exits interface configuration mode and returns to privileged EXEC mode.

Setting the MPLS MTU Value on an Ethernet Interface

Use the following steps to set the MPLS MTU value on an Ethernet interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type slot* / *port*
4. **mpls mtu override** *bytes*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type slot* / *port* **Example:** `Router(config)# interface ethernet 1/0`	Enters interface configuration mode to configure the Ethernet interface.

	Command or Action	Purpose
Step 4	**mpls mtu override** *bytes* **Example:** `Router(config-if)# mpls mtu override 1510`	Sets the MPLS MTU to a value higher than the interface MTU value. **Caution** Setting the MPLS MTU to a value higher than the Ethernet interface MTU value can lead to dropped packets, data corruption, or high CPU rates.

Step 5	**end**
	Example:
	`Router(config-if)# end`
		Exits interface configuration mode and returns to privileged EXEC mode.

Setting the MPLS MTU Value to the Maximum on L3VPN Profiles

Use the following steps to set the MPLS MTU value to the maximum on L3VPN profiles.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **l3vpn encapsulation ip** *profile*
4. **mpls mtu max**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**l3vpn encapsulation ip** *profile* **Example:** `Router(config)# l3vpn encapsulation ip profile1`	Configures an L3VPN encapsulation profile and enters the L3VPN encapsulation configuration mode.

	Command or Action	Purpose
Step 4	**mpls mtu max** **Example:** `Router(config-l3vpn-encap-ip)# mpls mtu max`	Sets the MPLS MTU value to the maximum MTU on the L3VPN profile.

Step 5	**end**
	Example:
	`Router(config-l3vpn-encap-ip)# end`
		Exits L3VPN encapsulation configuration mode and returns to privileged EXEC mode.

Configuration Examples for Setting the MPLS MTU Values

- Example Setting the Interface MTU and MPLS MTU, page 115
- Example Setting the MPLS MTU Value on an Ethernet Interface, page 116
- Example Setting the MPLS MTU Value to the Maximum MTU on L3VPN profiles, page 117

Example Setting the Interface MTU and MPLS MTU

The following example shows how to set the interface and MPLS MTU values. The serial interface in the following configuration examples is at the default interface MTU value. The MPLS MTU value is not set. The interface settings are as follows:

```
interface Serial 4/0
 ip unnumbered Loopback0
 mpls traffic-eng tunnels
 mpls ip
 serial restart-delay 0
 ip rsvp bandwidth 2000 2000
end
```

The following example attempts to set the MPLS MTU value to 1520. This returns an error because MPLS MTU cannot be set to a value greater than the value of the interface MTU.

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# interface serial 4/0
Router(config-if)# mpls mtu 1520
% Please increase interface mtu to 1520 and then set mpls mtu
```

The following example first sets the interface MTU to 1520 and then sets the MPLS MTU to 1520:

```
Router(config-if)# mtu 1520
Router(config-if)# mpls mtu 1520
```

The following example shows the new interface MTU value. The MPLS MTU value is not displayed because it is equal to the interface value.

```
Router#
```

```
show running-config interface serial 4/0
Building configuration...
interface Serial4/0
 mtu 1520
 ip unnumbered Loopback0
 mpls traffic-eng tunnels
 mpls ip
 serial restart-delay 0
 ip rsvp bandwidth 2000 2000
end
```

The following example sets the MPLS MTU value to 1510:

```
Router(config-if)# mpls mtu 1510
```

The following example shows the new interface MTU value. The MPLS MTU value is displayed because it is different than the interface MTU value.

```
Router#
show running-config interface serial 4/0
Building configuration...
interface Serial4/0
 mtu 1520
 ip unnumbered Loopback0
 mpls mtu 1510
 mpls traffic-eng tunnels
 mpls ip
 serial restart-delay 0
 ip rsvp bandwidth 2000 2000
end
```

Example Setting the MPLS MTU Value on an Ethernet Interface

Caution Setting the MPLS MTU to a value higher than the Ethernet interface MTU value can lead to dropped packets, data corruption, or high CPU rates.

The following example shows how to set the MPLS MTU values on an Ethernet interface. The Ethernet interface in the following configuration examples is at the default interface MTU value. The MPLS MTU value is not set. The interface settings are as follows:

```
interface Ethernet 2/0
 ip unnumbered Loopback0
 mpls traffic-eng tunnels
 mpls ip
 serial restart-delay 0
 ip rsvp bandwidth 2000 2000
end
```

The following example uses the **override** keyword to set the MPLS MTU to 1520, which is higher than the Ethernet interface's MTU value:

```
Router(config-if)# mpls mtu override 1520
%MFI-30-MPLS_MTU_SET: Setting the mpls mtu to 1520 on Ethernet2/0, which is higher than
the interface MTU 1500. This could lead to packet forwarding problems including packet
drops.
```

The following example shows the new MPLS MTU value:

```
Router#
show running-config interface ethernet 2/0
Building configuration...
interface Ethernet 2/0
 mtu 1500
 ip unnumbered Loopback0
 mpls mtu 1520
 mpls traffic-eng tunnels
 mpls ip
 serial restart-delay 0
 ip rsvp bandwidth 2000 2000
end
```

Example Setting the MPLS MTU Value to the Maximum MTU on L3VPN profiles

The following example shows how to set the MPLS MTU value to the maximum MTU on L3VPN profiles:

```
Router# configure terminal
Router(config)# l3vpn encapsulation ip profile1
Router(config-l3vpn-encap-ip)# mpls mtu max
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
MPLS commands	

MIBs

MIBs	MIBs Link
None.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for MPLS MTU Command Changes

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 10 Feature Information for MPLS MTU Command Changes

Feature Name	Releases	Feature Information
MPLS MTU Command Changes	12.2(27)SBC 12.2(28)SB 12.2(33)SRA 12.2(33)SXH 12.4(11)T 15.0(1)M1 15.1(2)S	This document explains the changes to the **mplsmtu** command. You cannot set the MPLS MTU value larger than the interface MTU value, except for Ethernet interfaces. In 12.2(28)SB, support was added for the Cisco 10000 router. In 12.2(33)SRA, support was added for the Cisco 7600 series router. This feature was integrated into Cisco IOS Release 12.4(11)T. This feature was integrated into Cisco IOS Release 12.2(33)SXH. In 15.0(1)M1, you can configure the interface MTU on PA-1FE and PA-2FE port adapters. In 15.1(2)S, the **mplsmtu** command was made available in L3VPN encapsulation configuration mode. The **maximum** keyword was replaced with the **max** keyword. The **override** keyword and the *bytes* argument were removed from the GRE tunnel interface.

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.

AToM Static Pseudowire Provisioning

The AToM Static Pseudowire Provisioning feature allows provisioning an Any Transport over Multiprotocol Label Switching (MPLS) (AToM) static pseudowire without the use of a directed control connection. In environments that do not or cannot use directed control protocols, this feature provides a means for provisioning the pseudowire parameters statically at the Cisco IOS command-line interface (CLI).

- Finding Feature Information, page 121
- Restrictions for AToM Static Pseudowire Provisioning, page 121
- Information About AToM Static Pseudowire Provisioning, page 122
- How to Provision an AToM Static Pseudowire, page 122
- Configuration Examples for AToM Static Pseudowire Provisioning, page 125
- Additional References, page 126
- Feature Information for AToM Static Pseudowire Provisioning, page 127

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for AToM Static Pseudowire Provisioning

The following parameters are exchanged using directed control protocol messages on pseudowires, but cannot be changed using the AToM Static Pseudowire Provisioning feature introduced in Cisco IOS Release 12.33(SRB). Instead, the software has preconfigured defaults.

- The Virtual Circuit Connectivity Verification (VCCV) options used for fault detection, isolation, and verification at both ends of the connection are set as follows:
    - Control channel type 1 sets the control word.
    - Control channel type 2 sets the MPLS router alert label.
    - Connectivity verification type 2 sets the label switched path (LSP) **ping** command.

In Cisco IOS Release 12.2(33)SRE, support for cell packing for static pseudowires was added. This feature has the following restrictions:

- Both provider-edge routers (PEs) must run Cisco IOS Release 12.2(33)SRE, and the maximum number of cells that can be packed must be set to the same value on each PE router.
- Autosensing of the virtual circuit type for Ethernet over MPLS is not supported.

Additionally, the following functionality is not supported for static pseudowires:

- Sequence number resynchronization—configured by the sequencing function in the **pseudowire-class** command—is not supported because the sequence number resynchronization is done when the Label Distribution Protocol (LDP) software sends an LDP Label Release or Withdraw message followed by a Label Request or Mapping message, and static pseudowires do not use LDP.
- Tunnel stitching is not supported because it requires an extension of the **neighbor** command to start the mode that allows configuring static pseudowire parameters such as remote and local labels. Note that a tunnel switch point can be configured using a different static label command. The tunnel switch point will not process control words, but label swapping will occur.
- Pseudowire redundancy is not supported because it requires using a directed control protocol between the peer provider edge routers.

Information About AToM Static Pseudowire Provisioning

- Pseudowire Provisioning, page 122
- Benefits of Statically Provisioned Pseudowires, page 122

Pseudowire Provisioning

The AToM Static Pseudowire Provisioning feature allows you to configure static pseudowires in cases where you cannot use directed control protocols. .In most cases, pseudowires are dynamically provisioned using LDP or another directed control protocol, such as Resource Reservation Protocol over traffic-engineered tunnels (RSVP-TE), to exchange the various parameters required for these connections.

The AToM Static Pseudowire Provisioning feature is platform-independent, but has been tested on only the Cisco 7600 series routers.

Benefits of Statically Provisioned Pseudowires

This feature allows provisioning an AToM label switching static pseudowire without the use of a directed control connection. This feature also includes static provisioning of the tunnel label and the pseudowire label.

How to Provision an AToM Static Pseudowire

- Provisioning an AToM Static Pseudowire, page 122
- Verifying the AToM Static Pseudowire Configuration, page 124

Provisioning an AToM Static Pseudowire

In this configuration task, you use options in the **xconnect** Ethernet interface configuration command to specify a static connection, and **mpls** commands in xconnect mode to statically set the following pseudowire parameters:

- Set the local and remote pseudowire labels
- Enable or disable sending the MPLS control word

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *ethernet-type interface-number*
4. **xconnect** *peer-ip-address vcid* **encapsulation mpls manual pw-class** *class-name*
5. **mpls label** *local-pseudowire-label remote-pseudowire-label*
6. [**no**] **mpls control-word**
7. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *ethernet-type interface-number* **Example:** `Router(config)# interface Ethernet 1/0`	Enters interface configuration mode for the specified interface.
Step 4	**xconnect** *peer-ip-address vcid* **encapsulation mpls manual pw-class** *class-name* **Example:** `Router(config-if)# xconnect 10.131.191.252 100 encapsulation mpls manual pw-class mpls`	Configures a static AToM pseudowire and enters xconnect configuration mode where the local and remote pseudowire labels are set.

	Command or Action	Purpose
Step 5	**mpls label** *local-pseudowire-label remote-pseudowire-label* **Example:** `Router(config-if-xconn)# mpls label 100 150`	Sets the local and remote pseudowire labels. • The label must be an unused static label within the static label range configured using the **mpls label range** command. • The **mpls label** command checks the validity of the label entered and displays an error message if it is not valid. The label supplied for the *remote-pseudowire-label*argument must be the value of the peer PE's local pseudowire label.
Step 6	[**no**] **mpls control-word** **Example:** `Router(config-if-xconn)# no mpls control-word`	Sets whether the MPLS control word is sent. • This command must be set for Frame Relay data-link connection identifier (DLCI) and ATM adaptation layer 5 (AAL5) attachment circuits. For other attachment circuits, the control word is included by default. • If you enable inclusion of the control word, it must be enabled on both ends of the connection for the circuit to work properly. • Inclusion of the control word can be explicitly disabled using the **no mpls control-word** command.
Step 7	**exit** **Example:** `Router(config-if-xconn)# exit`	Exits the configuration mode. • Continue entering the **exit**command at the router prompt until you reach the desired configuration mode.

Verifying the AToM Static Pseudowire Configuration

To verify the AToM static pseudowire configuration, use the **show running-config** EXEC command. To verify that the AToM static pseudowire was provisioned correctly, use the **show mpls l2transport vc detail**and **ping mpls pseudowire**EXEC commands as described in the following steps.

SUMMARY STEPS

1. **show mpls l2transport vc detail**
2. **ping mpls pseudowire** *ipv4-address* **vc-id** *vc-id*

DETAILED STEPS

Step 1 **show mpls l2transport vc detail**

For nonstatic pseudowire configurations, this command lists the type of protocol used to send the MPLS labels (such as LDP). For static pseudowire configuration, the value of the signaling protocol field should be Manual. Following is sample output:

Example:

```
Router# show mpls l2transport vc detail
```

```
Local interface: Et1/0 up, line protocol up, Ethernet up
  Destination address: 10.0.1.1, VC ID: 200, VC status: up
    Output interface: Et3/0, imposed label stack {17}
    Preferred path: not configured
    Default path:
    Next hop: 10.0.0.2
  Create time: 00:27:27, last status change time: 00:27:24
  Signaling protocol: Manual
    MPLS VC labels: local 17, remote 17
    Group ID: local 0, remote 0
    MTU: local 1500, remote 1500
  Sequencing: receive disabled, send disabled
  VC statistics:
    packet totals: receive 193, send 193
    byte totals:   receive 19728, send 23554
    packet drops:  receive 0, send 0
```

Step 2 **ping mpls pseudowire** *ipv4-address* **vc-id** *vc-id*

Because there is no directed control protocol exchange of parameters on a static pseudowire, both ends of the connection must be correctly configured. One way to detect mismatch of labels or control word options is to send an MPLS pseudowire LSP **ping** command as part of configuration task, and then reconfigure the connection if problems are detected. An exclamation point (!) is displayed when the **ping** command is successfully sent to its destination. An example of command use and output follows:

Example:

```
Router# ping mpls pseudowire 10.7.1.2 vc-id 1001
Sending 5, 100-byte MPLS Echos to 10.7.1.2,
      timeout is 2 seconds, send interval is 0 msec:
Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
   'L' - labeled output interface, 'B' - unlabeled output interface,
   'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
   'M' - malformed request, 'm' - unsupported tlvs, 'N' - no label entry,
   'P' - no rx intf label prot, 'p' - premature termination of LSP,
   'R' - transit router, 'I' - unknown upstream index,
   'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
```

Configuration Examples for AToM Static Pseudowire Provisioning

- Provisioning an AToM Pseudowire Example, page 125

Provisioning an AToM Pseudowire Example

The following examples show the configuration commands for an AToM static pseudowire connection between two PEs, PE1 and PE2.

The **mpls label range static** command must be used to configure the static label range prior to provisioning the AToM static pseudowire.

```
Router# configure terminal
```

```
Router(config)# mpls label range 200 16000 static 16 199
% Label range changes will take effect at the next reload.
```

The **mpls ip** command must also be configured on the core-facing interface of both PE1 and PE2 (which is also done for directed control protocol signaled pseudowires). Following is a configuration example:

```
Router(config)# interface Ethernet 0/0
Router(config-if)# description Backbone interface
Router(config-if)# ip address 10.0.0.1 255.255.255.0
Router(config-if)# mpls ip
Router(config-if)# exit
```

Following is an example AToM static pseudowire configuration for PE1:

```
Router(config)# interface Ethernet 1/0
Router(config-if)# no ip address
Router(config-if)# xconnect 10.131.191.251 100 encapsulation mpls manual pw-class mpls
Router(config-if-xconn)# mpls label 100 150
Router(config-if-xconn)# exit
```

Following is an example AToM static pseudowire configuration for PE2:

```
Router(config)# interface Ethernet 1/0
Router(config-if)# no ip address
Router(config-if)# xconnect 10.132.192.252 100 encapsulation mpls manual pw-class mpls
Router(config-if-xconn)# mpls label 150 100
Router(config-if-xconn)# exit
```

This feature also allows tunnel labels to be statically configured using the **mpls static binding ipv4 vrf** command. See the MPLS Static Labels feature module and the Cisco IOS Multiprotocol Label Switching Command Reference for information about static labels and the **mpls static binding ipv4 vrf**command.

Additional References

The following sections provide references related to the AToM Static Pseudowire Provisioning feature.

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
MPLS commands	*Cisco IOS Multiprotocol Label Switching Command Reference*
Configuring the pseudowire class	Any Transport over MPLS
MPLS and xconnect commands	*Cisco IOS Multiprotocol Label Switching Command Reference*
Static labels and the **mpls static binding ipv4 vrf**command	" MPLS Static Labels ” section of the *Cisco IOS Multiprotocol Label Switching Configuration Guide*

Standards

Standard	Title
IETF draft-ietf-pwe3-vccv-12.txt	Pseudo Wire Virtual Circuit Connectivity Verification (VCCV)

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
RFC 3036	LDP Specification

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for AToM Static Pseudowire Provisioning

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 11 Feature Information for AToM Static Pseudowire Provisioning

Feature Name	Releases	Feature Information
AToM Static Pseudowire Provisioning	12.2(33)SRB 12.2(33)SRE	This feature allows provisioning an AToM static pseudowire without the use of a directed control protocol connection. The AToM Static Pseudowire feature is platform-independent, but has been tested on only the Cisco 7600 series routers for Cisco IOS Release 12.33(SRB). In Cisco IOS Release 12.2(33)SRE, the L2VPN Support for Cell Packing on Static PW feature was added. The following commands were introduced or modified by this feature: **cell-packing**, **mpls control-word**, **mpls label**, **show mpls l2transport vc**, **xconnect**.

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.

L2VPN Interworking

Layer 2 Virtual Private Network (L2VPN) Interworking allows you to connect disparate attachment circuits. This feature module explains how to configure the following L2VPN Interworking features:

- Ethernet/VLAN to ATM AAL5 Interworking
- Ethernet/VLAN to Frame Relay Interworking
- Ethernet/VLAN to PPP Interworking
- Ethernet to VLAN Interworking
- Frame Relay to ATM AAL5 Interworking
- Frame Relay to PPP Interworking
- Ethernet/VLAN to ATM virtual channel identifier (VPI) and virtual channel identifier (VCI) Interworking
- L2VPN Interworking: VLAN Enable/Disable Option for AToM

- Finding Feature Information, page 129
- Prerequisites for L2VPN Interworking, page 129
- Restrictions for L2VPN Interworking, page 130
- Information About L2VPN Interworking, page 139
- How to Configure L2VPN Interworking, page 142
- Configuration Examples for L2VPN Interworking, page 149
- Additional References, page 163
- Feature Information for L2VPN Interworking, page 164

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for L2VPN Interworking

Before you configure L2VPN Interworking on a router:

- You must enable Cisco Express Forwarding.

- On the Cisco 12000 series Internet router, before you configure Layer 2 Tunnel Protocol version 3 (L2TPv3) for L2VPN Interworking on an IP Services Engine (ISE/Engine 3) or Engine 5 interface, you must also enable the L2VPN feature bundle on the line card.

To enable the feature bundle, enter the **hw-module slot np mode feature** command in global configuration mode as follows:

```
Router# configure terminal
Router(config)# hw-module slot slot-number np mode feature
```

Restrictions for L2VPN Interworking

- General Restrictions, page 130
- Cisco 7600 Series Routers Restrictions, page 130
- Cisco 12000 Series Router Restrictions, page 132
- ATM AAL5 Interworking Restrictions, page 135
- Ethernet VLAN Interworking Restrictions, page 135
- Restrictions, page 136
- Frame Relay Interworking Restrictions, page 138
- PPP Interworking Restrictions, page 138

General Restrictions

This section lists general restrictions that apply to L2VPN Interworking. Other restrictions that are platform-specific or device-specific are listed in the following sections.

- The interworking type on one provider edge (PE) router must match the interworking type on the peer PE router.
- The following quality of service (QoS) features are supported with L2VPN Interworking:
  - Static IP type of service (ToS) or Multiprotocol Label Switching (MPLS) experimental bit (EXP) setting in tunnel header
  - IP ToS reflection in tunnel header (Layer 2 Tunnel Protocol Version 3 (L2TPv3) only)
  - Frame Relay policing
  - Frame Relay data-link connection identifier (DLCI)-based congestion management (Cisco 7500/ Versatile Interface Processor (VIP))
  - One-to-one mapping of VLAN priority bits to MPLS EXP bits
- Only ATM AAL5 VC mode is supported; ATM VP and port mode are not supported.
- In Cisco IOS Release 12.2(52)SE and Cisco IOS Release 12.2(33)SRE, the **encapsulation** command supports only the **mpls** keyword. The **l2tpv3** keyword is not supported. The **interworking** command supports only the **ethernet** and **vlan** keywords. The **ip** keyword is not supported.

Cisco 7600 Series Routers Restrictions

The following line cards are supported on the Cisco 7600 series router. The first table below shows the line cards that are supported on the WAN (ATM, Frame Relay, or PPP) side of the interworking link. The second table below shows the line cards that are supported on the Ethernet side of the interworking link. For more details on the Cisco 7600 routers supported shared port adapters and line cards, see the following document:

- Release Notes for Cisco IOS Release 12.2SR for the Cisco 7600 Series Routers

Table 12 Cisco 7600 Series Routers: Supported Line Cards for the WAN Side

Interworking Type	Core-Facing Line Cards	Customer-Edge Line Cards
Ethernet (bridged) (ATM and Frame Relay)	Any	EflexWAN SIP-200 SIP-400
IP (routed) (ATM, Frame Relay, and PPP)	Any	EflexWAN SIP-200

Table 13 Cisco 7600 Series Routers: Supported Line Cards for the Ethernet Side

Interworking Type	Ethernet over MPLS Mode	Core-Facing Line Cards	Customer-Edge Line Cards
Ethernet (bridged)	Policy feature card (PFC) based	Any, except optical service module (OSM) and ES40	Catalyst LAN SIP-600
Ethernet (bridged)	Switched virtual interface (SVI) based	EflexWAN ES20 ES+40 SIP-200 SIP-400 SIP-600	Catalyst LAN EflexWAN (with MPB) ES20 ES+40 SIP-200 (with MPB) SIP-400 (with MPB) SIP-600
Ethernet (bridged)	Scalable (with E-MPB)	Any, except OSM	ES20 SIP-600 and SIP-400 with Gigabit Ethernet (GE) SPA
IP (routed)	PFC-based	Catalyst LAN SIP-600 **Note:** PFC-based mode is not supported with routed interworking in Cisco IOS Release 12.2(33)SRD. Use SVI, Scalable, or Ethernet virtual connection (EVC) based Ethernet over MPLS (EoMPLS) instead.	Catalyst LAN SIP-600 **Note:** PFC-based mode is not supported with routed interworking in Cisco IOS Release 12.2(33)SRD. Use SVI, Scalable, or EVC-based EoMPLS instead.
IP (routed)	SVI-based	Any, except Catalyst LAN and OSM.	Catalyst LAN EflexWAN (with MPB) ES20 SIP-200 (with MPB) SIP-400 (with MPB) SIP-600

The following restrictions apply to the Cisco 7600 series routers and L2VPN Interworking:

- OAM Emulation is not required with L2VPN Interworking on the SIP-200, SIP-400, and Flexwan2 line cards.
- Cisco 7600 series routers support the L2VPN Interworking: VLAN Enable/Disable Option for AToM feature starting in Cisco IOS Release 12.2(33)SRE. This feature has the following restrictions:
    - PFC-based EoMPLS is not supported.
    - Scalable and SVI-based EoMPLS are supported with the SIP-400 line card.
- The Cisco 7600 series routers do not support L2VPN Interworking over L2TPv3.
- Cisco 7600 series routers support only the following interworking types:
    - Ethernet/VLAN to Frame Relay (IP and Ethernet modes)
    - Ethernet/VLAN to ATM AAL5SNAP (IP and Ethernet modes)
    - Ethernet/VLAN to PPP (IP only)
    - Ethernet to VLAN Interworking
- Cisco 7600 series routers do not support the following interworking types:
    - Ethernet/VLAN to ATM AAL5MUX
    - Frame Relay to PPP Interworking
    - Frame Relay to ATM AAL5 Interworking
- Both ends of the interworking link must be configured with the same encapsulation and interworking type:
    - If you use Ethernet encapsulation, you must use the Ethernet (bridged) interworking type. If you are not using Ethernet encapsulation, you can use a bridging mechanism, such as routed bridge encapsulation (RBE).
    - If you use an IP encapsulation (such as ATM or Frame Relay), you must use the IP (routed) interworking type. The PE routers negotiate the process for learning and resolving addresses.
    - You must use the same MTU size on the attachment circuits at each end of the pseudowire.
- PFC-based EoMPLS is not supported on ES40 line cards. SVI and EVC/scalable EoMPLS are the alternative options.
- PFC-based EoMPLS is not supported for Routed/IP interworking in Cisco IOS Release 12.2(33)SRD and later releases. The alternative Routed/IP interworking options are SVI and EVC or scalable EoMPLS. However, PFC-based EoMPLS is supported for Ethernet/Bridged interworking and for like-to-like over AToM.

Cisco 12000 Series Router Restrictions

For more information about hardware requirements on the Cisco12000 series routers, see the Cross-Platform Release Notes for Cisco IOS Release 12.0S.

For QOS support on the Cisco 12000 series routers, see Any Transport over MPLS (AToM): Layer 2 QoS (Quality of Service) for the Cisco 12000 Series Router

Frame Relay to PPP and High-Level Data Link Control Interworking

The Cisco 12000 series Internet router does not support L2VPN Interworking with PPP and high-level data link control (HDLC) transport types in Cisco IOS releases earlier than Cisco IOS Release 12.0(32)S.

In Cisco IOS Release 12.0(32)S and later releases, the Cisco 12000 series Internet router supports L2VPN interworking for Frame Relay over MPLS and PPP and HDLC over MPLS only on the following shared port adapters (SPAs):

- ISE/Engine 3 SPAs:

    - SPA-2XCT3/DS0 (2-port channelized T3 to DS0)
    - SPA-4XCT3/DS0 (4-port channelized T3 to DS0)
- Engine 5 SPAs:
    - SPA-1XCHSTM1/OC-3 (1-port channelized STM-1c/OC-3c to DS0)
    - SPA-8XCHT1/E1 (8-port channelized T1/E1)
    - SPA-2XOC-48-POS/RPR (2-port OC-48/STM16 POS/RPR)
    - SPA-OC-192POS-LR (1-port OC-192/STM64 POS/RPR)
    - SPA-OC-192POS-XFP (1-port OC-192/STM64 POS/RPR)

L2VPN Interworking over L2TPv3

On the Cisco 12000 series Internet router, Ethernet (bridged) interworking is not supported for L2TPv3. Only IP (routed) interworking is supported.

IP (routed) interworking is not supported in an L2TPv3 pseudowire that is configured for data sequencing (using the **sequencing** command).

In Cisco IOS Release 12.0(32)SY and later releases, the Cisco 12000 series Internet router supports L2VPN Interworking over L2TPv3 tunnels in IP mode on ISE and Engine 5 line cards as follows:

- On an ISE interface configured for L2TPv3 tunneling, the following Layer 2 encapsulations are supported:
    - ATM adaptation layer type-5 (AAL5)
    - Ethernet
    - 802.1q (VLAN)
    - Frame Relay DLCI
- On an Engine 5 interface configured for L2TPv3 tunneling, the following Layer 2 encapsulations are supported:
    - Ethernet
    - 802.1q (VLAN)
    - Frame Relay DLCI

For more information, refer to Layer 2 Tunnel Protocol Version 3.

The only frame format supported for L2TPv3 interworking on Engine 5 Ethernet SPAs is Ethernet Version 2 (also known as Ethernet II) with the Ether type 0x0800 value set as Internet Protocol Payload and (optionally) 802.1q VLAN. Ethernet packets with other Ethernet frame formats are dropped.

Remote Ethernet Port Shutdown Support

The Cisco Remote Ethernet Port Shutdown feature (which minimizes potential data loss after a remote link failure) is supported only on the following Engine 5 Ethernet SPAs:

- SPA-8XFE (8-port Fast Ethernet)
- SPA-2X1GE (2-port Gigabit Ethernet)
- SPA-5X1GE (5-port Gigabit Ethernet)
- SPA-10X1GE (10-port Gigabit Ethernet)
- SPA-1X10GE (1-port 10-Gigabit Ethernet)

For more information about this feature, refer to Any Transport over MPLS (AToM): Remote Ethernet Port Shutdown.

L2VPN Any-to-Any Interworking on Engine 5 Line Cards

The table below shows the different combinations of transport types supported for L2VPN interworking on Engine 3 and Engine 5 SPA interfaces connected through an attachment circuit over MPLS or L2TPv3.

Table 14 Engine 3 and Engine 5 Line Cards/SPAs Supported for L2VPN Interworking

Attachment Circuit 1 (AC1)	Attachment Circuit 2 (AC2)	Interworking Mode	AC1 Engine Type and Line Card/SPA	AC2 Engine Type and Line Card/SPA
Frame Relay	Frame Relay	IP	Engine 5 POS and channelized	Engine 3 ATM line cards
Frame Relay	ATM	Ethernet	Engine 5 POS and channelized	Engine 3 ATM line cards
Frame Relay	ATM	IP	Engine 5 POS and channelized	Engine 3 ATM line cards
Frame Relay	Ethernet	Ethernet	Engine 5 POS and channelized	Engine 5 Gigabit Ethernet
Frame Relay	Ethernet	IP	Engine 5 POS and channelized	Engine 5 Gigabit Ethernet
Frame Relay	VLAN	Ethernet	Engine 5 POS and channelized	Engine 5 Gigabit Ethernet
Frame Relay	VLAN	IP	Engine 5 POS and channelized	Engine 5 Gigabit Ethernet
Ethernet	Ethernet	Ethernet	Engine 5 Gigabit Ethernet	Engine 5 Gigabit Ethernet
Ethernet	Ethernet	IP	Engine 5 Gigabit Ethernet	Engine 5 Gigabit Ethernet
Ethernet	VLAN	Ethernet	Engine 5 Gigabit Ethernet	Engine 5 Gigabit Ethernet
Ethernet	VLAN	IP	Engine 5 Gigabit Ethernet	Engine 5 Gigabit Ethernet
ATM	Ethernet	Ethernet	Engine 3 ATM line cards	Engine 5 Gigabit Ethernet
ATM	Ethernet	IP	Engine 3 ATM line cards	Engine 5 Gigabit Ethernet

On the Cisco 12000 series Engine 3 line card, Network Layer Protocol ID (NLPID) encapsulation is not supported in routed mode; and neither NLPID nor AAL5MUX is supported in bridged mode.

- On the Cisco 12000 series Internet router, Ethernet (bridged) interworking is not supported for L2TPv3.

In an L2VPN Interworking configuration, after you configure L2TPv3 tunnel encapsulation for a pseudowire using the **encapsulation l2tpv3**command, you cannot enter the **interworking ethernet** command.

- On Ethernet SPAs on the Cisco 12000 series Internet router, the only frame format supported for L2TPv3 interworking is Ethernet Version 2 (also known as Ethernet II) with the Ether type 0x0800 value set as Internet Protocol Payload and [optionally] 802.1q VLAN.

Ethernet packets with other Ethernet frame formats are dropped.

ATM AAL5 Interworking Restrictions

The following restrictions apply to ATM AAL5 Interworking:

- Switched virtual circuits (SVCs) are not supported.
- Inverse Address Resolution Protocol (ARP) is not supported with IP interworking.
- Customer edge (CE) routers must use point-to-point subinterfaces or static maps.
- Both AAL5MUX and AAL5SNAP encapsulation are supported. In the case of AAL5MUX, no translation is needed.
- In the Ethernet end-to-end over ATM scenario, the following translations are supported:
    - Ethernet without LAN frame check sequence (FCS) (AAAA030080C200070000)
    - Spanning tree (AAAA030080c2000E)

Everything else is dropped.

- In the IP over ATM scenario, the IPv4 (AAAA030000000800) translation is supported. Everything else is dropped.
- Operation, Administration, and Management (OAM) emulation for L2VPN Interworking is the same as like-to-like. The end-to-end F5 loopback cells are looped back on the PE router. When the pseudowire is down, an F5 end-to-end segment Alarm Indication Signal (AIS)/Remote Defect Identification (RDI) is sent from the PE router to the CE router.
- Interim Local Management Interface (ILMI) can manage virtual circuits (VCs) and permanent virtual circuits (PVCs).
- To enable ILMI management, configure ILMI PVC 0/16 on the PE router’s ATM interface. If a PVC is provisioned or deleted, an ilmiVCCChange trap is sent to the CE router.
- Only the user side of the User-Network Interface (UNI) is supported; the network side of the UNI is not supported.

Ethernet VLAN Interworking Restrictions

The following restrictions apply to Ethernet/VLAN interworking:

- When you configure VLAN to Ethernet interworking, VLAN to Frame Relay (routed), or ATM using Ethernet (bridged) interworking, the PE router on the Ethernet side that receives a VLAN tagged frame from the CE router removes the VLAN tag. In the reverse direction, the PE router adds the VLAN tag to the frame before sending the frame to the CE router.

(If you enable the L2VPN Interworking: VLAN Enable/Disable Option for AToM feature with the **interworking vlan** command, VLAN ID is included as part of the Ethernet frame. See the VLAN Interworking, page 141 for more information.)

- In bridged interworking from VLAN to Frame Relay, the Frame Relay PE router does not strip off VLAN tags from the Ethernet traffic it receives.

- The Cisco 10720 Internet router supports Ethernet to VLAN Interworking Ethernet only over L2TPv3.
- Ethernet interworking for a raw Ethernet port or a VLAN trunk is not supported. Traffic streams are not kept separate when traffic is sent between transport types.
- In routed mode, only one CE router can be attached to an Ethernet PE router.
- There must be a one-to-one relationship between an attachment circuit and the pseudowire. Point-to-multipoint or multipoint-to-point configurations are not supported.
- Configure routing protocols for point-to-point operation on the CE routers when configuring an Ethernet to non-Ethernet setup.
- In the IP interworking mode, the IPv4 (0800) translation is supported. The PE router captures ARP (0806) packets and responds with its own MAC address (proxy ARP). Everything else is dropped.
- The Ethernet or VLAN must contain only two IP devices: PE router and CE router. The PE router performs proxy ARP and responds to all ARP requests it receives. Therefore, only one CE and one PE router should be on the Ethernet or VLAN segment.
- If the CE routers are doing static routing, you can perform the following tasks:
    - The PE router needs to learn the MAC address of the CE router to correctly forward traffic to it. The Ethernet PE router sends an Internet Control Message Protocol (ICMP) Router discovery protocol (RDP) solicitation message with the source IP address as zero. The Ethernet CE router responds to this solicitation message. To configure the Cisco CE router's Ethernet or VLAN interface to respond to the ICMP RDP solicitation message, issue the **ip irdp**command in interface configuration mode. If you do not configure the CE router, traffic is dropped until the CE router sends traffic toward the PE router.
    - To disable the CE routers from running the router discovery protocol, issue the **ip irdp maxadvertinterval 0** command in interface mode.
- This restriction applies if you configure interworking between Ethernet and VLAN with Catalyst switches as the CE routers. The spanning tree protocol is supported for Ethernet interworking. Ethernet interworking between an Ethernet port and a VLAN supports spanning tree protocol only on VLAN 1. Configure VLAN 1 as a nonnative VLAN.
- When you change the interworking configuration on an Ethernet PE router, clear the ARP entry on the adjacent CE router so that it can learn the new MAC address. Otherwise, you might experience traffic drops.

Restrictions

The following restrictions apply to the L2VPN Interworking: VLAN Enable/Disable Option for AToM feature, which allows the VLAN ID to be included as part of the Ethernet frame:

- The L2VPN Interworking: VLAN Enable/Disable Option for AToM feature is supported on the following releases:
    - Cisco IOS release 12.2(52)SE for the Cisco Catalyst 3750 Metro switches
    - Cisco IOS Release 12.2(33)SRE for the Cisco 7600 series routers
- L2VPN Interworking: VLAN Enable/Disable Option for AToM is not supported with L2TPv3. You can configure the featue only with AToM.
- If the interface on the PE router is a VLAN interface, it is not necessary to specify the **interworking vlan** command on that PE router.
- The L2VPN Interworking: VLAN Enable/Disable Option for AToM feature works only with the following attachment circuit combinations:
    - Ethernet to Ethernet
    - Ethernet to VLAN

  - VLAN to VLAN

- If you specify an interworking type on a PE router, that interworking type must be enforced. The interworking type must match on both PE routers. Otherwise, the VC may be in an incompatible state and remain in the down state. If the attachment circuit (AC) is VLAN, the PE router can negotiate (autosense) the VC type using Label Distribution Protocol (LDP).

For example, both PE1 and PE2 use Ethernet interfaces, and VLAN interworking is specified on PE1 only. PE2 is not configured with an interworking type and cannot autosense the interworking type. The result is an incompatible state where the VC remains in the down state.

On the other hand, if PE1 uses an Ethernet interface and VLAN interworking is enabled (which will enforce VLAN as the VC type), and PE2 uses a VLAN interface and interworking is not enabled (which causes PE2 to use Ethernet as its default VC type), PE2 can autosense and negotiate the interworking type and select VLAN as the VC type.

The table below summarizes shows the AC types, interworking options, and VC types after negotiation.

Table 15 Negotiating Ethernet and VLAN Interworking Types

PE1 AC Type	Interworking Option	PE2 AC Type	Interworking Option	VC Type after Negotiation
Ethernet	none	Ethernet	none	Ethernet
Vlan	none	Ethernet	none	Ethernet
Ethernet	none	Vlan	none	Ethernet
Vlan	none	Vlan	none	Ethernet
Ethernet	Vlan	Ethernet	none	Incompatible
Vlan	Vlan	Ethernet	none	Incompatible
Ethernet	Vlan	Vlan	none	Vlan
Vlan	Vlan	Vlan	none	Vlan
Ethernet	none	Ethernet	Vlan	Incompatible
Vlan	none	Ethernet	Vlan	Vlan
Ethernet	none	Vlan	Vlan	Incompatible
Vlan	none	Vlan	Vlan	Vlan
Ethernet	Vlan	Ethernet	Vlan	Vlan
Vlan	Vlan	Ethernet	Vlan	Vlan
Ethernet	Vlan	Vlan	Vlan	Vlan
Vlan	Vlan	Vlan	Vlan	Vlan

Frame Relay Interworking Restrictions

The following restrictions apply to Frame Relay interworking:

- The attachment circuit maximum transmission unit (MTU) sizes must match when you connect them over MPLS. By default, the MTU size associated with a Frame Relay DLCI is the interface MTU. This may cause problems, for example, when connecting some DLCIs on a PoS interface (with a default MTU of 4470 bytes) to Ethernet or VLAN (with a default MTU of 1500 bytes) and other DLCIs on the same PoS interface to ATM (with a default MTU of 4470 bytes). To avoid reducing all the interface MTUs to the lowest common denominator (1500 bytes in this case), you can specify the MTU for individual DLCIs using the **mtu** command.
- Only DLCI mode is supported. Port mode is not supported.
- Configure Frame Relay switching to use DCE or Network-to-Network Interface (NNI). DTE mode does not report status in the Local Management Interface (LMI) process. If a Frame Relay over MPLS circuit goes down and the PE router is in DTE mode, the CE router is never informed of the disabled circuit. You must configure the **frame-relay switching** command in global configuration mode in order to configure DCE or NNI.
- Frame Relay policing is non-distributed on the Cisco 7500 series routers. If you enable Frame Relay policing, traffic is sent to the route switch processor for processing.
- Inverse ARP is not supported with IP interworking. CE routers must use point-to-point subinterfaces or static maps.
- The PE router automatically supports translation of both the Cisco encapsulations and the Internet Engineering Task Force (IETF) encapsulations that come from the CE, but translates only to IETF when sending to the CE router. This is not a problem for the Cisco CE router, because it can handle IETF encapsulation on receipt even if it is configured to send Cisco encapsulation.
- With Ethernet interworking, the following translations are supported:
    - Ethernet without LAN FCS (0300800080C20007 or 6558)
    - Spanning tree (0300800080C2000E or 4242)

All other translations are dropped.

- With IP interworking, the IPv4 (03CC or 0800) translation is supported. All other translations are dropped.
- PVC status signaling works the same way as in like-to-like case. The PE router reports the PVC status to the CE router, based on the availability of the pseudowire. PVC status detected by the PE router will also be reflected into the pseudowire. LMI to OAM interworking is supported when you connect Frame Relay to ATM.

PPP Interworking Restrictions

The following restrictions apply to PPP interworking:

- There must be a one-to-one relationship between a PPP session and the pseudowire. Multiplexing of multiple PPP sessions over the pseudowire is not supported.
- There must be a one-to-one relationship between a PPP session and a Frame Relay DLCI. Each Frame Relay PVC must have only one PPP session.
- Only IP (IPv4 (0021) interworking is supported. Link Control Protocol (LCP) packets and Internet Protocol Control Protocol (IPCP) packets are terminated at the PE router. Everything else is dropped.
- Proxy IPCP is automatically enabled on the PE router when IP interworking is configured on the pseudowire.
- By default, the PE router assumes that the CE router knows the remote CE router’s IP address.

- Password Authentication Protocol (PAP) and Challenge-Handshake Authentication Protocol (CHAP) authentication are supported.

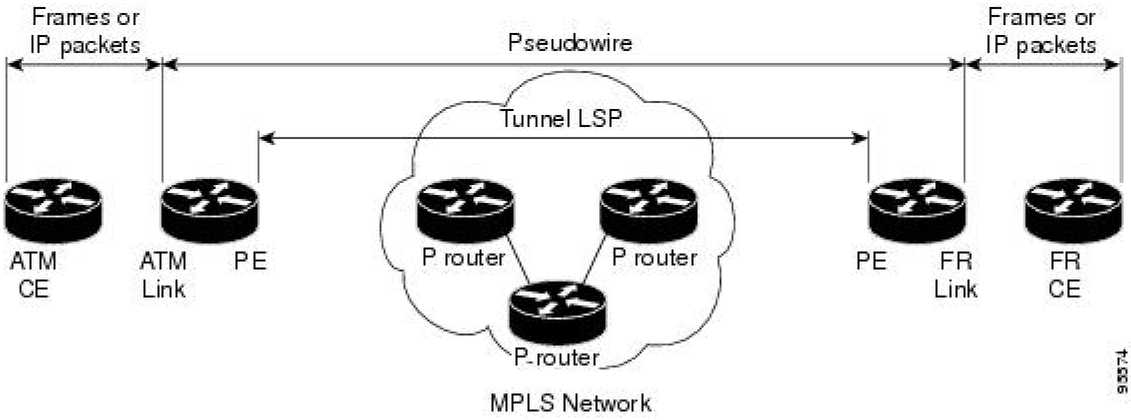
Information About L2VPN Interworking

- Overview of L2VPN Interworking, page 139
- L2VPN Interworking Modes, page 139
- L2VPN Interworking Support Matrix, page 141
- Static IP Addresses for L2VPN Interworking for PPP, page 142

Overview of L2VPN Interworking

Layer 2 transport over MPLS and IP already exists for like-to-like attachment circuits, such as Ethernet-to-Ethernet or PPP-to-PPP. L2VPN Interworking builds on this functionality by allowing disparate attachment circuits to be connected. An interworking function facilitates the translation between the different Layer 2 encapsulations. The figure below is an example of Layer 2 interworking, where ATM and Frame Relay packets travel over the MPLS cloud.

Figure 4 ATM to Frame Relay Interworking Example



The L2VPN Interworking feature supports Ethernet, 802.1Q (VLAN), Frame Relay, ATM AAL5, and PPP attachment circuits over MPLS and L2TPv3. The features and restrictions for like-to-like functionality also apply to L2VPN Interworking.

L2VPN Interworking Modes

L2VPN Interworking works in either Ethernet (“bridged”) mode, IP (“routed”), or Ethernet VLAN mode. You specify the mode by issuing the **interworking** {**ethernet** | **ip** |**vlan**} command in pseudowire-class configuration mode.

- Ethernet (Bridged) Interworking, page 140
- IP (Routed) Interworking, page 140
- VLAN Interworking, page 141

Ethernet (Bridged) Interworking

The **ethernet** keyword causes Ethernet frames to be extracted from the attachment circuit and sent over the pseudowire. Ethernet end-to-end transmission is assumed. Attachment circuit frames that are not Ethernet are dropped. In the case of VLAN, the VLAN tag is removed, leaving an untagged Ethernet frame.

Ethernet Interworking is also called bridged interworking. Ethernet frames are bridged across the pseudowire. The CE routers could be natively bridging Ethernet or could be routing using a bridged encapsulation model, such as Bridge Virtual Interface (BVI) or RBE. The PE routers operate in Ethernet like-to-like mode.

This mode is used to offer the following services:

- LAN services--An example is an enterprise that has several sites, where some sites have Ethernet connectivity to the service provider (SP) network and others have ATM connectivity. The enterprise wants LAN connectivity to all its sites. In this case, traffic from the Ethernet or VLAN of one site can be sent through the IP/MPLS network and encapsulated as bridged traffic over an ATM VC of another site.
- Connectivity services--An example is an enterprise that has different sites that are running an Internal Gateway Protocol (IGP) routing protocol, which has incompatible procedures on broadcast and nonbroadcast links. The enterprise has several sites that are running an IGP, such as Open Shortest Path First (OSPF) or Intermediate System to Intermediate System (IS-IS), between the sites. In this scenario, some of the procedures (such as route advertisement or designated router) depend on the underlying Layer 2 protocol and are different for a point-to-point ATM connection versus a broadcast Ethernet connection. Therefore, the bridged encapsulation over ATM can be used to achieve homogenous Ethernet connectivity between the CE routers running the IGP.

IP (Routed) Interworking

The **ip** keyword causes IP packets to be extracted from the attachment circuit and sent over the pseudowire. Attachment circuit frames that do not contain IPv4 packets are dropped.

IP Interworking is also called routed interworking. The CE routers encapsulate IP on the link between the CE and PE routers. A new VC type is used to signal the IP pseudowire in MPLS and L2TPv3. Translation between the Layer 2 and IP encapsulations across the pseudowire is required. Special consideration needs to be given to address resolution and routing protocol operation, because these are handled differently on different Layer 2 encapsulations.

This mode is used to provide IP connectivity between sites, regardless of the Layer 2 connectivity to these sites. It is different from a Layer 3 VPN because it is point-to-point in nature and the service provider does not maintain any customer routing information.

Address resolution is encapsulation dependent:

- Ethernet uses ARP
- Frame Relay and ATM use Inverse ARP
- PPP uses IPCP

Therefore, address resolution must be terminated on the PE router. End-to-end address resolution is not supported. Routing protocols operate differently over broadcast and point-to-point media. For Ethernet, the CE routers must either use static routing or configure the routing protocols to treat the Ethernet side as a point-to-point network.

VLAN Interworking

The **vlan** keyword allows the VLAN ID to be included as part of the Ethernet frame. In Cisco IOS Release 12.2(52)SE, you can configure Catalyst 3750 Metro switches to use Ethernet VLAN for Ethernet (bridged) interworking. You can specify the Ethernet VLAN (type 4) by issuing the **interworking vlan** command in pseudowire-class configuration mode. This allows the VLAN ID to be included as part of the Ethernet frame. In releases previous to Cisco IOS Release 12.2(52)SE, the only way to achieve VLAN encapsulation is to ensure the CE router is connected to the PE router through an Ethernet VLAN interface/subinterface.

L2VPN Interworking Support Matrix

The supported L2VPN Interworking features are listed in the table below.

Table 16 L2VPN Interworking Supported Features

Feature	MPLS or L2TPv3 Support	IP or Ethernet Support
Ethernet/VLAN to ATM AAL5	MPLS L2TPv3 (12000 series only)	IP Ethernet
Ethernet/VLAN to Frame Relay	MPLS L2TPv3	IP Ethernet
Ethernet/VLAN to PPP	MPLS	IP
Ethernet to VLAN	MPLS L2TPv3	IP Ethernet[1]
L2VPN Interworking: VLAN Enable/Disable Option for AToM	MPLS	Ethernet VLAN
Frame Relay to ATM AAL5	MPLS L2TPv3 (12000 series only)	IP
Frame Relay to Ethernet or VLAN	MPLS L2TPv3	IP Ethernet
Frame Relay to PPP	MPLS L2TPv3	IP
Note : On the Cisco 12000 series Internet router: • Ethernet (bridged) interworking is not supported for L2TPv3. • IP (routed) interworking is not supported in an L2TPv3 pseudowire configured for data sequencing (using the **sequencing** command).		

[1] With the L2VPN Interworking: VLAN Enable/Disable Option for AToM feature, VLAN interworking can also be supported. For more information, see the “VLAN Interworking” section on page 14 .

Static IP Addresses for L2VPN Interworking for PPP

If the PE router needs to perform address resolution with the local CE router for PPP, you can configure the remote CE router's IP address on the PE router. Issue the **ppp ipcp address proxy** command with the remote CE router's IP address on the PE router's xconnect PPP interface. The following example shows a sample configuration:

```
pseudowire-class ip-interworking
 encapsulation mpls
 interworking ip
interface Serial2/0
 encapsulation ppp
 xconnect 10.0.0.2 200 pw-class ip-interworking
 ppp ipcp address proxy 10.65.32.14
```

You can also configure the remote CE router's IP address on the local CE router with the **peer default ip address** command if the local CE router performs address resolution.

How to Configure L2VPN Interworking

- Configuring L2VPN Interworking, page 142
- Verifying the L2VPN Interworking Configuration, page 143
- Configuring L2VPN Interworking: VLAN Enable-Disable Option for AToM, page 147

Configuring L2VPN Interworking

L2VPN Interworking allows you to connect disparate attachment circuits. Configuring the L2VPN Interworking feature requires that you add the **interworking** command to the list of commands that make up the pseudowire. The steps for configuring the pseudowire for L2VPN Interworking are included in this section. You use the **interworking**command as part of the overall AToM or L2TPv3 configuration. For specific instructions on configuring AToM or L2TPv3, see the following documents:

- Layer 2 Tunnel Protocol Version 3
- Any Transport over MPLS

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **hw-module slot** *slot-number* **np mode feature**
4. **pseudowire-class** *name*
5. **encapsulation** {**mpls** | **l2tpv3**}
6. **interworking** {**ethernet** | **ip**} | **vlan**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. - Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**hw-module slot** *slot-number* **np mode feature** **Example:** `Router(config)# hw-module slot 3 np mode feature`	(Optional) Enables L2VPN Interworking functionality on the Cisco 12000 series router. **Note** Enter this command only on a Cisco 12000 series Internet router if you use L2TPv3 for L2VPN Interworking on an ISE (Engine 3) or Engine 5 interface. In this case, you must first enable the L2VPN feature bundle on the line card by entering the **hw-module slot** *slot-number* **np mode feature** command.
Step 4	**pseudowire-class** *name* **Example:** `Router(config)# pseudowire-class class1`	Establishes a pseudowire class with a name that you specify and enters pseudowire class configuration mode.

Step 5	**encapsulation** {**mpls** | **l2tpv3**}
	Example:
	`Router(config-pw)# encapsulation mpls`
		Specifies the tunneling encapsulation, which is either **mpls** or **l2tpv3**.

Step 6	**interworking** {**ethernet** | **ip**} | **vlan**}
	Example:
	`Router(config-pw)# interworking ip`
		Specifies the type of pseudowire and the type of traffic that can flow across it.
		Note On the Cisco 12000 series Internet router, Ethernet (bridged) interworking is not supported for L2TPv3. After you configure the L2TPv3 tunnel encapsulation for the pseudowire using the **encapsulation l2tpv3**command, you cannot enter the **interworking ethernet** command.

Verifying the L2VPN Interworking Configuration

To verify the L2VPN Interworking configuration, you can use the following commands.

SUMMARY STEPS

1. **enable**
2. **show l2tun session all (L2TPv3 only)**
3. **show arp**
4. **ping**
5. **show l2tun session interworking (L2TPv3 only)**
6. **show mpls l2transport vc detail (AToM only)**

DETAILED STEPS

Step 1 **enable**

Enables privileged EXEC mode. Enter your password if prompted.

Step 2 **show l2tun session all (L2TPv3 only)**

For L2TPv3, you can verify the L2VPN Interworking configuration using the **show l2tun session all** command on the PE routers.

In the following example, the interworking type is shown in bold.

PE1	PE2
Router# **show l2tun session all**	Router# **show l2tun session all**
Session Information Total tunnels 1 sessions 1	Session Information Total tunnels 1 sessions 1
Session id 15736 is up, tunnel id 35411	Session id 26570 is up, tunnel id 46882
Call serial number is 4035100045	Call serial number is 4035100045
Remote tunnel name is PE2	Remote tunnel name is PE1
Internet address is 10.9.9.9	Internet address is 10.8.8.8
Session is L2TP signalled	Session is L2TP signalled
Session state is established, time since change 1d22h	Session state is established, time since change 1d22h
16 Packets sent, 16 received	16 Packets sent, 16 received
1518 Bytes sent, 1230 received	1230 Bytes sent, 1230 received
Receive packets dropped:	Receive packets dropped:
out-of-order: 0	out-of-order: 0
total: 0	total: 0
Send packets dropped:	Send packets dropped:
exceeded session MTU: 0	exceeded session MTU: 0
total: 0	total: 0
Session vcid is 123	Session vcid is 123
Session Layer 2 circuit, type is Ethernet, name is FastEthernet1/1/0	Session Layer 2 circuit, type is Ethernet Vlan, name is FastEthernet2/0.1:10
Circuit state is UP	Circuit state is UP, **interworking type is Ethernet**
Remote session id is 26570, remote tunnel id 46882	Remote session id is 15736, remote tunnel id 35411
DF bit off, ToS reflect disabled, ToS value 0, TTL value 255	DF bit off, ToS reflect disabled, ToS value 0, TTL value 255
No session cookie information available	No session cookie information available
FS cached header information:	FS cached header information:
encap size = 24 bytes	encap size = 24 bytes
00000000 00000000 00000000 00000000	00000000 00000000 00000000 00000000
00000000 00000000	00000000 00000000
Sequencing is off	Sequencing is off

You can issue the **show arp** command between the CE routers to ensure that data is being sent:

Example:

```
Router# show arp
Protocol   Address       Age (min)   Hardware Addr    Type    Interface
Internet   10.1.1.5           134    0005.0032.0854   ARPA    FastEthernet0/0
Internet   10.1.1.7             -    0005.0032.0000   ARPA    FastEthernet0/0
```

Step 4 **ping**

You can issue the **ping** command between the CE routers to ensure that data is being sent:

Example:

```
Router# ping 10.1.1.5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.5, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
```

Step 5 **show l2tun session interworking (L2TPv3 only)**

For L2TPv3, you can verify that the interworking type is correctly set using the **show l2tun session interworking** command. Enter the command on the PE routers that are performing the interworking translation.

- In Example 1, the PE router performs the raw Ethernet translation. The command output displays the interworking type with a dash (-).
- In Example 2, the PE router performs the Ethernet VLAN translation. The command output displays the interworking type as ETH.

Command Output for Raw Ethernet Translation

Example:

```
Router# show l2tun session interworking
Session Information Total tunnels 1 sessions 1
LocID      TunID      Peer-address    Type IWrk Username, Intf/Vcid, Circuit
15736      35411      10.9.9.9         ETH   -   123,      Fa1/1/0
```

Command Output for Ethernet VLAN Translation

Example:

```
Router# show l2tun session interworking
Session Information Total tunnels 1 sessions 1
LocID      TunID      Peer-address    Type IWrk Username, Intf/Vcid, Circuit
26570      46882      10.8.8.8         VLAN ETH  123,      Fa2/0.1:10
```

Step 6 **show mpls l2transport vc detail (AToM only)**

You can verify the AToM configuration by using the **show mpls l2transport vc detail** command. In the following example, the interworking type is shown in bold.

PE1	PE2
Router# **show mpls l2transport vc detail** Local interface: Fa1/1/0 up, line protocol up, Ethernet up Destination address: 10.9.9.9, VC ID: 123, VC status: up Preferred path: not configured Default path: active Tunnel label: 17, next hop 10.1.1.3 Output interface: Fa4/0/0, imposed label stack {17 20} Create time: 01:43:50, last status change time: 01:43:33 Signaling protocol: LDP, peer 10.9.9.9:0 up MPLS VC labels: local 16, remote 20 Group ID: local 0, remote 0 MTU: local 1500, remote 1500 Remote interface description: Sequencing: receive disabled, send disabled VC statistics: packet totals: receive 15, send 4184 byte totals: receive 1830, send 309248 packet drops: receive 0, send 0	Router# **show mpls l2transport vc detail** Local interface: Fa2/0.3 up, line protocol up, Eth VLAN 10 up MPLS VC type is Ethernet, **interworking type is Ethernet** Destination address: 10.8.8.8, VC ID: 123, VC status: up Preferred path: not configured Default path: active Tunnel label: 16, next hop 10.1.1.3 Output interface: Fa6/0, imposed label stack {16 16} Create time: 00:00:26, last status change time: 00:00:06 Signaling protocol: LDP, peer 10.8.8.8:0 up MPLS VC labels: local 20, remote 16 Group ID: local 0, remote 0 MTU: local 1500, remote 1500 Remote interface description: Sequencing: receive disabled, send disabled VC statistics: packet totals: receive 5, send 0 byte totals: receive 340, send 0 packet drops: receive 0, send 0

Configuring L2VPN Interworking: VLAN Enable-Disable Option for AToM

You can specify the Ethernet VLAN (type 4) by issuing the **interworking vlan** command in pseudowire-class configuration mode. This allows the VLAN ID to be included as part of the Ethernet frame. In releases previous to Cisco IOS Release 12.2(52)SE and Cisco IOS Release 12.2(33)SRE, the only way to achieve VLAN encapsulation is to ensure the CE router is connected to the PE router through an Ethernet link.

For complete instructions on configuring AToM, see "Any Transport over MPLS".

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **pseudowire-class** *name*
4. **encapsulation** {**mpls** | **l2tpv3**}
5. **interworking** {**ethernet** | **ip**| **vlan**}
6. **end**
7. **show mpls l2transport vc** [**vcid** *vc-id* | **vcid** *vc-id-min vc-id-max*] [**interface** *type number* [*local-circuit-id*]] [**destination** *ip-address* | *name*] [**detail**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**pseudowire-class** *name* **Example:** `Router(config)# pseudowire-class class1`	Establishes a pseudowire class with a name that you specify and enters pseudowire class configuration mode.

Step 4	**encapsulation** {**mpls** | **l2tpv3**}
	Example:
	`Router(config-pw)# encapsulation mpls`
		Specifies the tunneling encapsulation, which is either **mpls** or **l2tpv3**.
		• For the L2VPN Internetworking: VLAN Enable/Disable option for AToM feature, only MPLS encapsulation is supported.

Step 5	**interworking** {**ethernet** | **ip**| **vlan**}
	Example:
	`Router(config-pw)# interworking vlan`
		Specifies the type of pseudowire and the type of traffic that can flow across it.
		• For the L2VPN Internetworking: VLAN Enable/Disable option for AToM feature, specify the **vlan** keyword.

	Command or Action	Purpose

Step 6	**end**
	Example:
	`Router(config-pw)# end`
		Exits pseudowire class configuration mode and enters privileged EXEC mode.

Step 7	**show mpls l2transport vc** [**vcid** *vc-id* | **vcid** *vc-id-min vc-id-max*] [**interface** *type number* [*local-circuit-id*]] [**destination** *ip-address* | *name*] [**detail**]
	Example:
	`Router#`
	`show mpls l2transport vc detail`
		Displays information about AToM VCs.

Examples

When the pseudowire on an interface is different from the VC type, the interworking type is displayed in the **show mpls l2transport vc detail** command output. In the following example, the pseudowire is configured on an Ethernet port and VLAN interworking is configured in the pseudowire class. The relevant output is shown in bold:

```
PE1# show mpls l2 vc 34 detail
Local interface: Et0/1 up, line protocol up, Ethernet up
  MPLS VC type is Ethernet, interworking type is Eth VLAN
  Destination address: 10.1.1.2, VC ID: 34, VC status: down
    Output interface: if-?(0), imposed label stack {}
    Preferred path: not configured
    Default path: no route
    No adjacency
  Create time: 00:00:13, last status change time: 00:00:13
  Signaling protocol: LDP, peer unknown
    Targeted Hello: 10.1.1.1(LDP Id) -> 10.1.1.2
    Status TLV support (local/remote)   : enabled/None (no remote binding)
      LDP route watch                   : enabled
      Label/status state machine        : local standby, AC-ready, LnuRnd
      Last local dataplane   status rcvd: No fault
      Last local SSS circuit status rcvd: No fault
      Last local SSS circuit status sent: Not sent
      Last local  LDP TLV    status sent: None
      Last remote LDP TLV    status rcvd: None (no remote binding)
      Last remote LDP ADJ    status rcvd: None (no remote binding)
    MPLS VC labels: local 2003, remote unassigned
    Group ID: local 0, remote unknown
    MTU: local 1500, remote unknown
    Remote interface description:
  Sequencing: receive disabled, send disabled
  VC statistics:
    packet totals: receive 0, send 0
    byte totals:   receive 0, send 0
    packet drops:  receive 0, seq error 0, send 0
```

Configuration Examples for L2VPN Interworking

- Ethernet to VLAN over L2TPV3 (Bridged) Example, page 150

- Ethernet to VLAN over AToM (Bridged) Example, page 151
- Frame Relay to VLAN over L2TPV3 (Routed) Example, page 152
- Frame Relay to VLAN over AToM (Routed) Example, page 154
- Frame Relay to ATM AAL5 over AToM (Routed) Example, page 155
- VLAN to ATM AAL5 over AToM (Bridged) Example, page 156
- Frame Relay to PPP over L2TPv3 (Routed) Example, page 157
- Frame Relay to PPP over AToM (Routed) Example, page 159
- Ethernet VLAN to PPP over AToM (Routed) Example, page 161

Ethernet to VLAN over L2TPV3 (Bridged) Example

The following example shows the configuration of Ethernet to VLAN over L2TPv3:

PE1	PE2
`ip cef`	`ip cef`
`!`	`!`
`l2tp-class interworking-class`	`l2tp-class interworking-class`
`authentication`	`authentication`
`hostname PE1`	`hostname PE2`
`password 0 lab`	`password 0 lab`
`!`	`!`
`pseudowire-class inter-ether-vlan`	`pseudowire-class inter-ether-vlan`
`encapsulation l2tpv3`	`encapsulation l2tpv3`
`interworking ethernet`	`interworking ethernet`
`protocol l2tpv3 interworking-class`	`protocol l2tpv3 interworking-class`
`ip local interface Loopback0`	`ip local interface Loopback0`
`!`	`!`
`interface Loopback0`	`interface Loopback0`
`ip address 10.8.8.8 255.255.255.255`	`ip address 10.9.9.9 255.255.255.255`
`!`	`!`
`interface FastEthernet1/0`	`interface FastEthernet0/0`
`xconnect 10.9.9.9 1 pw-class inter-ether-vlan`	`no ip address`
	`!`
	`interface FastEthernet0/0.3`
	`encapsulation dot1Q 10`
	`xconnect 10.8.8.8 1 pw-class inter-ether-vlan`

Ethernet to VLAN over AToM (Bridged) Example

The following example shows the configuration of Ethernet to VLAN over AToM:

PE1	PE2
ip cef	ip cef
!	!
mpls label protocol ldp	mpls label protocol ldp
mpls ldp router-id Loopback0 force	mpls ldp router-id Loopback0 force
!	!
pseudowire-class atom-eth-iw	pseudowire-class atom
encapsulation mpls	encapsulation mpls
interworking ethernet	!
!	interface Loopback0
interface Loopback0	ip address 10.9.9.9 255.255.255.255
ip address 10.8.8.8 255.255.255.255	!
!	interface FastEthernet0/0
interface FastEthernet1/0.1	no ip address
encapsulation dot1q 100	!
xconnect 10.9.9.9 123 pw-class atom-eth-iw	interface FastEthernet1/0
	xconnect 10.9.9.9 123 pw-class atom

Frame Relay to VLAN over L2TPV3 (Routed) Example

The following example shows the configuration of Frame Relay to VLAN over L2TPv3:

PE1	PE2
configure terminal	configure terminal
ip cef	ip routing
frame-relay switching	ip cef
!	frame-relay switching
!	!
interface loopback 0	interface loopback 0
ip address 10.8.8.8 255.255.255.255	ip address 10.9.9.9 255.255.255.255
no shutdown	no shutdown
!	!
pseudowire-class ip	pseudowire-class ip
encapsulation l2tpv3	encapsulation l2tpv3
interworking ip	interworking ip
ip local interface loopback0	ip local interface loopback0
!	!
interface POS1/0	interface FastEthernet1/0/1
encapsulation frame-relay	speed 10
clock source internal	no shutdown
logging event dlci-status-change	!
no shutdown	interface FastEthernet1/0/1.6
no fair-queue	encapsulation dot1Q 6
!	xconnect 10.8.8.8 6 pw-class ip
connect fr-vlan POS1/0 206 l2transport	no shutdown
xconnect 10.9.9.9 6 pw-class ip	!
!	router ospf 10
router ospf 10	network 10.0.0.2 0.0.0.0 area 0
network 10.0.0.2 0.0.0.0 area 0	network 10.9.9.9 0.0.0.0 area 0
network 10.8.8.8 0.0.0.0 area 0	

Frame Relay to VLAN over AToM (Routed) Example

The following example shows the configuration of Frame Relay to VLAN over AToM:

PE1	PE2
`configure terminal`	`configure terminal`
`ip cef`	`ip routing`
`frame-relay switching`	`ip cef`
`!`	`frame-relay switching`
`mpls label protocol ldp`	`!`
`mpls ldp router-id loopback0`	`mpls label protocol ldp`
`mpls ip`	`mpls ldp router-id loopback0`
`!`	`mpls ip`
`pseudowire-class atom`	`!`
` encapsulation mpls`	`pseudowire-class atom`
` interworking ip`	` encapsulation mpls`
`!`	` interworking ip`
`interface loopback 0`	`!`
` ip address 10.8.8.8 255.255.255.255`	`interface loopback 0`
` no shutdown`	` ip address 10.9.9.9 255.255.255.255`
`!`	` no shutdown`
`connect fr-vlan POS1/0 206 l2transport`	`!`
` xconnect 10.9.9.9 6 pw-class atom`	`interface FastEthernet1/0/1.6`
	` encapsulation dot1Q 6`
	` xconnect 10.8.8.8 6 pw-class atom`
	` no shutdown`

Frame Relay to ATM AAL5 over AToM (Routed) Example

Note Frame Relay to ATM AAL5 is available only with AToM in IP mode.

The following example shows the configuration of Frame Relay to ATM AAL5 over AToM:

PE1	PE2
ip cef	ip cef
frame-relay switching	mpls ip
mpls ip	mpls label protocol ldp
mpls label protocol ldp	mpls ldp router-id loopback0 force
mpls ldp router-id loopback0 force	pseudowire-class fratmip
pseudowire-class fratmip	encapsulation mpls
encapsulation mpls	interworking ip
interworking ip	interface Loopback0
interface Loopback0	ip address 10.22.22.22 255.255.255.255
ip address 10.33.33.33 255.255.255.255	interface ATM 2/0
interface serial 2/0	pvc 0/203 l2transport
encapsulation frame-relay ietf	encapsulation aa5snap
frame-relay intf-type dce	xconnect 10.33.33.33 333 pw-class fratmip
connect fr-eth serial 2/0 100 l2transport	interface POS1/0
xconnect 10.22.22.22 333 pw-class fratmip	ip address 10.1.1.2 255.255.255.0
interface POS1/0	crc 32
ip address 10.1.7.3 255.255.255.0	clock source internal
crc 32	mpls ip
clock source internal	mpls label protocol ldp
mpls ip	router ospf 10
mpls label protocol ldp	passive-interface Loopback0
router ospf 10	network 10.22.22.22 0.0.0.0 area 10
passive-interface Loopback0	network 10.1.1.0 0.0.0.255 area 10
network 10.33.33.33 0.0.0.0 area 10	
network 10.1.7.0 0.0.0.255 area 10	

VLAN to ATM AAL5 over AToM (Bridged) Example

The following example shows the configuration of VLAN to ATM AAL5 over AToM:

PE1	PE2
`ip cef`	`ip cef`
`!`	`!`
`mpls ip`	`mpls ip`
`mpls label protocol ldp`	`mpls label protocol ldp`
`mpls ldp router-id Loopback0`	`mpls ldp router-id Loopback0`
`!`	`!`
`pseudowire-class inter-ether`	`pseudowire-class inter-ether`
`encapsulation mpls`	` encapsulation mpls`
`interworking ethernet`	` interworking ethernet`
`!`	`!`
`interface Loopback0`	`interface Loopback0`
` ip address 10.8.8.8 255.255.255.255`	` ip address 10.9.9.9 255.255.255.255`
`!`	`!`
`interface ATM1/0.1 point-to-point`	`interface FastEthernet0/0`
`pvc 0/100 l2transport`	` no ip address`
`encapsulation aal5snap`	`!`
`xconnect 10.9.9.9 123 pw-class inter-ether`	`interface FastEthernet0/0.1`
`!`	`encapsulation dot1Q 10`
`interface FastEthernet1/0`	`xconnect 10.8.8.8 123 pw-class inter-ether`
`xconnect 10.9.9.9 1 pw-class inter-ether`	`!`
`!`	`router ospf 10`
`router ospf 10`	` log-adjacency-changes`
` log-adjacency-changes`	` network 10.9.9.9 0.0.0.0 area 0`
` network 10.8.8.8 0.0.0.0 area 0`	` network 10.1.1.2 0.0.0.0 area 0`
` network 10.1.1.1 0.0.0.0 area 0`	

Frame Relay to PPP over L2TPv3 (Routed) Example

The following example shows the configuration of Frame Relay to PPP over L2TPv3:

PE1	PE2
ip cef	ip cef
ip routing	ip routing
!	!
!	frame-relay switching
!	!
pseudowire-class ppp-fr	pseudowire-class ppp-fr
encapsulation l2tpv3	encapsulation l2tpv3
interworking ip	interworking ip
ip local interface Loopback0	ip local interface Loopback0
!	!
interface Loopback0	interface Loopback0
ip address 10.1.1.1 255.255.255.255	ip address 10.2.2.2 255.255.255.255
!	!
interface FastEthernet1/0/0	interface FastEthernet1/0/0
ip address 10.16.1.1 255.255.255.0	ip address 10.16.2.1 255.255.255.0
!	!
interface Serial3/0/0	interface Serial3/0/0
no ip address	no ip address
encapsulation ppp	encapsulation frame-relay
ppp authentication chap	frame-relay intf-type dce
!	!
ip route 10.0.0.0 255.0.0.0 10.16.1.2	ip route 10.0.0.0 255.0.0.0 10.16.2.2
!	!
xconnect 10.2.2.2 1 pw-class ppp-fr	connect ppp-fr Serial3/0/0 100 l2transport
ppp ipcp address proxy 10.65.32.14	xconnect 10.1.1.1 100 pw-class ppp-fr

Frame Relay to PPP over AToM (Routed) Example

The following example shows the configuration of Frame Relay to PPP over AToM:

PE1	PE2
`ip cef`	`ip cef`
`ip routing`	`ip routing`
`mpls label protocol ldp`	`mpls label protocol ldp`
`mpls ldp router-id loopback0 force`	`mpls ldp router-id loopback0 force`
`!`	`!`
`!`	`frame-relay switching`
`!`	`!`
`pseudowire-class ppp-fr`	`pseudowire-class ppp-fr`
`encapsulation mpls`	`encapsulation mpls`
`interworking ip`	`interworking ip`
`ip local interface Loopback0`	`ip local interface Loopback0`
`!`	`!`
`interface Loopback0`	`interface Loopback0`
` ip address 10.1.1.1 255.255.255.255`	` ip address 10.2.2.2 255.255.255.255`
`!`	`!`
`interface FastEthernet1/0/0`	`interface FastEthernet1/0/0`
`ip address 10.16.1.1 255.255.255.0`	`ip address 10.16.2.1 255.255.255.0`
`mpls ip`	`mpls ip`
`label protocol ldp`	`mpls label protocol ldp`
`!`	`!`
`interface Serial3/0/0`	`interface Serial3/0/0`
` no ip address`	`no ip address`
` encapsulation ppp`	`encapsulation frame-relay`
` ppp authentication chap`	`frame-relay intf-type dce`
` xconnect 10.2.2.2 1 pw-class ppp-fr`	`!`
`ppp ipcp address proxy 10.65.32.14`	`ip route 10.0.0.0 255.0.0.0 10.16.2.2`
`!`	`!`
`ip route 10.0.0.0 255.0.0.0 10.16.1.2`	`connect ppp-fr Serial3/0/0 100 l2transport`
	` xconnect 10.1.1.1 100 pw-class ppp-fr`

Ethernet VLAN to PPP over AToM (Routed) Example

The following example shows the configuration of Ethernet VLAN to PPP over AToM:

PE1	PE2
`configure terminal`	`configure terminal`
`mpls label protocol ldp`	`mpls label protocol ldp`
`mpls ldp router-id Loopback0`	`mpls ldp router-id Loopback0`
`mpls ip`	`mpls ip`
`!`	`!`
`pseudowire-class ppp-ether`	`pseudowire-class ppp-ether`
` encapsulation mpls`	` encapsulation mpls`
` interworking ip`	` interworking ip`
`!`	`!`
`interface Loopback0`	`interface Loopback0`
` ip address 10.8.8.8 255.255.255.255`	` ip address 10.9.9.9 255.255.255.255`
` no shutdown`	` no shutdown`
`!`	`!`
`interface POS2/0/1`	`interface vlan300`
` no ip address`	` mtu 4470`
` encapsulation ppp`	` no ip address`
` no peer default ip address`	` xconnect 10.8.8.8 300 pw-class ppp-ether`
` ppp ipcp address proxy 10.10.10.1`	` no shutdown`
` xconnect 10.9.9.9 300 pw-class ppp-ether`	`!`
` no shutdown`	`interface GigabitEthernet6/2`
	` switchport`
	` switchport trunk encapsulation dot1q`
	` switchport trunk allowed vlan 300`
	` switchport mode trunk`
	` no shutdown`

Additional References

The following sections provide references related to the L2VPN Interworking feature.

Related Documents

Related Topic	Document Title
Layer 2 Tunnel Protocol Version 3	Layer 2 Tunnel Protocol Version 3
Any Transport over MPLS	Any Transport over MPLS
Cisco 12000 series routers hardware support	http://www.cisco.com/univercd/cc/td/doc/product/core/cis12000/linecard/lc_spa/spa_swcs/1232sy/index.htm http://www.cisco.com/en/US/products/sw/iosswrel/ps1829/prod_release_notes_list.html Cross-Platform Release Notes for Cisco IOS Release 12.0S.
Cisco 7600 series routers hardware support	• Release Notes for Cisco IOS Release 12.2SR for the Cisco 7600 Series Routers
Cisco 3270 series routers hardware support	Cisco IOS Software Releases 12.2SE Release Notes

Standards

Standards	Title
draft-ietf-l2tpext-l2tp-base-03.txt	*Layer Two Tunneling Protocol (Version 3) 'L2TPv3'*
draft-martini-l2circuit-trans-mpls-09.txt	*Transport of Layer 2 Frames Over MPLS*
draft-ietf-pwe3-frame-relay-03.txt.	*Encapsulation Methods for Transport of Frame Relay over MPLS Networks*
draft-martini-l2circuit-encap-mpls-04.txt.	*Encapsulation Methods for Transport of Layer 2 Frames Over IP and MPLS Networks*
draft-ietf-pwe3-ethernet-encap-08.txt.	*Encapsulation Methods for Transport of Ethernet over MPLS Networks*
draft-ietf-pwe3-hdlc-ppp-encap-mpls-03.txt.	*Encapsulation Methods for Transport of PPP/HDLC over MPLS Networks*
draft-ietf-ppvpn-l2vpn-00.txt.	*An Architecture for L2VPNs*

MIBs

MIBs	MIBs Link
No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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.	--

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

Feature Information for L2VPN Interworking

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 17 Feature Information for L2VPN Interworking

Feature Name	Releases	Feature Information
L2VPN Interworking	12.0(26)S 12.0(30)S 12.0(32)S 12.0(32)SY 12.2(33)SRA 12.4(11)T 12.2(33)SXH 12.2(33)SRD 12.2(52)SE 12.2(33)SRE	This feature allows disparate attachment circuits to be connected. An interworking function facilitates the translation between the different Layer 2 encapsulations. This feature was introduced in Cisco IOS Release 12.0(26)S. In Cisco IOS Release 12.0(30)S, support was added for Cisco 12000 series Internet routers. In Cisco IOS Release 12.0(32)S, support was added on Engine 5 line cards (SIP-401, SIP-501, SIP-600, and SIP-601) in Cisco 12000 series routers for the following four transport types: • Ethernet/VLAN to Frame Relay Interworking • Ethernet/VLAN to ATM AAL5 Interworking • Ethernet to VLAN Interworking • Frame Relay to ATM AAL5 Interworking On the Cisco 12000 series Internet router, support was added for IP Services Engine (ISE) and Engine 5 line cards that are configured for L2TPv3 tunneling. In Cisco IOS Release 12.2(33)SRA, support was added for the Cisco 7600 series routers. In Cisco IOS Release 12.4(11)T, support was added for the following transport types: • Ethernet to VLAN Interworking • Ethernet/VLAN to Frame Relay Interworking This feature was integrated into Cisco IOS Release 12.2(33)SXH. In Cisco IOS Release 12.2(33)SRD, support for routed

Feature Name	Releases	Feature Information
		and bridged interworking on SIP-400 was added for the Cisco 7600 series routers. In Cisco IOS Release 12.2(52)SE, the L2VPN Internetworking: VLAN Enable/Disable option for AToM feature was added for the Cisco 3750 Metro switch. In Cisco IOS Release 12.2(33)SRE, the L2VPN Internetworking: VLAN Enable/ Disable option for AToM feature was added for the Cisco 7600 series router. The following commands were introduced or modified: **interworking**

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.

L2VPN Pseudowire Switching

This feature module explains how to configure L2VPN Pseudowire Switching, which extends Layer 2 Virtual Private Network (L2VPN) pseudowires across an interautonomous system (inter-AS) boundary or across two separate Multiprotocol Label Switching (MPLS) networks. The feature supports ATM and time-division multiplexing (TDM) attachment circuits (ACs) and Ethernet ACs.

- Finding Feature Information, page 167
- Prerequisites for L2VPN Pseudowire Switching, page 167
- Restrictions for L2VPN Pseudowire Switching, page 168
- Information About L2VPN Pseudowire Switching, page 168
- How to Configure L2VPN Pseudowire Switching, page 169
- Configuration Examples for L2VPN Pseudowire Switching, page 172
- Additional References, page 177
- Feature Information for L2VPN Pseudowire Switching, page 178

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for L2VPN Pseudowire Switching

For the Cisco 12000 series routers, the L2VPN Pseudowire Switching feature for Any Transport over MPLS (AToM) is supported on the following engines:

- E2
- E3
- E4+
- E5
- E6

For engines that do not support this feature, the packets are sent to the software and forwarded through the slow path.

Note Engines E1 and E4 do not support L2VPN Pseudowire Switching, even in the slow path.

Restrictions for L2VPN Pseudowire Switching

- L2VPN Pseudowire Switching is supported with AToM.
- Only static, on-box provisioning is supported.
- Sequencing numbers in AToM packets are not processed by L2VPN Pseudowire Switching. The feature blindly passes the sequencing data through the xconnect packet paths, a process that is called transparent sequencing. The endpoint provider-edge (PE) to customer-edge (CE) connections enforce the sequencing.
- You can ping the adjacent next-hop PE router. End-to-end label switched path (LSP) pings are not supported.
- Do not configure IP or Ethernet interworking on a router where L2VPN Pseudowire Switching is enabled. Instead, configure interworking on the routers at the edge PEs of the network.
- The control word negotiation results must match. If either segment does not negotiate the control word, the control word is disabled for both segments.
- AToM Graceful Restart is negotiated independently on each pseudowire segment. If there is a transient loss of the label distribution protocol (LDP) session between two AToM PE routers, packets continue to flow.
- Per-pseudowire quality of service (QoS) is not supported. Traffic engineering (TE) tunnel selection is supported.
- Attachment circuit interworking is not supported.

Information About L2VPN Pseudowire Switching

- How L2VPN Pseudowire Switching Works, page 168
- How Packets Are Manipulated at the L2VPN Pseudowire Switching Aggregation Point, page 169

How L2VPN Pseudowire Switching Works

L2VPN Pseudowire Switching allows the user to extend L2VPN pseudowires across two separate MPLS networks or across an inter-AS boundary, as shown in the two figures below.

L2VPN Pseudowire Switching connects two or more contiguous pseudowire segments to form an end-to-end multihop pseudowire. This end-to-end pseudowire functions as a single point-to-point pseudowire.

As shown in the second figure below, L2VPN Pseudowire Switching enables you to keep the IP addresses of the edge PE routers private across inter-AS boundaries. You can use the IP address of the Autonomous System Boundary Routers (ASBRs) and treat them as pseudowire aggregation (PE-agg) routers. The ASBRs join the pseudowires of the two domains.

L2VPN Pseudowire Switching also enables you to keep different administrative or provisioning domains to manage the end-to-end service. At the boundaries of these networks, PE-agg routers delineate the management responsibilities.

Figure 5 L2VPN Pseudowire Switching in an Intra-AS Topology

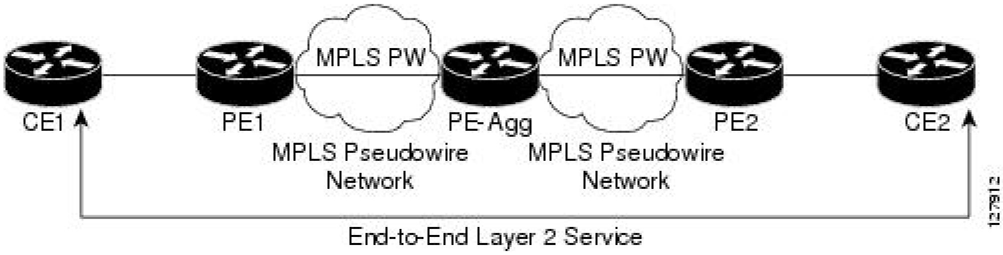
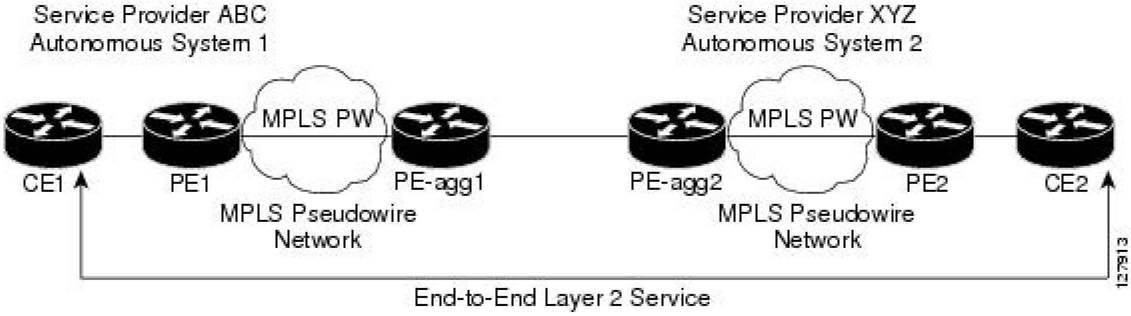


Figure 6 L2VPN Pseudowire Switching in an Inter-AS Topology


Service Provider ABC Autonomous System 1
Service Provider XYZ Autonomous System 2
MPLS PW
MPLS PW
CE1
PE1
PE-agg1
PE-agg2
PE2
CE2
MPLS Pseudowire Network
MPLS Pseudowire Network
End-to-End Layer 2 Service
127913

How Packets Are Manipulated at the L2VPN Pseudowire Switching Aggregation Point

Switching AToM packets between two AToM pseudowires is the same as switching any MPLS packet. The MPLS switching data path switches AToM packets between two AToM pseudowires. The following list explains exceptions:

- The outgoing virtual circuit (VC) label replaces the incoming VC label in the packet. New Internal Gateway Protocol (IGP) labels and Layer 2 encapsulation are added.
- The incoming VC label time-to-live (TTL) field is decremented by one and copied to the outgoing VC label TTL field.
- The incoming VC label EXP value is copied to the outgoing VC label EXP field.
- The outgoing VC label “Bottom of Stack” S bit in the outgoing VC label is set to 1.
- AToM control word processing is not performed at the L2VPN Pseudowire Switching aggregation point. Sequence numbers are not validated. Use the Router Alert label for LSP Ping; do not require control word inspection to determine an LSP Ping packet.

How to Configure L2VPN Pseudowire Switching

Use the following procedure to configure L2VPN Pseudowire Switching on each of the PE-agg routers. In this configuration, you are limited to two **neighbor**commands after entering the **l2 vfi**command.

- This procedure assumes that you have configured basic AToM L2VPNs. This procedure does not explain how to configure basic AToM L2VPNs that transport Layer 2 packets over an MPLS backbone. For information on the basic configuration, see Any Transport over MPLS .
- For interautonomous configurations, ASBRs require a labeled interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **l2 vfi** *name* **point-to-point**
4. **neighbor** *ip-address vcid* [**encapsulation mpls** | **pw-class** *pw-class-name]*
5. **exit**
6. **exit**
7. **show mpls l2transport vc** [**vcid** [*vc-id* | *vc-id-min vc-id-max*]] [**interface** *name*[*local-circuit-id*]] [**destination** *ip-address* | *name*] [**detail**]
8. **show vfi** [*vfi-name*]
9. **ping** [*protocol*] [**tag**] {*host-name*| *system-address*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**l2 vfi** *name* **point-to-point** **Example:** `Router(config)# l2 vfi atomtunnel point-to-point`	Creates a point-to-point Layer 2 virtual forwarding interface (VFI) and enters VFI configuration mode.
Step 4	**neighbor** *ip-address vcid* [**encapsulation mpls** | **pw-class** *pw-class-name]* **Example:** `Router(config-vfi)# neighbor 10.0.0.1 100 pw-class mpls`	Configures an emulated VC. • Specify the IP address and the VC ID of the remote router. • Also specify the pseudowire class to use for the emulated VC. **Note** Only two **neighbor**commands are allowed for each **l2 vfi point-to-point** command.

	Command or Action	Purpose
Step 5	**exit** **Example:** `Router(config-vfi)# exit`	Exits VFI configuration mode.
Step 6	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode.
Step 7	**show mpls l2transport vc** [**vcid** [*vc-id* | *vc-id-min vc-id-max*]] [**interface** *name*[*local-circuit-id*]] [**destination** *ip-address* | *name*] [**detail**] **Example:** `Router# show mpls l2transport vc`	Verifies that the L2VPN Pseudowire Switching session has been established.
Step 8	**show vfi** [*vfi-name*] **Example:** Router# **show vfi atomtunnel**	Verifies that a point-to-point VFI has been established.

Step 9	**ping** [*protocol*] [**tag**] {*host-name*| *system-address*}
	Example:
	`Router# ping 10.1.1.1`
		When issued from the CE routers, verifies end-to-end connectivity.

- Examples, page 171

Examples

The following example displays output from the **show mpls l2transport vc** command:

```
Router# show mpls l2transport vc
Local intf     Local circuit              Dest address    VC ID Status
-------------  -------------------------- --------------- ----- ----
MPLS PW        10.0.1.1:100               10.0.1.1        100   UP
MPLS PW        10.0.1.1:100               10.0.1.1        100   UP
```

The following example displays output from the **show vfi**command:

```
Router# show vfi
VFI name: test, type: point-to-point
 Neighbors connected via pseudowires:
```

```
Router ID         Pseudowire ID
10.0.1.1          100
10.0.1.1          100
```

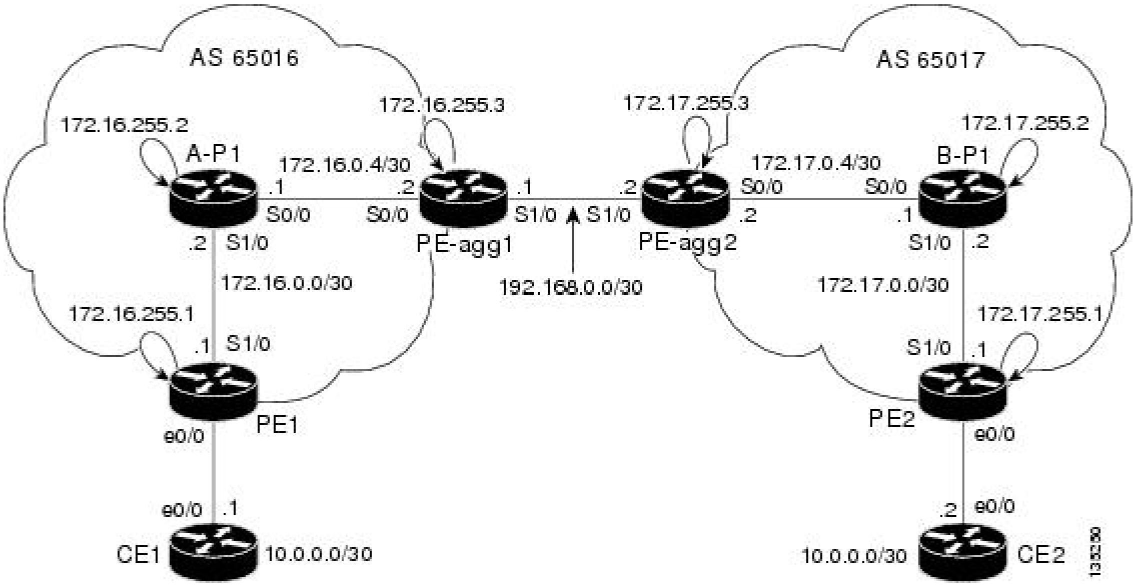
Configuration Examples for L2VPN Pseudowire Switching

- L2VPN Pseudowire Switching in an Inter-AS Configuration Example, page 172

L2VPN Pseudowire Switching in an Inter-AS Configuration Example

Two separate autonomous systems are able to pass L2VPN packets, because the two PE-agg routers have been configured with L2VPN Pseudowire Switching. This example configuration is shown in the figure below.

Figure 7 L2VPN Pseudowire Switching in an Interautonomous System



PE-agg-1	PE-agg-2
version 12.0	version 12.0
service timestamps debug uptime	service timestamps debug uptime
service timestamps log uptime	service timestamps log uptime
service password-encryption	service password-encryption
!	!
hostname [pe-agg1]	hostname [pe-agg2]
!	!
boot-start-marker	boot-start-marker
boot-end-marker	boot-end-marker
!	!
enable secret 5 $1$Q0Bb$32sIU82pHRgyddWaeB4zs/	enable secret 5 $1$32jd$zQRfxXzjstr4llV9DcWf7/
!	!
ip subnet-zero	ip subnet-zero
ip cef	ip cef
no ip domain-lookup	no ip domain-lookup
mpls label protocol ldp	mpls label protocol ldp
pseudowire-class SW-PW	pseudowire-class SW-PW
encapsulation mpls	encapsulation mpls
!	!
l2 vfi PW-SWITCH-1 point-to-point	l2 vfi PW-SWITCH-1 point-to-point
neighbor 172.17.255.3 100 pw-class SW-PW	neighbor 172.16.255.3 100 pw-class SW-PW
neighbor 172.16.255.1 16 pw-class SW-PW	neighbor 172.17.255.1 17 pw-class SW-PW
!	!
interface Loopback0	interface Loopback0
ip address 172.16.255.3 255.255.255.255	ip address 172.17.255.3 255.255.255.255
no ip directed-broadcast	no ip directed-broadcast
!	!
interface Serial0/0	interface Serial0/0
ip address 172.16.0.6 255.255.255.252	ip address 172.17.0.6 255.255.255.252
no ip directed-broadcast	no ip directed-broadcast
mpls ip	mpls ip

A-P1	B-P1
version 12.0	version 12.0
service timestamps debug uptime	service timestamps debug uptime
service timestamps log uptime	service timestamps log uptime
service password-encryption	service password-encryption
!	!
hostname [a-p1]	hostname [b-p1]
!	!
boot-start-marker	boot-start-marker
boot-end-marker	boot-end-marker
!	!
enable secret 5 $1$eiUn$rTMnZiYnJxtMTpO0NKpQQ/	enable secret 5 $1$svU/$2JmJZ/5gxlW4nVXVniIJe1
!	!
ip subnet-zero	ip subnet-zero
ip cef	ip cef
no ip domain-lookup	no ip domain-lookup
mpls label protocol ldp	mpls label protocol ldp
!	!
interface Loopback0	interface Loopback0
ip address 172.16.255.2 255.255.255.255	ip address 172.17.255.2 255.255.255.255
no ip directed-broadcast	no ip directed-broadcast
!	!
interface Serial0/0	interface Serial0/0
ip address 172.16.0.5 255.255.255.252	ip address 172.17.0.5 255.255.255.252
no ip directed-broadcast	no ip directed-broadcast
mpls ip	mpls ip
!	!
interface Serial1/0	interface Serial1/0
ip address 172.16.0.2 255.255.255.252	ip address 172.17.0.2 255.255.255.252
no ip directed-broadcast	no ip directed-broadcast
mpls ip	mpls ip
!	!

PE1	PE2
version 12.0	version 12.0
service timestamps debug uptime	service timestamps debug uptime
service timestamps log uptime	service timestamps log uptime
service password-encryption	service password-encryption
!	!
hostname [pe1]	hostname [pe2]
!	!
boot-start-marker	boot-start-marker
boot-end-marker	boot-end-marker
!	!
enable secret 5 $1$9z8F$2A1/YLc6NB6d.WLQXF0Bz1	enable secret 5 $1$rT.V$8Z6Dy/r8/eaRdx2TR/O5r/
!	!
ip subnet-zero	ip subnet-zero
ip cef	ip cef
no ip domain-lookup	no ip domain-lookup
mpls label protocol ldp	mpls label protocol ldp
pseudowire-class ETH-PW	pseudowire-class ETH-PW
encapsulation mpls	encapsulation mpls
!	!
interface Loopback0	interface Loopback0
ip address 172.16.255.1 255.255.255.255	ip address 172.17.255.1 255.255.255.255
no ip directed-broadcast	no ip directed-broadcast
!	!
interface Ethernet0/0	interface Ethernet0/0
no ip address	no ip address
no ip directed-broadcast	no ip directed-broadcast
no cdp enable	no cdp enable
xconnect 172.16.255.3 16 pw-class ETH-PW	xconnect 172.17.255.3 17 pw-class ETH-PW
!	!
interface Serial1/0	interface Serial1/0
ip address 172.16.0.1 255.255.255.252	ip address 172.17.0.1 255.255.255.252

CE1	CE2
version 12.0	version 12.0
service timestamps debug uptime	service timestamps debug uptime
service timestamps log uptime	service timestamps log uptime
service password-encryption	service password-encryption
!	!
hostname [ce1]	hostname [ce2]
!	!
boot-start-marker	boot-start-marker
boot-end-marker	boot-end-marker
!	!
enable secret 5 $1$o9N6$LSrxHufTn0vjCY0nW8hQX.	enable secret 5 $1$YHo6$LQ4z5PdrF5B9dnL75Xvvm1
!	!
ip subnet-zero	ip subnet-zero
ip cef	ip cef
no ip domain-lookup	no ip domain-lookup
!	!
interface Ethernet0/0	interface Ethernet0/0
ip address 10.0.0.1 255.255.255.252	ip address 10.0.0.2 255.255.255.252
no ip directed-broadcast	no ip directed-broadcast
!	!
ip classless	ip classless
!	!
control-plane	control-plane
!	!
line con 0	line con 0
exec-timeout 0 0	exec-timeout 0 0
line aux 0	line aux 0
line vty 0 4	line vty 0 4
login	login
!	!
no cns aaa enable	no cns aaa enable

Additional References

Related Documents

Related Topic	Document Title
Any Transport over MPLS	Any Transport over MPLS
Pseudowire redundancy	http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120limit/120s/120s31/fsstitch.htm *L2VPN Pseudowire Redundancy*
High availability for AToM	AToM Graceful Restart
L2VPN interworking	L2VPN Interworking
Layer 2 local switching	Layer 2 Local Switching
PWE3 MIB	Pseudowire Emulation Edge-to-Edge MIBs for Ethernet and Frame Relay Services
Packet sequencing	Any Transport over MPLS (AToM) Sequencing Support

Standards

Standard	Title
draft-ietf-pwe3-control-protocol-14.txt	*Pseudowire Setup and Maintenance using LDP*
draft-martini-pwe3-pw-switching-01.txt	*Pseudo Wire Switching*

MIBs

MIB	MIBs Link
• CISCO-IETF-PW-MIB • CISCO-IETF-PW-MPLS-MIB • CISCO-IETF-PW-ENET-MIB • CISCO-IETF-PW-FR-MIB	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
None	—

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for L2VPN Pseudowire Switching

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 18 ***Feature Information for L2VPN Pseudowire Switching***

Feature Name	Releases	Feature Information
L2VPN Pseudowire Switching	12.0(31)S, 12.2(28)SB, 12.2(33)SRB, 12.2(33)SRD2, 12.2(33)SRE	This feature configures L2VPN Pseudowire Switching, which extends L2VPN pseudowires across an interautonomous system (inter-AS) boundary or across two separate MPLS networks. In Cisco IOS Release 12.2(28)SB, support was added for the Cisco 7200 and 7301 series routers. In 12.2(33)SRD2, support was added for ATM and TDM ACs. The following commands were introduced or modified: **l2 vfi point-to-point**, **neighbor**(L2VPN Pseudowire Switching), **show vfi**.

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.

L2VPN Pseudowire Redundancy

The L2VPN Pseudowire Redundancy feature lets you configure your network to detect a failure in the network and reroute the Layer 2 (L2) service to another endpoint that can continue to provide service. This feature provides the ability to recover from a failure either of the remote provider edge (PE) router or of the link between the PE and customer edge (CE) routers.

- Finding Feature Information, page 181
- Prerequisites for L2VPN Pseudowire Redundancy, page 181
- Restrictions for L2VPN Pseudowire Redundancy, page 182
- Information About L2VPN Pseudowire Redundancy, page 182
- How to Configure L2VPN Pseudowire Redundancy, page 184
- Configuration Examples for L2VPN Pseudowire Redundancy, page 189
- Additional References, page 191
- Feature Information for L2VPN Pseudowire Redundancy, page 192

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for L2VPN Pseudowire Redundancy

- This feature module requires that you understand how to configure basic L2 virtual private networks (VPNs). You can find that information in the following documents:
    - *Any Transport over MPLS*
    - *L2 VPN Interworking*
- The L2VPN Pseudowire Redundancy feature requires that the following mechanisms be in place to enable you to detect a failure in the network:
    - Label-switched paths (LSP) Ping/Traceroute and Any Transport over MPLS Virtual Circuit Connection Verification (AToM VCCV)
    - Local Management Interface (LMI)

- ◦ Operation, Administration, and Maintenance (OAM)

Restrictions for L2VPN Pseudowire Redundancy

General Restrictions

- The primary and backup pseudowires must run the same type of transport service. The primary and backup pseudowires must be configured with AToM.
- Only static, on-box provisioning is supported.
- If you use L2VPN Pseudowire Redundancy with L2VPN Interworking, the interworking method must be the same for the primary and backup pseudowires.
- Setting the experimental (EXP) bit on the Multiprotocol Label Switching (MPLS) pseudowire is supported.
- Different pseudowire encapsulation types on the MPLS pseudowire are not supported.
- The **mpls l2transport route** command is not supported. Use the **xconnect** command instead.
- The ability to have the backup pseudowire fully operational at the same time that the primary pseudowire is operational is not supported. The backup pseudowire becomes active only after the primary pseudowire fails.
- The AToM VCCV feature is supported only on the active pseudowire.
- More than one backup pseudowire is not supported.

Restrictions for Layer 2 Tunnel Protocol Version 3 (L2TPv3) Xconnect Configurations

- Interworking is not supported.
- Local switching backup by pseudowire redundancy is not supported.
- PPP, HDLC, and Frame-Relay attachment circuit (AC) types of L2TPv3 pseudowire redundancy are not supported.
- For the edge interface, only the Cisco 7600 series SPA Interface Processor-400 (SIP-400) linecard with the following shared port adapters (SPAs) is supported:

Cisco 2-Port Gigabit Ethernet Shared Port Adapter (SPA-2X1GE) Cisco 2-Port Gigabit Ethernet Shared Port Adapter, Version 2 (SPA-2X1GE-V2) Cisco 5-Port Gigabit Ethernet Shared Port Adapter, Version 2 (SPA-5X1GE-V2) Cisco 10-Port Gigabit Ethernet Shared Port Adapter, Version 2 (SPA-10X1GE-V2) Cisco 2-Port OC3c/STM1c ATM Shared Port Adapter (SPA-2XOC3-ATM) Cisco 4-Port OC3c/STM1c ATM Shared Port Adapter (SPA-4XOC3-ATM) Cisco 1-Port OC12c/STM4c ATM Shared Port Adapter (SPA-1XOC12-ATM) Cisco 1-Port OC-48c/STM-16 ATM Shared Port Adapter (SPA-1XOC48-ATM)

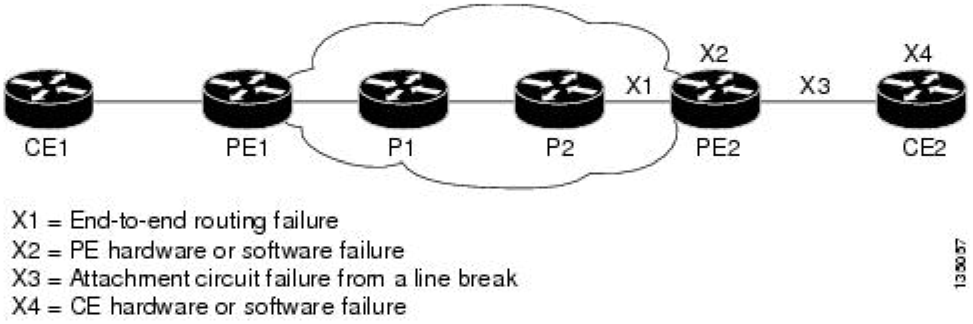
Information About L2VPN Pseudowire Redundancy

- Introduction to L2VPN Pseudowire Redundancy, page 182

Introduction to L2VPN Pseudowire Redundancy

L2VPNs can provide pseudowire resiliency through their routing protocols. When connectivity between end-to-end PE routers fails, an alternative path to the directed LDP session and the user data can take over. However, there are some parts of the network where this rerouting mechanism does not protect against interruptions in service. The figure below shows those parts of the network that are vulnerable to an interruption in service.

Figure 8 Points of Potential Failure in an L2VPN Network

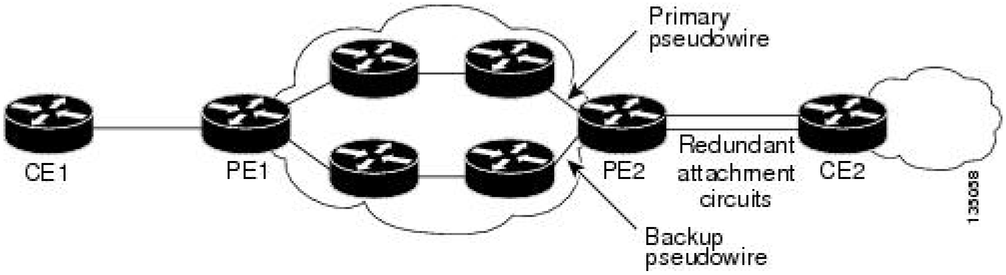

X2
X4
X1
X3
CE1
PE1
P1
P2
PE2
CE2
X1 = End-to-end routing failure
X2 = PE hardware or software failure
X3 = Attachment circuit failure from a line break
X4 = CE hardware or software failure
135057

The L2VPN Pseudowire Redundancy feature provides the ability to ensure that the CE2 router in the figure above can always maintain network connectivity, even if one or all the failures in the figure occur.

The L2VPN Pseudowire Redundancy feature enables you to set up backup pseudowires. You can configure the network with redundant pseudowires (PWs) and redundant network elements, which are shown in the three figures below.

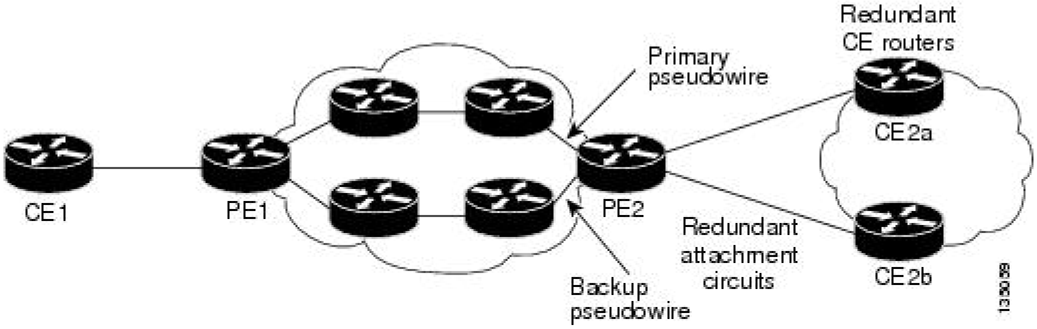
The figure below shows a network with redundant pseudowires and redundant attachment circuits.

Figure 9 L2VPN Network with Redundant PWs and Attachment Circuits



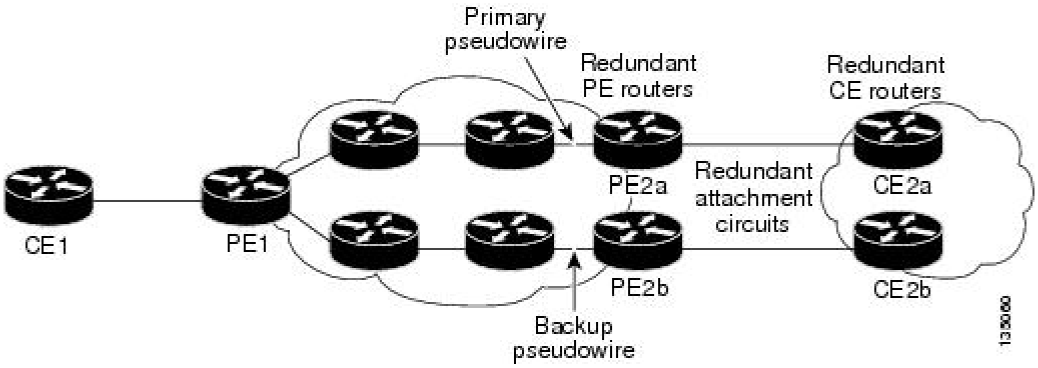
The figure below shows a network with redundant pseudowires, attachment circuits, and CE routers.

Figure 10 L2VPN Network with Redundant PWs, Attachment Circuits, and CE Routers



The figure below shows a network with redundant pseudowires, attachment circuits, CE routers, and PE routers.

Figure 11 L2VPN Network with Redundant PWs, Attachment Circuits, CE Routers, and PE Routers



How to Configure L2VPN Pseudowire Redundancy

The L2VPN Pseudowire Redundancy feature enables you to configure a backup pseudowire in case the primary pseudowire fails. When the primary pseudowire fails, the PE router can switch to the backup pseudowire. You can have the primary pseudowire resume operation after it comes back up.

The default Label Distribution Protocol (LDP) session hold-down timer will enable the software to detect failures in about 180 seconds. That time can be configured so that the software can detect failures more quickly. See the **mpls ldp holdtime** command for more information.

- Configuring the Pseudowire, page 184
- Configuring L2VPN Pseudowire Redundancy, page 185
- Forcing a Manual Switchover to the Backup Pseudowire VC, page 187
- Verifying the L2VPN Pseudowire Redundancy Configuration, page 188

Configuring the Pseudowire

The successful transmission of the Layer 2 frames between PE routers is due to the configuration of the PE routers. You set up the connection, called a pseudowire, between the routers.

The pseudowire-class configuration group specifies the characteristics of the tunneling mechanism, which are:

- Encapsulation type
- Control protocol
- Payload-specific options

You must specify the **encapsulation mpls**command as part of the pseudowire class for the AToM VCs to work properly. If you omit the **encapsulation mpls**command as part of the **xconnect**command, you receive the following error:

```
% Incomplete command.
```

Perform this task to configure a pseudowire class.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **pseudowire-class** name
4. **encapsulation mpls**
5. **interworking** {**ethernet** | **ip**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**pseudowire-class** name **Example:** `Router(config)# pseudowire-class atom`	Establishes a pseudowire class with a name that you specify. Enters pseudowire class configuration mode.
Step 4	**encapsulation mpls** **Example:** `Router(config-pw-class)# encapsulation mpls`	Specifies the tunneling encapsulation. For AToM, the encapsulation type is **mpls**.

Step 5	**interworking** {**ethernet** | **ip**}
	Example:
	`Router(config-pw-class)# interworking ip`
		(Optional) Enables the translation between the different Layer 2 encapsulations.

Configuring L2VPN Pseudowire Redundancy

Use the following steps to configure the L2VPN Pseudowire Redundancy feature.

For each transport type, the **xconnect**command is configured slightly differently. The following configuration steps use Ethernet VLAN over MPLS, which is configured in subinterface configuration mode. See *Any Transport over MPLS* to determine how to configure the **xconnect** command for other transport types.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface gigabitethernet** *slot* **/** *subslot* **/** *interface* **.** *subinterface*
4. **encapsulation dot1q** vlan-id
5. **xconnect** *peer-router-id vcid* {**encapsulation mpls**| **pw-class** *pw-class-name}*
6. **backup peer** *peer-router-ip-addr vcid* [**pw-class** *pw-class-name*]
7. **backup delay** *e nable-delay* {*disable-delay* | **never**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface gigabitethernet** *slot* **/** *subslot* **/** *interface* **.** *subinterface* **Example:** `Router(config)# interface gigabitethernet0/0/0.1`	Specifies the Gigabit Ethernet subinterface and enters subinterface configuration mode. Make sure that the subinterface on the adjoining CE router is on the same VLAN as this PE router.
Step 4	**encapsulation dot1q** vlan-id **Example:** `Router(config-subif)# encapsulation dot1q 100`	Enables the subinterface to accept 802.1Q VLAN packets. The subinterfaces between the CE and PE routers that are running Ethernet over MPLS must be in the same subnet. All other subinterfaces and backbone routers do not.

	Command or Action	Purpose

Step 5	**xconnect** *peer-router-id vcid* {**encapsulation mpls**| **pw-class** *pw-class-name}*
	Example:
	`Router(config-subif)# xconnect 10.0.0.1 123 pw-class atom`
		Binds the attachment circuit to a pseudowire VC.
		The syntax for this command is the same as for all other Layer 2 transports.
		Enters xconnect configuration mode.

Step 6	**backup peer** *peer-router-ip-addr vcid* [**pw-class** *pw-class-name*] **Example:** `Router(config-if-xconn)# backup peer 10.0.0.3 125 pw-class atom`	Specifies a redundant peer for the pseudowire VC. The pseudowire class name must match the name you specified when you created the pseudowire class, but you can use a different pw-class in the **backup peer** command than the name that you used in the primary **xconnect** command.

Step 7	**backup delay** *e nable-delay* {*disable-delay* | **never**}
	Example:
	`Router(config-if-xconn)# backup delay 5 never`
		Specifies how long (in seconds) the backup pseudowire VC should wait to take over after the primary pseudowire VC goes down. The range is 0 to 180.
		Specifies how long the primary pseudowire should wait after it becomes active to take over for the backup pseudowire VC. The range is 0 to 180 seconds. If you specify the **never keyword**, the primary pseudowire VC never takes over for the backup.

Forcing a Manual Switchover to the Backup Pseudowire VC

To force the router switch over to the backup or primary pseudowire, you can enter the **xconnect backup force switchover** command in privileged EXEC mode. You can specify either the interface of the primary attachment circuit (AC) to switch to or the IP-address and VC ID of the peer router.

A manual switchover can be made only if the interface or peer specified in the command is actually available and the xconnect will move to the fully active state when the command is entered.

SUMMARY STEPS

1. **enable**
2. **xconnect backup force-switchover { interface** *interface-info* | **peer** *ip-address vcid}*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose

Step 2	**xconnect backup force-switchover { interface** *interface-info* | **peer** *ip-address vcid}*
	Example:
	`Router# xconnect backup force-switchover peer 10.10.10.1 123`
		Specifies that the router should switch to the backup or to the primary pseudowire.

Verifying the L2VPN Pseudowire Redundancy Configuration

Use the following commands to verify that the L2VPN Pseudowire Redundancy feature is correctly configured.

SUMMARY STEPS

1. **show mpls l2transport vc**
2. **show xconnect all**
3. **xconnect logging redundancy**

DETAILED STEPS

Step 1 **show mpls l2transport vc**

In this example, the primary attachment circuit is up. The backup attachment circuit is available, but not currently selected. The **show** output displays as follows:

Example:

```
Router# show mpls l2transport vc
Local intf     Local circuit           Dest address    VC ID      Status
-------------  ----------------------- --------------- ---------- ----------
Et0/0.1        Eth VLAN 101            10.0.0.2        101        UP
Et0/0.1        Eth VLAN 101            10.0.0.3        201        DOWN
Router# show mpls l2transport vc detail
Local interface: Et0/0.1 up, line protocol up, Eth VLAN 101 up
   Destination address 10.0.0.2 VC ID: 101, VC status UP
   .
   .
   .
Local interface: Et0/0.1 down, line protocol down, Eth VLAN 101 down
   Destination address 10.0.0.3 VC ID: 201, VC status down
   .
   .
   .
```

Step 2 **show xconnect all**

In this example, the topology is Attachment Circuit 1 to Pseudowire 1 with a Pseudowire 2 as a backup:

Example:

```
Router# show xconnect all
Legend: XC ST=Xconnect State, S1=Segment1 State, S2=Segment2 State
```

```
UP=Up, DN=Down, AD=Admin Down, IA=Inactive, NH=No Hardware
XC ST  Segment 1                         S1 Segment 2                         S2
------+---------------------------------+--+---------------------------------+--
UP pri ac   Et0/0(Ethernet)              UP mpls 10.55.55.2:1000               UP
IA sec ac   Et0/0(Ethernet)              UP mpls 10.55.55.3:1001               DN
```

In this example, the topology is Attachment Circuit 1 to Attachment Circuit 2 with a Pseudowire backup for Attachment Circuit 2:

Example:

```
Router# show xconnect all
Legend: XC ST=Xconnect State, S1=Segment1 State, S2=Segment2 State
UP=Up, DN=Down, AD=Admin Down, IA=Inactive, NH=No Hardware
XC ST  Segment 1                         S1 Segment 2                         S2
------+---------------------------------+--+---------------------------------+--
UP pri ac   Se6/0:150(FR DLCI)           UP ac   Se8/0:150(FR DLCI)            UP
IA sec ac   Se6/0:150(FR DLCI)           UP mpls 10.55.55.3:7151               DN
```

Step 3 **xconnect logging redundancy**

In addition to the **show mpls l2transport vc**command and the **show xconnect** command, you can use the **xconnect logging redundancy** command to track the status of the xconnect redundancy group:

Example:

```
Router(config)# xconnect logging redundancy
```

When this command is configured, the following messages will be generated during switchover events:

Activating the primary member:

Example:

```
00:01:07: %XCONNECT-5-REDUNDANCY: Activating primary member 10.55.55.2:1000
```

Activating the backup member:

Example:

```
00:01:05: %XCONNECT-5-REDUNDANCY: Activating secondary member 10.55.55.3:1001
```

Configuration Examples for L2VPN Pseudowire Redundancy

Each of the configuration examples refers to one of the following pseudowire classes:

- AToM (like-to-like) pseudowire class:

```
pseudowire-class mpls
 encapsulation mpls
```

- L2VPN IP interworking:

```
pseudowire-class mpls-ip
```

```
 encapsulation mpls
 interworking ip
```

- L2VPN Pseudowire Redundancy and AToM Like to Like Examples, page 190
- L2VPN Pseudowire Redundancy and L2VPN Interworking Examples, page 190
- L2VPN Pseudowire Redundancy with Layer 2 Local Switching Examples, page 191

L2VPN Pseudowire Redundancy and AToM Like to Like Examples

The following example shows a High-Level Data Link Control (HDLC) attachment circuit xconnect with a backup pseudowire:

```
interface Serial4/0
 xconnect 10.55.55.2 4000 pw-class mpls
 backup peer 10.55.55.3 4001 pw-class mpls
```

The following example shows a Frame Relay attachment circuit xconnect with a backup pseudowire:

```
connect fr-fr-pw Serial6/0 225 l2transport
 xconnect 10.55.55.2 5225 pw-class mpls
 backup peer 10.55.55.3 5226 pw-class mpls
```

L2VPN Pseudowire Redundancy and L2VPN Interworking Examples

The following example shows an Ethernet attachment circuit xconnect with L2VPN IP interworking and a backup pseudowire:

```
interface Ethernet0/0
 xconnect 10.55.55.2 1000 pw-class mpls-ip
 backup peer 10.55.55.3 1001 pw-class mpls-ip
```

The following example shows an Ethernet VLAN attachment circuit xconnect with L2VPN IP interworking and a backup pseudowire:

```
interface Ethernet1/0.1
 encapsulation dot1Q 200
 no ip directed-broadcast
 xconnect 10.55.55.2 5200 pw-class mpls-ip
 backup peer 10.55.55.3 5201 pw-class mpls-ip
```

The following example shows a Frame Relay attachment circuit xconnect with L2VPN IP interworking and a backup pseudowire:

```
connect fr-ppp-pw Serial6/0 250 l2transport
 xconnect 10.55.55.2 8250 pw-class mpls-ip
 backup peer 10.55.55.3 8251 pw-class mpls-ip
```

The following example shows a PPP attachment circuit xconnect with L2VPN IP interworking and a backup pseudowire:

```
interface Serial7/0
 encapsulation ppp
 xconnect 10.55.55.2 2175 pw-class mpls-ip
 backup peer 10.55.55.3 2176 pw-class mpls-ip
```

L2VPN Pseudowire Redundancy with Layer 2 Local Switching Examples

The following example shows an Ethernet VLAN-VLAN local switching xconnect with a pseudowire backup for Ethernet segment E2/0.2. If the subinterface associated with E2/0.2 goes down, the backup pseudowire is activated.

```
connect vlan-vlan Ethernet1/0.2 Ethernet2/0.2
 backup peer 10.55.55.3 1101 pw-class mpls
```

The following example shows a Frame Relay-to-Frame Relay local switching connect with a pseudowire backup for Frame Relay segment S8/0 150. If data-link connection identifier (DLCI) 150 on S8/0 goes down, the backup pseudowire is activated.

```
connect fr-fr-ls Serial6/0 150 Serial8/0 150
 backup peer 10.55.55.3 7151 pw-class mpls
```

Additional References

Related Documents

Related Topic	Document Title
Any Transport over MPLS	Any Transport over MPLS
High Availability for AToM	AToM Graceful Restart
L2VPN Interworking	L2VPN Interworking
Layer 2 local switching	Layer 2 Local Switching
PWE3 MIB	Pseudowire Emulation Edge-to-Edge MIBs for Ethernet and Frame Relay Services
Packet sequencing	Any Transport over MPLS (AToM) Sequencing Support

Standards

Standards	Title
None	--

MIBs

MIBs	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
None	--

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

Feature Information for L2VPN Pseudowire Redundancy

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 19 Feature Information for L2VPN Pseudowire Redundancy

Feature Name	Releases	Feature Information
L2VPN Pseudowire Redundancy	12.0(31)S 12.2(28)SB 12.4(11)T 12.2(33)SRB 12.2(22)SXI 15.0(1)S	This feature enables you to set up your network to detect a failure in the network and reroute the Layer 2 service to another endpoint that can continue to provide service. In Cisco IOS Release 12.0(31)S, the L2VPN Pseudowire Redundancy feature was introduced for Any Transport over MPLS (AToM) on the Cisco 12000 series routers. This feature was integrated into Cisco IOS Release 12.2(28)SB. This feature was integrated into Cisco IOS Release 12.4(11)T. This feature was integrated into Cisco IOS Release 12.2(33)SRB. This feature was integrated into Cisco IOS Release 12.2(33)SXI. The following commands were introduced or modified: **backup delay (L2VPN local switching)**, **backup peer**, **show xconnect**, **xconnect backup force-switchover**, **xconnect logging redundancy**.
L2VPN Pseudowire Redundancy for L2TPv3	12.2(33)SRE 15.0(1)S	This feature provides L2VPN pseudowire redundancy for L2TPv3 xconnect configurations. In Cisco IOS Release 12.2(33)SRE, this feature was implemented on the Cisco 7600 series routers.
Resilient Pseudowire (RPW): PW Fast Recovery	15.2(1)S	This feature was integrated into Cisco IOS Release 15.2(1)S. The following commands were introduced or modified: **aps hspw-icrm-grp** , **show hspw-aps-icrm**.

VPLS Autodiscovery BGP Based

VPLS Autodiscovery enables each Virtual Private LAN Service (VPLS) provider edge (PE) router to discover which other PE routers are part of the same VPLS domain. VPLS Autodiscovery also automatically detects when PE routers are added to or removed from the VPLS domain. You no longer need to manually configure the VPLS and maintain the configuration when a PE router is added or deleted. VPLS Autodiscovery uses the Border Gateway Protocol (BGP) to discover the VPLS members and to set up and tear down pseudowires in the VPLS.

- Feature Information For, page 195
- Prerequisites for VPLS Autodiscovery BGP Based, page 195
- Restrictions for VPLS Autodiscovery BGP Based, page 195
- Information About VPLS Autodiscovery BGP Based, page 196
- How to Configure VPLS Autodiscovery BGP Based, page 199
- Configuration Examples for VPLS Autodiscovery BGP Based, page 206
- Additional References, page 208
- Feature Information for VPLS Autodiscovery BGP Based, page 210

Feature Information For

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for VPLS Autodiscovery BGP Based

Before configuring VPLS Autodiscovery, if you are using a Cisco 7600 series router, perform the Cisco 7600 router-specific tasks listed in the section called “Virtual Private LAN Services on the Optical Service Modules” in the Cisco 7600 Series Router IOS Software Configuration Guide.

Restrictions for VPLS Autodiscovery BGP Based

- VPLS Autodiscovery supports only IPV4 addresses.

- VPLS Autodiscovery uses Forwarding Equivalence Class (FEC) 129 to convey endpoint information. Manually configured pseudowires use FEC 128.
- VPLS Autodiscovery is not supported with Layer 2 Tunnel Protocol Version 3 (L2TPv3).
- VPLS Autodisocovery is not supported with interautonomous system configurations.
- You can configure both autodiscovered and manually configured pseudowires in a single virtual forwarding instance (VFI). However, the pseudowires cannot go to the same peer PE router.
- If you manually configure a neighbor using the **neighbor (VPLS)**command after you have enabled VPLS Autodiscovery and both peers are in autodiscovery mode, manually configure the route target (RT) values to prevent each peer from receiving discovery data for that VPLS.
- If you manually configure multiple pseudowires and target different IP addresses on the same PE router for each pseudowire, do not use the same virtual circuit identifier (VC ID) to identify the pseudowires terminated at the same PE router.
- You cannot configure a pseudowire by manually configuring a neighbor on one PE router and using autodiscovery on the other PE router to configure the same pseudowire in the other direction.
- Tunnel selection is not supported with autodiscovered neighbors.
- You can have up to 16 route targets only per VFI.
- The same RT is not allowed in multiple VFIs in the same PE router.
- The BGP autodiscovery process does not support dynamic hierarchical VPLS. User-facing PE (U-PE) routers cannot discover the network-facing PE (N-PE) routers, and N-PE routers cannot discover U-PE routers.
- Pseudowires for autodiscovered neighbors are provisioned with split horizon enabled. Therefore, manually configure the pseudowires for hierarchical VPLS. Make sure the U-PE routers do not participate in BGP autodiscovery for those pseudowires.
- Do not disable split horizon on autodiscovered neighbors. Split horizon is required with VPLS Autodiscovery.
- The provisioned peer address must be a /32 address bound to the peer’s Label Distribution Protocol (LDP) router ID.
- The peer PE router must be able to access the IP address that is used as the local LDP router ID. Even though the IP address need not be used in the **xconnect** command on the peer PE router, that IP address must be reachable.
- VPLS Autodiscovery is supported on the Cisco 7600 router hardware. For details on supported shared port adapters and line cards, see the following documents:

    - Cisco 7600 Series Router Cisco IOS Software Configuration Guide
    - Release Notes for Cisco IOS Release 12.2SR for the Cisco 7600 Series Routers

Information About VPLS Autodiscovery BGP Based

- How the VPLS Feature Works, page 197
- How the VPLS Autodiscovery BGP Based Feature Works, page 197
- How Enabling VPLS Autodiscovery Differs from Manually Configuring VPLS, page 197
- Show Commands Affected by VPLS Autodiscovery BGP Based, page 198
- BGP VPLS Autodiscovery Support on a Route Reflector, page 198

How the VPLS Feature Works

VPLS allows Multiprotocol Label Switching (MPLS) networks to provide multipoint Ethernet LAN services, also known as Transparent LAN Services (TLS). All customer sites in a VPLS appear to be on the same LAN, even though those sites might be in different geographic locations.

How the VPLS Autodiscovery BGP Based Feature Works

VPLS Autodiscovery enables each VPLS PE router to discover the other PE routers that are part of the same VPLS domain. VPLS Autodiscovery also tracks when PE routers are added to or removed from the VPLS domain. The autodiscovery and signaling functions use BGP to find and track the PE routers.

BGP uses the L2VPN Routing Information Base (RIB) to store endpoint provisioning information, which is updated each time any Layer 2 VFI is configured. Prefix and path information is stored in the L2VPN database, allowing BGP to make decisions on the best path. When BGP distributes the endpoint provisioning information in an update message to all its BGP neighbors, the endpoint information is used to configure a pseudowire mesh to support L2VPN-based services.

The BGP autodiscovery mechanism facilitates the configuration of L2VPN services, which are an integral part of the Cisco IOS Virtual Private LAN Service (VPLS) feature. VPLS enables flexibility in deploying services by connecting geographically dispersed sites as a large LAN over high-speed Ethernet in a robust and scalable IP MPLS network. For more information about BGP and the L2VPN address family in relation to VPLS Autodiscovery, see the following documents:

- The section called “L2VPN Address Family” in the Cisco BGP Overview .
- The document called BGP Support for the L2VPN Address Family

How Enabling VPLS Autodiscovery Differs from Manually Configuring VPLS

With VPLS Autodiscovery, you no longer need to manually set up the VPLS. The commands you use to set up VPLS Autodiscovery are similar to those you use to manually configure a VPLS, as shown in the table below. VPLS Autodiscovery uses **neighbor** commands in L2VPN address family mode to distribute endpoint information to configure a pseudowire.

Table 20 VPLS Autodiscovery Configuration versus Manual VPLS Configuration

Manual Configuration of VPLS	VPLS Autodiscovery: BGP Based

`l2 vfi vpls1 manual`
` vpn id 100`
` neighbor 10.10.10.1 encapsulation mpls`
` neighbor 10.10.10.0 encapsulation mpls`
` exit`
	`l2 vfi vpls1 autodiscovery`
	` vpn id 100`
	` exit`
	`router bgp 1`
	` no bgp default ipv4-unicast`
	` bgp log-neighbor-changes`
	` bgp update-delay 1`
	` neighbor 10.1.1.2 remote-as 1`
	` neighbor 10.1.1.2 update-source`
	`Loopback1`
	`.`
	`.`
	`.`
	` address-family l2vpn vpls`
	` neighbor 10.1.1.2 activate`
	` neighbor 10.1.1.2 send-community`
	`extended`
	`exit-address-family`

When you configure VPLS Autodiscovery, you enter the **l2vfi autodiscovery** command. This command allows the VFI to learn and advertise the pseudowire endpoints. As a result, you no longer need to enter the **neighbor (VPLS)**command in L2 VFI configuration mode.

However, the **neighbor (VPLS)**command is still supported with VPLS Autodiscovery in L2 VFI command mode. You can use the **neighbor (VPLS)**command to allow PE routers that do not participate in the autodiscovery process to join the VPLS. You can also use the **neighbor (VPLS)**command with PE routers that have been configured using the Tunnel Selection feature. You can also use the **neighbor (VPLS)**command in hierarchical VPLS configurations that have U-PE routers that do not participate in the autodiscovery process and have split-horizon forwarding disabled.

Show Commands Affected by VPLS Autodiscovery BGP Based

VPLS Autodiscovery changes the following show commands:

- The **show mpls l2transport vc** command with the **detail** keyword has been updated to include FEC 129 signaling information for the autodiscovered VPLS pseudowires.
- The **show vfi** command now displays information related to autodiscovered VFIs. The new information includes the VPLS ID, the route distinguisher (RD), the RT, and the router IDs of the discovered peers.
- The **show xconnect** command has been updated with the **rib** keyword to provide RIB information about the pseudowires.

BGP VPLS Autodiscovery Support on a Route Reflector

VPLS Autodiscovery is normally run on PE routers to support endpoint discovery and the setup of pseudowires between the PEs (typically a full mesh). VPLS does not normally run on a BGP route reflector. In Cisco IOS Release 12.2(33)SRE, VPLS Autodiscovery support was added to route reflectors. The BGP route reflector can be used to reflect the BGP VPLS prefixes without having VPLS explicitly configured on the route reflector.

The route reflector does not participate in the autodiscovery, meaning that no pseudowires are set up between the route reflector and the PEs. The route reflector reflects the VPLS prefixes to other PEs, so that the PEs do not need to have a full mesh of BGP sessions. The network administrator configures only the

BGP VPLS address family on the route reflector. For an example configuration of VPLS autodiscovery support on a route reflector, see the BGP VPLS Autodiscovery Support on Route Reflector Example, page 208.

How to Configure VPLS Autodiscovery BGP Based

- Enabling VPLS Autodiscovery BGP Based, page 199
- Configuring BGP to Enable VPLS Autodiscovery, page 200
- Customizing the VPLS Autodiscovery Settings, page 204

Enabling VPLS Autodiscovery BGP Based

Perform the following task to enable each VPLS PE router to discover the other PE routers that are part of the same VPLS domain.

Before configuring VPLS Autodiscovery, perform the Cisco 7600 router-specific tasks listed in the “Virtual Private LAN Services on the Optical Services Modules” chapter in the Cisco 7600 Series Router Cisco IOS Software Configuration Guide , Release 12.2SR.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **l2 vfi** *vfi-name* **autodiscovery**
4. **vpn id** *vpn-id*
5. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**l2 vfi** *vfi-name* **autodiscovery** **Example:** `Router(config)# l2 vfi vpls1 autodiscovery`	Enables VPLS Autodiscovery on the PE router and enters L2 VFI configuration mode.
Step 4	**vpn id** *vpn-id* **Example:** `Router(config-vfi)# vpn id 10`	Configures a VPN ID for the VPLS domain.
Step 5	**exit** **Example:** `Router(config-vfi)# exit`	Exits L2 VFI configuration mode. Commands take effect after the router exits L2 VFI configuration mode.

Configuring BGP to Enable VPLS Autodiscovery

In Cisco IOS Release 12.2(33)SRB, the BGP L2VPN address family was introduced with a separate L2VPN RIB that contains endpoint provisioning information for VPLS Autodiscovery. BGP learns the endpoint provisioning information from the L2VPN database which is updated each time a Layer 2 virtual forwarding instance (VFI) is configured. When BGP distributes the endpoint provisioning information in an update message to all its BGP neighbors, the endpoint information is used to configure a pseudowire mesh to support aL2VPN-based services.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router bgp** *autonomous-system-number*
4. **no bgp default ipv4-unicast**
5. **bgp log-neighbor-changes**
6. **neighbor** {*ip-address*| *peer-group-name*} **remote-as** *autonomous-system-number*
7. **neighbor** {*ip-address*| *peer-group-name*} **update-source** *interface-type interface-number*
8. Repeat Step 6 and Step 7 to configure other BGP neighbors
9. **address-family l2vpn** [**vpls**]
10. **neighbor** {*ip-address* | *peer-group-name*} **activate**
11. **neighbor** {*ip-address*| *peer-group-name*} **send-community**{**both**| **standard**| **extended**}
12. Repeat Step 10 and Step 11 to activate other BGP neighbors under an L2VPN address family.
13. **exit-address-family**
14. **exit**
15. **exit**
16. **show vfi**
17. **show ip bgp l2vpn vpls** {**all** | **rd** *vpn-rd*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**router bgp** *autonomous-system-number* **Example:** `Router(config)# router bgp 65000`	Enters router configuration mode for the specified routing process.

	Command or Action	Purpose
Step 4	**no bgp default ipv4-unicast** **Example:** `Router(config-router)# no bgp default ipv4-unicast`	Disables the IPv4 unicast address family for the BGP routing process. **Note** Routing information for the IPv4 unicast address family is advertised by default for each BGP routing session configured with the **neighbor remote-as** router configuration command unless you configure the **no bgp default ipv4-unicast**router configuration command before configuring the **neighbor remote-as** command. Existing neighbor configurations are not affected.
Step 5	**bgp log-neighbor-changes** **Example:** `Router(config-router)# bgp log-neighbor-changes`	Enables logging of BGP neighbor resets.
Step 6	**neighbor** {*ip-address*| *peer-group-name*} **remote-as** *autonomous-system-number* **Example:** `Router(config-router)# neighbor 10.10.10.1 remote-as 65000`	Adds the IP address or peer group name of the neighbor in the specified autonomous system to the IPv4 multiprotocol BGP neighbor table of the local router. • If the *autonomous-system-number* argument matches the autonomous system number specified in the **router bgp** command, the neighbor is an internal neighbor. • If the *autonomous-system-number* argument does not match the autonomous system number specified in the **router bgp** command, the neighbor is an external neighbor. • In this example, the neighbor at 10.10.10.1 is an internal BGP neighbor.
Step 7	**neighbor** {*ip-address*| *peer-group-name*} **update-source** *interface-type interface-number* **Example:** `Router(config-router)# neighbor 10.10.10.1 update-source loopback1`	(Optional) Configures a router to select a specific source or interface to receive routing table updates. • This example uses a loopback interface. The advantage to this configuration is that the loopback interface is not affected by the effects of a flapping interface.
Step 8	Repeat Step 6 and Step 7 to configure other BGP neighbors	—
Step 9	**address-family l2vpn** [**vpls**] **Example:** `Router(config-router)# address-family l2vpn vpls`	Specifies the L2VPN address family and enters address family configuration mode. • The optional **vpls** keyword specifies that VPLS endpoint provisioning information is to be distributed to BGP peers. • In this example, an L2VPN VPLS address family session is created.

	Command or Action	Purpose
Step 10	**neighbor** {*ip-address* | *peer-group-name*} **activate** **Example:** `Router(config-router-af)# neighbor 10.10.10.1 activate`	Enables the neighbor to exchange information for the L2VPN VPLS address family with the local router.

Step 11	**neighbor** {*ip-address*| *peer-group-name*} **send-community**{**both**| **standard**| **extended**}
	Example:
	`Router(config-router-af)# neighbor 10.10.10.1 send-community extended`
		Specifies that a communities attribute should be sent to a BGP neighbor.
		• In this example, an extended communities attribute is sent to the neighbor at 10.10.10.1.

Step 12	Repeat Step 10 and Step 11 to activate other BGP neighbors under an L2VPN address family.	—
Step 13	**exit-address-family** **Example:** `Router(config-router-af)# exit-address-family`	Exits address family configuration mode and returns to router configuration mode.
Step 14	**exit** **Example:** `Router(config-router)# exit`	Exits router configuration mode.
Step 15	**exit** **Example:** `Router(config)# exit`	Exits privileged EXEC mode.
Step 16	**show vfi** **Example:** `Router# show vfi`	(Optional) Displays information about the configured VFI instances.

	Command or Action	Purpose
Step 17	**show ip bgp l2vpn vpls** {**all** | **rd** *vpn-rd*} **Example:** `Router# show ip bgp l2vpn vpls all`	(Optional) Displays information about the Layer2 VPN VPLS address family.

Customizing the VPLS Autodiscovery Settings

Several commands allow you to customize the VPLS environment. You can specify identifiers for the VPLS domain, the route distinguisher, the route target, and the PE router. Perform the following steps to customize these settings.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **l2 vfi** *vfi-name* **autodiscovery**
4. **vpn id** *vpn-id*
5. **vpls-id** {*autonomous-system-number* **:** *nn* | *ip-address* **:** *nn*}
6. **rd** {*autonomous-system-number* **:** *nn* | *ip-address* **:** *nn*}
7. **route-target** [**import** | **export** | **both**] {*autonomous-system-number* **:** *nn* | *ip-address* **:** *nn*}
8. **l2 router-id** *ip-address*
9. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**l2 vfi** *vfi-name* **autodiscovery** **Example:** `Router(config)# l2 vfi vpls1 autodiscovery`	Enables VPLS Autodiscovery on the PE router and enters L2 VFI configuration mode.
Step 4	**vpn id** *vpn-id* **Example:** `Router(config-vfi)# vpn id 10`	Configures a VPN ID for the VPLS domain.

Step 5	**vpls-id** {*autonomous-system-number* **:** *nn* | *ip-address* **:** *nn*}
	Example:
	`Router(config-vfi)# vpls-id 5:300`
		(Optional) Specifies the VPLS domain. This command is optional, because VPLS Autodiscovery automatically generates a VPLS ID using the BGP autonomous system number and the configured VFI VPN ID. You can use this command to change the automatically generated VPLS ID.
		There are two formats for configuring the VPLS ID argument. It can be configured in the *autonomous-system-number:network number* (*ASN:nn*) format, as shown in the example, or it can be configured in the *IP-address:network number* format (*IP-address:nn)*.

Step 6	**rd** {*autonomous-system-number* **:** *nn* | *ip-address* **:** *nn*}
	Example:
	`Router(config-vfi)# rd 2:3`
		(Optional) Specifies the RD to distribute endpoint information. This command is optional, because VPLS Autodiscovery automatically generates an RD using the BGP autonomous system number and the configured VFI VPN ID. You can use this command to change the automatically generated route distinguisher.
		There are two formats for configuring the route distinguisher argument.It can be configured in the *autonomous-system-number:network number* (*ASN:nn*) format, as shown in the example, or it can be configured in the *IP-address:network number* format (*IP-address:nn)*.

Step 7	**route-target** [**import** | **export** | **both**] {*autonomous-system-number* **:** *nn* | *ip-address* **:** *nn*}
	Example:
	`Router(config-vfi)# route-target 600:2222`
		(Optional) Specifies the route target (RT). This command is optional, because VPLS Autodiscovery automatically generates a route target using the lower 6 bytes of the RD and VPLS ID. You can use this command to change the automatically generated route target.
		There are two formats for configuring the route target argument. It can be configured in the *autonomous-system-number:network number* (*ASN:nn*) format, as shown in the example, or it can be configured in the *IP-address:network number* format (*IP-address:nn)*.

Step 8	**l2 router-id** *ip-address* **Example:** `Router(config-vfi)# l2 router-id 10.10.10.10`	(Optional) Specifies a unique identifier for the PE router. This command is optional, because VPLS Autodiscovery automatically generates a Layer 2 router ID using the MPLS global router ID. You can use this command to change the automatically generated ID.

	Command or Action	Purpose
Step 9	**exit** **Example:** `Router(config-vfi)# exit`	Exits L2 VFI configuration mode. Commands take effect after the router exits L2 VFI configuration mode.

Configuration Examples for VPLS Autodiscovery BGP Based

The following examples shows the configuration of a network using VPLS Autodiscovery and VPLS Autodiscovery supported on a route reflector:

- VPLS Autodiscovery BGP Based Basic Example, page 206
- BGP VPLS Autodiscovery Support on Route Reflector Example, page 208

VPLS Autodiscovery BGP Based Basic Example

The figure below show a basic configuration of VPLS Autodiscovery.

Figure 12 Basic VPLS Autodiscovery Configuration

PE1

```
l2 router-id 10.1.1.1
l2 vfi auto autodiscovery
 vpn id 100
!
pseudowire-class mpls
 encapsulation mpls
!
interface Loopback1
 ip address 10.1.1.1 255.255.255.255
!
interface Ethernet0/0
 description Backbone interface
 ip address 192.168.0.1 255.255.255.0
 mpls ip
!
router ospf 1
 log-adjacency-changes
 network 10.1.1.0 0.0.0.255 area 0
 network 172.16.0.0 0.0.0.255 area 0
!
router bgp 1
 no bgp default ipv4-unicast
 bgp log-neighbor-changes
 bgp update-delay 1
 neighbor 10.1.1.2 remote-as 1
 neighbor 10.1.1.2 update-source Loopback1
 neighbor 10.1.1.3 remote-as 1
 neighbor 10.1.1.3 update-source Loopback1
!
 address-family ipv4
 no synchronization
 no auto-summary
 exit-address-family
 !
 address-family l2vpn vpls
 neighbor 10.1.1.2 activate
 neighbor 10.1.1.2 send-community extended
 neighbor 10.1.1.3 activate
 neighbor 10.1.1.3 send-community extended
 exit-address-family
```

PE2

```
l2 router-id 10.1.1.2
l2 vfi auto autodiscovery
 vpn id 100
!
 pseudowire-class mpls
 encapsulation mpls
!
interface Loopback1
 ip address 10.1.1.2 255.255.255.255
!
interface Ethernet0/0
 description Backbone interface
 ip address 192.168.0.2 255.255.255.0
 mpls ip
!
router ospf 1
 log-adjacency-changes
 network 10.1.1.0 0.0.0.255 area 0
 network 172.16.0.0 0.0.0.255 area 0
!
router bgp 1
 no bgp default ipv4-unicast
 bgp log-neighbor-changes
 bgp update-delay 1
 neighbor 10.1.1.1 remote-as 1
 neighbor 10.1.1.1 update-source Loopback1
 neighbor 10.1.1.3 remote-as 1
 neighbor 10.1.1.3 update-source Loopback1
!
 address-family ipv4
 no synchronization
 no auto-summary
 exit-address-family
 !
 address-family l2vpn vpls
 neighbor 10.1.1.1 activate
 neighbor 10.1.1.1 send-community extended
 neighbor 10.1.1.3 activate
 neighbor 10.1.1.3 send-community extended
 exit-address-family
```

PE3

```
l2 router-id 10.1.1.3
l2 vfi auto autodiscovery
 vpn id 100
!
pseudowire-class mpls
 encapsulation mpls
!
interface Loopback1
 ip address 10.1.1.3 255.255.255.255
!
interface Ethernet0/0
 description Backbone interface
 ip address 192.168.0.3 255.255.255.0
 mpls ip
!
router ospf 1
 log-adjacency-changes
 network 10.1.1.0 0.0.0.255 area 0
 network 172.16.0.0 0.0.0.255 area 0
!
router bgp 1
 no bgp default ipv4-unicast
 bgp log-neighbor-changes
 bgp update-delay 1
 neighbor 10.1.1.1 remote-as 1
 neighbor 10.1.1.1 update-source Loopback1
 neighbor 10.1.1.2 remote-as 1
 neighbor 10.1.1.2 update-source Loopback1
!
 address-family ipv4
 no synchronization
 no auto-summary
 exit-address-family
 !
 address-family l2vpn vpls
 neighbor 10.1.1.1 activate
 neighbor 10.1.1.1 send-community extended
 neighbor 10.1.1.2 activate
 neighbor 10.1.1.2 send-community extended
 exit-address-family
```

BGP VPLS Autodiscovery Support on Route Reflector Example

In the following example, a host named PE-RR (indicating Provider Edge-Route Reflector) is configured as a route reflector capable of reflecting VPLS prefixes. The VPLS address family is configured by **address-family l2vpn vpls** below.

```
hostname PE-RR
!
router bgp 1
 bgp router-id 1.1.1.3
 no bgp default route-target filter
 bgp log-neighbor-changes
neighbor iBGP_PEERS peer-group
neighbor iBGP_PEERS remote-as 1
neighbor iBGP_PEERS update-source Loopback1
neighbor 1.1.1.1 peer-group iBGP_PEERS
neighbor 1.1.1.2 peer-group iBGP_PEERS
!
address-family l2vpn vpls
  neighbor iBGP_PEERS send-community extended
  neighbor iBGP_PEERS route-reflector-client
  neighbor 1.1.1.1 peer-group iBGP_PEERS
  neighbor 1.1.1.2 peer-group iBGP_PEERS
exit-address-family
```

Additional References

The following sections provide references related to the VPLS Autodiscovery: BGP Based feature.

Related Documents

Related Topic	Document Title
Virtual Private LAN Services on the Cisco 7600 series router	“Virtual Private LAN Services on the Optical Services Modules” chapter in the Cisco 7600 Series Router Cisco IOS Software Configuration Guide , Release 12.2SR

Related Topic	Document Title
L2 VPNs on the Cisco 7600 router	Configuration information for Layer 2 VPNs on the Cisco 7600 router is included in the following documents: - The “Configuring PFC3BXL and PFC3B Mode Multiprotocol Label Switching” module of the Cisco 7600 Series Cisco IOS Software Configuration Guide , Release 12.2SR - The “Configuring Multiprotocol Label Switching on the Optical Services Modules” module of the OSM Configuration Note , Release 12.2SR - The “Configuring Multiprotocol Label Switching on FlexWAN and Enhanced FlexWAN Modules” module of the FlexWAN and Enhanced FlexWAN Modules Configuration Guide - The “Configuring Any Transport over MPLS on a SIP” section of the Cisco 7600 Series Router SIP, SSC, and SPA Software Configuration Guide - The “Configuring AToM VP Cell Mode Relay Support” section of the Cisco 7600 Series Router SIP, SSC, and SPA Software Configuration Guide - The Release Notes for Cisco IOS Release 12.2SR for the Cisco 7600 Series Routers
MPLS Commands	*Cisco IOS Multiprotocol Label Switching Command Reference*

Standards

Standard	Title
draft-ietf-l2vpn-signaling-08.txt	*Provisioning, Autodiscovery, and Signaling in L2VPNs*
draft-ietf-l2vpn-vpls-bgp-08.8	*Virtual Private LAN Service (VPLS) Using BGP for Autodiscovery and Signaling*
draft-ietf-mpls-lsp-ping-03.txt	*Detecting MPLS Data Plane Failures*
draft-ietf-pwe3-vccv-01.txt	*Pseudo-Wire (PW) Virtual Circuit Connection Verification (VCCV)*

MIBs

MIB	MIBs Link
• CISCO-IETF-PW-MIB (PW-MIB) • CISCO-IETF-PW-MPLS-MIB (PW-MPLS-MIB) • CISCO-IETF-PW-ENET-MIB (PW-ENET-MIB) • CISCO-IETF-PW-FR-MIB (PW-FR-MIB) • CISCO-IETF-PW-ATM-MIB (PW-ATM-MIB)	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
RFC 3916	*Requirements for Pseudo-wire Emulation Edge-to-Edge (PWE3)*
RFC 3981	*Pseudo Wire Emulation Edge-to-Edge Architecture*

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password. If you have a valid service contract but do not have a user ID or password, you can register on Cisco.com.	http://www.cisco.com/techsupport

Feature Information for VPLS Autodiscovery BGP Based

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 21 Feature Information for VPLS Autodiscovery: BGP Based

Feature Name	Releases	Feature Information
VPLS Autodiscovery: BGP Based	12.2(33)SRB	VPLS Autodiscovery enables each Virtual Private LAN Service (VPLS) provider edge (PE) router to discover which other PE routers are part of the same VPLS domain. In 12.2(33)SRB, this feature was introduced on the Cisco 7600 router. The following commands were introduced or modified for this feature: • **auto-route-target** • **l2 router-id** • **l2 vfi autodiscovery** • **neighbor (VPLS)** • **rd (VPLS)** • **route-target (VPLS)** • **show mpls l2transport vc** • **show vfi** • **show xconnect** • **vpls-id** • **xconnect**
BGP VPLS Autodiscovery Support on Route Reflector	12.2(33)SRE	This feature was introduced on the Cisco 7600 series routers. This feature is documented in the following sections:

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.

H-VPLS N-PE Redundancy for QinQ and MPLS Access

The H-VPLS N-PE Redundancy for QinQ feature and the H-VPLS N-PE Redundancy for MPLS Access feature enable two network provider edge (N-PE) routers to provide failover services to a user provider edge (U-PE) router in a hierarchical virtual private LAN service (H-VPLS). Having redundant N-PE routers provides improved stability and reliability against link and node failures. The VPLS MAC Address Withdrawal feature provides faster convergence by removing (or unlearning) MAC addresses that have been dynamically learned. This document explains how to implement these features.

- Finding Feature Information, page 213
- Prerequisites for H-VPLS N-PE Redundancy for QinQ and MPLS Access, page 213
- Restrictions for H-VPLS N-PE Redundancy for QinQ and MPLS Access, page 214
- Information About H-VPLS N-PE Redundancy for QinQ and MPLS Access, page 215
- How to Configure H-VPLS N-PE Redundancy for QinQ and MPLS Access, page 217
- Configuration Examples for H-VPLS N-PE Redundancy for QinQ and MPLS Access, page 220
- Additional References, page 221
- Feature Information for H-VPLS N-PE Redundancy for QinQ and MPLS Access, page 223
- Glossary, page 225

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for H-VPLS N-PE Redundancy for QinQ and MPLS Access

- Before configuring the H-VPLS N-PE Redundancy for QinQ and MPLS Access feature, configure your H-VPLS network and make sure it is operating correctly. For more information about configuring the H-VPLS network, see the “ Configuring VPLS” section in Configuring Multiprotocol Label Switching on the Optical Services Modules .

- Make sure that the PE-to-customer edge (CE) interface is configured with a list of allowed VLANs. For more information, see the " Configuring VPLS" section in Configuring Multiprotocol Label Switching on the Optical Services Modules .
- To provide faster convergence, you can enable the MPLS Traffic Engineering: Fast Reroute feature in the Multiprotocol Label Switching (MPLS) core. For more information about MPLS traffic engineering, see the "MPLS Traffic Engineering (TE)—Fast Reroute (FRR) Link and Node Protection" section in the *Cisco IOS Multiprotocol Label Switching Configuration Guide* .
- Enable the L2VPN Pseudowire Redundancy feature on the U-PE routers for MPLS access. For information about configuring the L2VPN Pseudowire Redundancy feature, see the "L2VPN Pseudowire Redundancy" section in the *Cisco IOS Wide-Area Networking Configuration Guide* .
- When configuring Multiple Spanning Tree Protocol (MSTP), specify that one of the N-PE routers is the root by assigning it the lowest priority using the **spanning-tree mst** *instance-id* **priority** *priority*command.

For information about configuring MSTP, see the "Configuring MST Instance Parameters" section in the Cisco 7600 Series Cisco IOS Software Configuration Guide.

- When configuring MSTP, make sure that each router participating in the spanning tree is in the same region and is the same revision by issuing the **revision**, **name**, and **instance** commands in MST configuration mode. For more information on configuring these MSTP parameters, see the " Configuring Spanning Tree and IEEE 802.1s MST" section in the Cisco 7600 Series Cisco IOS Software Configuration Guide.

Restrictions for H-VPLS N-PE Redundancy for QinQ and MPLS Access

- The H-VPLS N-PE Redundancy for QinQ and MPLS Access feature cannot be used with the VPLS Autodiscovery feature on pseudowires that attach to U-PE routers. When you create the VPLS, you can manually create the virtual forwarding interface (VFI).
- You cannot configure more than one pseudowire to carry the bridge protocol data unit (BPDU) information between the N-PE routers. If you attempt to enter the **forward permit l2protocol all** command for multiple VFIs, an error message is displayed.
- You cannot configure a local loopback address as a neighbor when you configure the H-VPLS N-PE Redundancy for QinQ and MPLS Access feature on N-PE routers. If you do so, the following error is displayed:

```
VPLS local switching to peer address not supported
```

- Only two N-PE routers can be connected to each U-PE router.
- For a list of supported hardware for this feature, see the Release Notes for Cisco IOS Release 12.2SR for the Cisco 7600 Series Routers.
- The spanning-tree mode must be MSTP for the H-VPLS N-PE Redundancy for QinQ and MPLS Access feature. If the spanning-tree mode changes, the H-VPLS N-PE Redundancy for QinQ and MPLS Access feature may not work correctly, even though the pseudowire that carries the BPDU information still exists and the H-VPLS N-PE Redundancy feature is still configured.

Information About H-VPLS N-PE Redundancy for QinQ and MPLS Access

- How H-VPLS N-PE Redundancy for QinQ and MPLS Access Works, page 215
- VPLS MAC Address Withdrawal, page 216

How H-VPLS N-PE Redundancy for QinQ and MPLS Access Works

In a network configured with the H-VPLS N-PE Redundancy for QinQ and MPLS Access feature, the U-PE router is connected to two N-PE routers. This feature provides a level of redundancy that can tolerate both link and device faults. If a failure occurs in the network that disables one N-PE router from transmitting data, the other N-PE router takes over. This feature works with both QinQ access based on MSTP and MPLS access based on pseudowire redundancy.

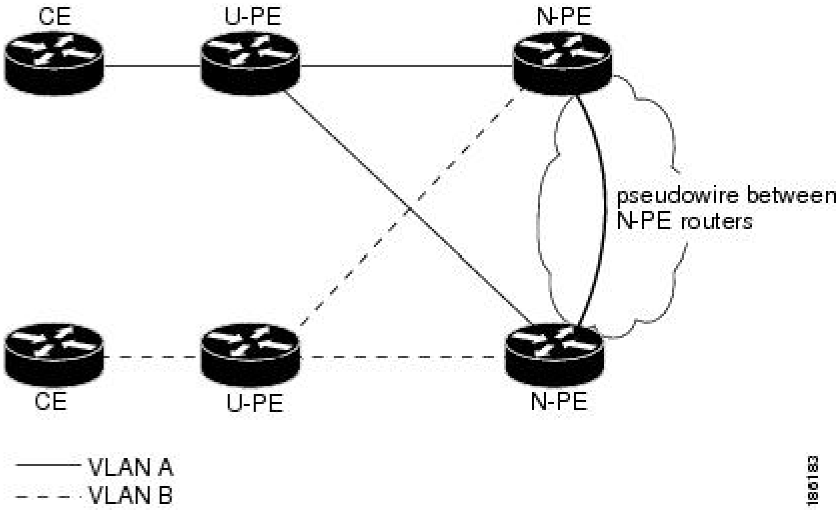
- H-VPLS N-PE Redundancy with QinQ Access Based on MSTP, page 215
- H-VPLS N-PE Redundancy with MPLS Access Based on Pseudowire Redundancy, page 216

H-VPLS N-PE Redundancy with QinQ Access Based on MSTP

H-VPLS N-PE redundancy with QinQ access uses the MSTP running on the N-PE routers and U-PE routers in an H-VPLS network. A pseudowire running between N-PE routers carries only MSTP BPDUs. The pseudowire running between the N-PE routers is always up and is used to create a loop path between N-PE routers so that MSTP will block one of the redundant paths between the U-PE router and the N-PE routers. If the primary N-PE router or the path to it fails, MSTP will enable the path to the backup N-PE router.

The figure below shows an H-VPLS network with redundant access. Each U-PE router has two trunk connections, one to each N-PE router. Between the two N-PE routers is a pseudowire to provide a loop path for MSTP BPDUs. The network topology allows for the backup N-PE router to take over if the primary N-PE router or the path to it fails.

Figure 13 H-VPLS N-PE Redundancy with QinQ Access Based on MSTP

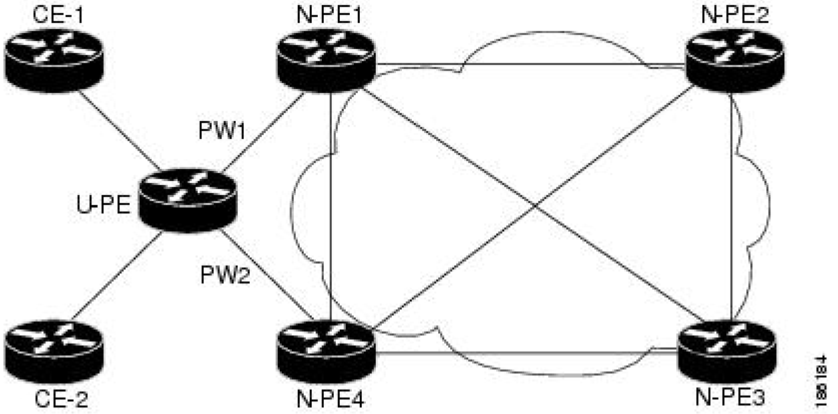


H-VPLS N-PE Redundancy with MPLS Access Based on Pseudowire Redundancy

For H-VPLS redundancy with MPLS access based on pseudowire redundancy, the MPLS network has pseudowires to the VPLS core N-PE routers.

As shown in the figure below, one pseudowire transports data between the U-PE router and its peer N-PE routers. When a failure occurs along the path of the U-PE router, the backup pseudowire and the redundant N-PE router become active and start transporting data.

Figure 14 H-VPLS N-PE Redundancy for QinQ and MPLS Access with MPLS Access Based on Pseudowire Redundancy



VPLS MAC Address Withdrawal

The VPLS MAC Address Withdrawal feature provides faster convergence by removing (or unlearning) MAC addresses that have been dynamically learned. A Label Distribution Protocol (LDP)-based MAC address withdrawal message is used for this purpose. A MAC list Type Length Value (TLV) is part of the MAC address withdrawal message.

The **debug mpls ldp messages** and **debug mpls ldp session io** commands support monitoring of MAC address withdrawal messages being exchanged between LDP peers. Any Transport over MPLS (AToM) may provide other means to display or monitor MAC address withdrawal messages. The Tag Distribution Protocol (TDP) is not supported as AToM uses only LDP for the MAC address withdrawal message.

PE routers learn the remote MAC addresses and directly attached MAC addresses on customer-facing ports by deriving the topology and forwarding information from packets originating at customer sites. To display the number of MAC address withdrawal messages, enter the **show mpls l2transport vc detail** command, as shown in the following example:

```
Router# show mpls l2transport vc detail
Local interface: VFI TEST VFI up
  MPLS VC type is VFI, interworking type is Ethernet
  Destination address: 10.1.1.1, VC ID: 1000, VC status: up
    Output interface: Se2/0, imposed label stack {17}
    Preferred path: not configured
    Default path: active
    Next hop: point2point
  Create time: 00:04:34, last status change time: 00:04:15
  Signaling protocol: LDP, peer 10.1.1.1:0 up
    Targeted Hello: 10.1.1.1(LDP Id) -> 10.1.1.1
    MPLS VC labels: local 16, remote 17
    Group ID: local 0, remote 0
    MTU: local 1500, remote 1500
    Remote interface description:
    MAC Withdraw: sent 5, received 3
  Sequencing: receive disabled, send disabled
  VC statistics:
    packet totals: receive 0, send 0
    byte totals:   receive 0, send 0
    packet drops:  receive 0, send 0
```

- How MAC Address Withdrawal Works with H-VPLS N-PE Redundancy with QinQ Access, page 217
- How MAC Address Withdrawal Works with H-VPLS N-PE Redundancy with MPLS Access, page 217

How MAC Address Withdrawal Works with H-VPLS N-PE Redundancy with QinQ Access

If a failure occurs in the customer-switched network, a spanning-tree Topology Change Notification (TCN) is issued to the N-PE router, which issues an LDP-based MAC address withdrawal message to the peer N-PE routers and flushes its MAC address table.

How MAC Address Withdrawal Works with H-VPLS N-PE Redundancy with MPLS Access

If the pseudowire between the U-PE router and N-PE router fails, then the L2VPN Pseudowire Redundancy feature on the U-PE router activates the standby pseudowire. In addition, the U-PE router sends an LDP MAC address withdrawal request to the new N-PE router, which forwards the message to all pseudowires in the VPLS core and flushes its MAC address table.

If a switched virtual interface (SVI) on the N-PE router fails, the L2VPN Pseudowire Redundancy feature activates the standby pseudowire and the U-PE router sends a MAC withdrawal message to the newly active N-PE router.

For information about the L2VPN Pseudowire Redundancy feature, see the “L2VPN Pseudowire Redundancy” feature module.

How to Configure H-VPLS N-PE Redundancy for QinQ and MPLS Access

- Configuring the VPLS Pseudowire Between the N-PE Routers, page 217
- Configuring the SVI for the Native VLAN, page 219

Configuring the VPLS Pseudowire Between the N-PE Routers

Configuring N-PE redundancy in an H-VPLS network requires two steps. First, you must define the VPLS pseudowire for transporting BPDU data. Then, you must connect that pseudowire to the native VLAN. This configuration provides a redundancy that provides improved reliability against link and node failures.

Review the Prerequisites for H-VPLS N-PE Redundancy for QinQ and MPLS Access, page 213 to ensure that your H-VPLS network is configured and operating correctly.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **l2 vfi** *name* **manual**
4. **vpn id** *id-number*
5. **forward permit l2protocol all**
6. **neighbor** *remote-router-id vc-id* {**encapsulation** *encapsulation-type* | **pw-class** *pw-name*} [**no-split-horizon**]
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**l2 vfi** *name* **manual** **Example:** `Router(config)# l2 vfi vfitest1 manual`	Creates a Layer 2 VFI and enters Layer 2 VFI manual configuration mode.
Step 4	**vpn id** *id-number* **Example:** `Router(config-vfi)# vpn id 200`	Specifies the VPN ID.
Step 5	**forward permit l2protocol all** **Example:** `Router(config-vfi)# forward permit l2protocol all`	Creates a pseudowire that is to be used to transport BPDU data between the two N-PE routers.

	Command or Action	Purpose
Step 6	**neighbor** *remote-router-id vc-id* {**encapsulation** *encapsulation-type* | **pw-class** *pw-name*} [**no-split-horizon**] **Example:** `Router(config-vfi)# neighbor 10.2.2.2 3 encapsulation mpls`	Specifies the peer IP address of the redundant N-PE router and the type of tunnel signaling and encapsulation mechanism.

Step 7	**end**
	Example:
	`Router(config-vfi)# end`
		Exits Layer 2 VFI manual configuration mode and returns to privileged EXEC mode.

Configuring the SVI for the Native VLAN

Perform the following task to configure the switched virtual interface for the native VLAN and verify that it is correctly configured.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface vlan** *vlanid*
4. **xconnect vfi** *vfi-name*
5. **end**
6. **show vfi** *vfi-name*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**interface vlan** *vlanid* **Example:** `Router(config)# interface vlan 23`	Creates a dynamic SVI. • To make the SVI active when you create a VLAN, you must configure the VLAN with at least one physical interface that is in the “up” state. Use the **show vfi** command to display the status of the SVI. The state field will display “up” when the SVI is active.
Step 4	**xconnect vfi** *vfi-name* **Example:** `Router(config)# xconnect vfi vfitest1`	Specifies the Layer 2 VFI that you are binding to the VLAN port.

Step 5	**end**
	Example:
	`Router(config-vfi)# end`
		Ends the current configuration session and returns to privileged EXEC mode.

Step 6	**show vfi** *vfi-name* **Example:** `Router# show vfi VPLS-2`	(Optional) Displays information about the pseudowire between the two N-PE routers so that you can verify that the H-VPLS N-PE Redundancy for QinQ and MPLS Access feature is correctly configured.

Step 7	**end**
	Example:
	`Router# end`
		Exits privileged EXEC mode and returns to user EXEC mode.

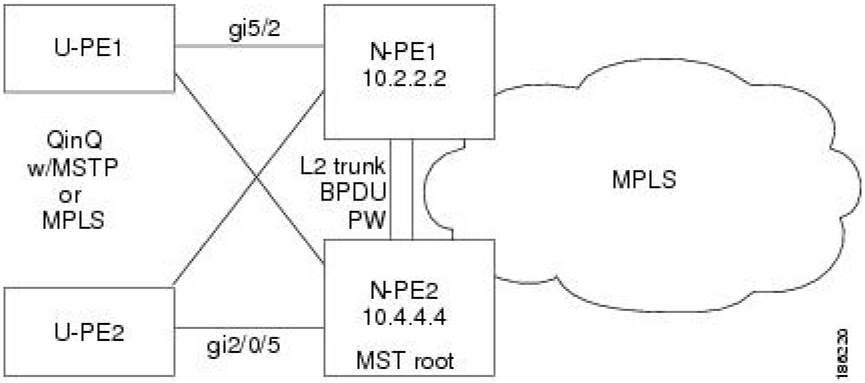
Configuration Examples for H-VPLS N-PE Redundancy for QinQ and MPLS Access

- Example H-VPLS N-PE Redundancy for QinQ Access, page 221

Example H-VPLS N-PE Redundancy for QinQ Access

The figure below shows a configuration that is set up for H-VPLS N-PE redundancy with QinQ access.

Figure 15 H-VPLS N-PE Redundancy with QinQ Access Topology



The table below shows the configuration of two N-PE routers for H-VPLS N-PE redundancy with QinQ access.

Table 22 Example: H-VPLS N-PE Redundancy for QinQ Access

N-PE1	N-PE2
l2 vfi l2trunk manual vpn id 10 forward permit l2protocol all neighbor 10.4.4.4 encapsulation mpls ! interface Vlan1 no ip address xconnect vfi l2trunk ! spanning-tree mode mst spanning-tree extend system-id ! spanning-tree mst configuration revision 10 instance 1 vlan 20 ! interface GigabitEthernet5/2 switchport switchport trunk encapsulation dot1q switchport trunk allowed vlan 20 switchport mode trunk	l2 vfi l2trunk manual vpn id 10 forward permit l2protocol all neighbor 10.2.2.2 encapsulation mpls ! interface Vlan1 no ip address xconnect vfi l2trunk ! spanning-tree mode mst spanning-tree extend system-id ! spanning-tree mst configuration revision 10 instance 1 vlan 20 ! spanning-tree mst 1 priority 0 ! interface GigabitEthernet2/0/5 switchport switchport trunk allowed vlan 20 switchport mode trunk mls qos trust dscp

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
MPLS commands	*Cisco IOS Multiprotocol Label Switching Command Reference*
L2VPN pseudowire redundancy	"L2VPN Pseudowire Redundancy" section in the *Cisco IOS Wide-Area Networking Configuration Guide*
H-VPLS	"Configuring VPLS" section in the Configuring Multiprotocol Label Switching on the Optical Services Modules
Multiple spanning tree configuration	"Configuring MST Instance Parameters" section in the *Cisco 7600 Series Cisco IOS Software Configuration Guide*
MPLS traffic engineering	"MPLS Traffic Engineering (TE)—Fast Reroute (FRR) Link and Node Protection" section in the *Cisco IOS Multiprotocol Label Switching Configuration Guide*
Configuring MSTP	"Configuring MST Instance Parameters" section in the *Cisco 7600 Series Cisco IOS Software Configuration Guide*
Supported hardware on the Cisco 7600 series routers	Release Notes for Cisco IOS Release 12.2SR for the Cisco 7600 Series Routers

Standards

Standard	Title
http://www.ietf.org/rfc/rfc4447.txt	*Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP)*
http://www3.ietf.org/proceedings/06mar/IDs/draft-ietf-l2vpn-vpls-ldp-08.txt	*Virtual Private LAN Services over MPLS*
http://www.ietf.org/internet-drafts/draft-ietf-pwe3-segmented-pw-02.txt	*Segmented Pseudo Wire*
draft-ietf-pwe3-vccv-10.txt	*Pseudo Wire Virtual Circuit Connectivity Verification (VCCV)*
draft-ietf-pwe3-oam-msg-map-03.txt	*Pseudo Wire (PW) OAM Message Mapping*

MIBs

MIB	MIBs Link
Pseudowire Emulation Edge-to-Edge MIBs for Ethernet, Frame Relay, and ATM Services	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
None	—

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for H-VPLS N-PE Redundancy for QinQ and MPLS Access

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 23 Feature Information for H-VPLS N-PE Redundancy for QinQ and MPLS Access

Feature Name	Releases	Feature Information
H-VPLS N-PE Redundancy for MPLS Access	12.2(33)SRC 12.2(50)SY	The H-VPLS N-PE Redundancy for MPLS Access feature enables two N-PE routers to provide redundancy to a U-PE router in an H-VPLS. Having redundant N-PE routers provides improved stability and reliability against link and node failures. In Cisco IOS Release 12.2(33)SRC, this feature was introduced on the Cisco 7600 series routers. The following sections provide information about this feature: The following commands were introduced or modified: **forward permit l2protocol**, **show mpls l2transport vc**.
H-VPLS N-PE Redundancy for QinQ Access	12.2(33)SRC	The H-VPLS N-PE Redundancy for QinQ Access feature provides the capability to dual-home a given U-PE router to two N-PE routers in order to provide protection against link and node failures. In Cisco IOS Release 12.2(33)SRC, this feature was introduced on the Cisco 7600 series routers. The following sections provide information about this feature: The following commands were introduced or modified: **forward permit l2protocol**, **show mpls l2transport vc**.

Feature Name	Releases	Feature Information
VPLS MAC Address Withdrawal	12.2(33)SXI4 12.2(50)SY XE 3.5S 15.2(1)S	The VPLS MAC Address Withdrawal feature provides faster convergence by removing (or unlearning) MAC addresses that have been dynamically learned. The following sections provide information about this feature: "MAC Address Withdrawal" In Cisco IOS XE Release 3.5S, this feature was introduced on the Cisco ASR 1000 Series Aggregation Services Routers. In Cisco IOS XE Release 3.5S, support was added for the Cisco ASR 903 Router. In Cisco IOS Release 15.2(1)S, this feature was integrated.

Glossary

CE router —customer edge router. A router that belongs to a customer network, which connects to a PE router to utilize MPLS VPN network services.

LAN —local-area network. High-speed, low-error data network covering a relatively small geographic area. LANs connect workstations, peripherals, terminals, and other devices in a single building or other geographically limited areas.

MPLS —Multiprotocol Label Switching. A packet-forwarding technology, used in the network core, that applies data link layer labels to tell switching nodes how to forward data, resulting in faster and more scalable forwarding than network layer routing normally can do.

MSTP —Multiple Spanning Tree Protocol. MSTP enables multiple VLANs to be mapped to the same spanning-tree instance, reducing the number of spanning-tree instances needed to support a large number of VLANs.

N-PE —network provider edge router. This router acts as a gateway between the MPLS core and edge domains.

PE router —provider edge router. The PE router is the entry point into the service provider network. The PE router is typically deployed on the edge of the network and is administered by the service provider.

pseudowire —A pseudowire is a virtual connection that, in the context of VPLS, connects two SVIs. It is a mechanism that carries the elements of an emulated service from one PE router to one or more PE routers over a packet switched network (PSN). A pseudowire is bidirectional and consists of a pair of unidirectional MPLS virtual circuits (VCs). A pseudowire can be used to connect a point-to-point circuit.

QinQ —An IEEE 802.1Q VLAN tunnel. A mechanism for constructing multipoint Layer 2 VPN using Ethernet switches.

redundancy —The duplication of devices, services, or connections so that, in the event of a failure, they can perform the work of those that failed.

router —A network layer device that uses one or more metrics to determine the optimal path along which network traffic should be forwarded. Routers forward packets from one network to another based on network layer information.

spanning tree —Loop-free subset of a network topology.

U-PE —user provider edge router. This router connects CE routers to the service.

VFI —virtual forwarding instance. A VFI is a collection of data structures used by the data plane, software-based or hardware-based, to forward packets to one or more VCs.

VLAN —Virtual LAN. Group of devices on one or more LANs that are configured (using management software) 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.

VPLS —Virtual Private LAN Service. VPLS describes an architecture that delivers Layer 2 service that emulates an Ethernet LAN across a wide-area network (WAN) and inherits the scaling characteristics of a LAN.

VPLS redundancy —Also called N-PE redundancy. Allows U-PEs to be dual-honed (to their N-PEs) in a loop-free topology with MPLS or QinQ as the access or aggregation domain.

VPN —Virtual Private Network. Allows IP traffic to travel securely over public TCP/IP networks and the Internet by encapsulating and encrypting all IP packets. VPN uses a tunnel to encrypt all information at the IP level.

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.

L2VPN Multisegment Pseudowires

The L2VPN Multisegment Pseudowires feature enables you to configure two or more Layer 2 pseudowire segments that function as a single pseudowire. Layer 2 Virtual Private Network (L2VPN) multisegment pseudowires span multiple cores or autonomous systems of the same or different carrier networks. L2VPN multisegment pseudowires are also used in L2VPN Virtual Private LAN Services (VPLS) Inter-AS Option B networks.

This document explains Multiprotocol Label Switching (MPLS) Operations, Administration, and Maintenance (OAM) Support for L2VPN Multisegment Pseudowires and the MPLS OAM Support for the L2VPN VPLS Inter-AS Option B feature. These features allow you to use **ping mpls** and **trace mpls** commands to ensure pseudowire connectivity.

- Finding Feature Information, page 227
- Prerequisites for L2VPN Multisegment Pseudowires, page 227
- Restrictions for L2VPN Multisegment Pseudowires, page 228
- Information About L2VPN Multisegment Pseudowires, page 228
- How to Configure L2VPN Multisegment Pseudowires, page 230
- Configuration Examples for L2VPN Multisegment Pseudowires, page 238
- Additional References, page 241
- Feature Information for L2VPN Multisegment Pseudowires, page 243

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for L2VPN Multisegment Pseudowires

Before configuring this feature, see the following documents:

- Any Transport over MPLS
- L2VPN Pseudowire Switching
- MPLS LSP Ping/Traceroute for LDP/TE, and LSP Ping for VCCV
- Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP) (RFC 4447)

Restrictions for L2VPN Multisegment Pseudowires

- Only Multiprotocol Label Switching (MPLS) Layer 2 pseudowires are supported.
- In Cisco IOS Release 12.3(33)SRE, only static configuration of the pseudowires is supported for the L2VPN Multisegment Pseudowires feature.
- In Cisco IOS Release 15.1(1)S, dynamic configuration of the pseudowires is supported and required for the L2VPN VPLS Inter-AS Option B feature.
- In Cisco IOS Release 12.3(33)SRE, only pseudowires advertised with forwarding equivalence class (FEC) 128 are supported for the L2VPN Multisegment Pseudowires feature. FEC 129 is not supported.
- In Cisco IOS Release 15.1(1)S, FEC 129 is supported and used to exchange information about the pseudowires for the L2VPN VPLS Inter-AS Option B feature.
- The S-PE router is limited to 1600 pseudowires.

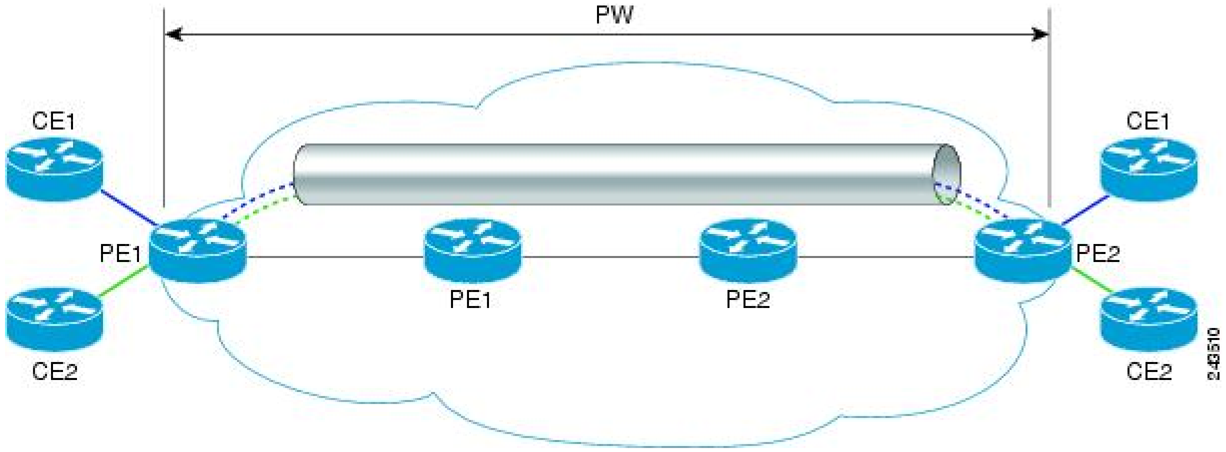
Information About L2VPN Multisegment Pseudowires

- L2VPN Pseudowire Defined, page 228
- L2VPN Multisegment Pseudowire Defined, page 229
- MPLS OAM Support for Multisegment Pseudowires, page 229
- MPLS OAM Support for L2VPN VPLS Inter-AS Option B, page 230

L2VPN Pseudowire Defined

An L2VPN pseudowire (PW) is a tunnel established between two provider edge (PE) routers across the core carrying the Layer 2 payload encapsulated as MPLS data, as shown in the figure below. This helps carriers migrate from traditional Layer 2 networks such as Frame Relay and ATM to an MPLS core. The PWs between two PE routers are located within the same autonomous system (AS). Routers PE1 and PE2 are called terminating PE routers (T-PEs). Attachment circuits are bounded to the PW on these PE routers.

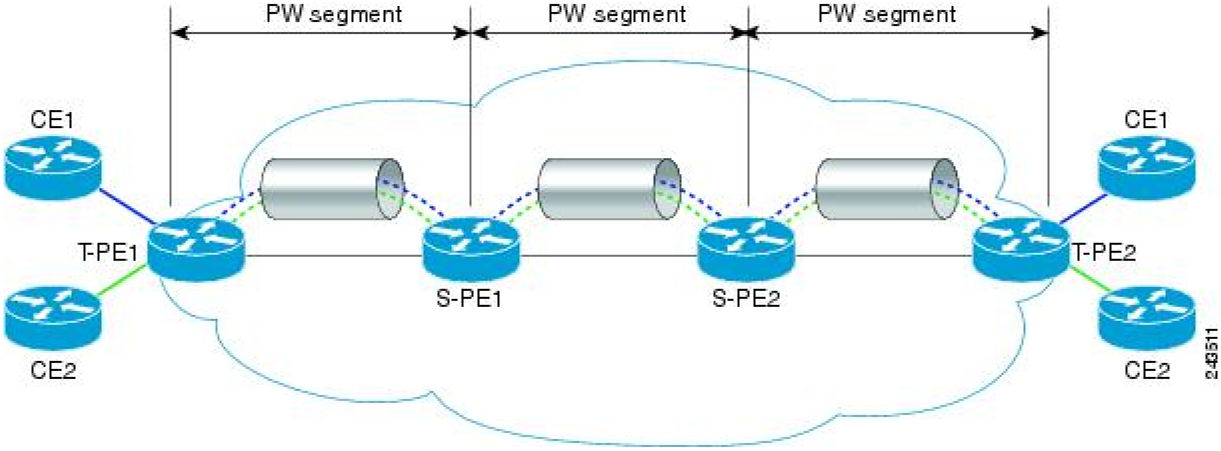
Figure 16 An L2VPN Pseudowire



L2VPN Multisegment Pseudowire Defined

An L2VPN multisegment pseudowire (MS-PW) is a set of two or more PW segments that function as a single PW, as shown in the figure below. It is also known as switched PW. MS-PWs span multiple cores or autonomous systems of the same or different carrier networks. An L2VPN MS-PW can include up to 254 PW segments.

Figure 17 A Multisegment Pseudowire



The end routers are called terminating PE routers (T-PEs), and the switching routers are called S-PE routers. The S-PE router terminates the tunnels of the preceding and succeeding PW segments in an MS-PW. The S-PE router can switch the control and data planes of the preceding and succeeding PW segments of the MS-PW. An MS-PW is declared to be up when all the single-segment PWs are up. For more information, see the L2VPN Pseudowire Switching document.

With the L2VPN Multisegment Pseudowire feature introduced in Cisco IOS Release 12.2(33)SRE, the pseudowires are created statically, and FEC 128 information is used to exchange the information about each AS.

MPLS OAM Support for Multisegment Pseudowires

You can use the **ping mpls** and **trace mpls**commands to verify that all the segments of the MPLS multisegment pseudowire are operating.

You can use the **ping mpls** command to verify connectivity at the following pseudowire points:

- From one end of the pseudowire to the other
- From one of the pseudowires to a specific segment
- The segment between two adjacent S-PE routers

You can use the **trace mpls**command to verify connectivity at the following pseudowire points:

- From one end of the pseudowire to the other
- From one of the pseudowires to a specific segment
- The segment between two adjacent S-PE routers
- A range of segments

MPLS OAM Support for L2VPN VPLS Inter-AS Option B

The L2VPN VPLS Inter-AS Option B feature introduced in Cisco IOS Release 15.1(1)S uses multisegment pseudowires to connect Autonomous System Border Routers (ASBRs) in different autonomous systems. With this feature, the pseudowires are created dynamically, and FEC 129 information is used to exchange the information about each ASBR.

The differences between static multisegment pseudowires and dynamic multisegment pseudowires are listed in the table below.

Table 24 Comparison of Static and Dynamic Multisegment Pseudowires

Static Multisegment Pseudowires	Dynamic Multisegment Pseudowires
Are statically stitched and dynamically signalled.	Are dynamically stitched and dynamically signalled.
Label Distribution Protocol (LDP) exchanges the type length value (TLV) and FEC 128 information is exchanged between segments.	Border Gateway Protocol (BGP) exchanges the TLV and FEC 129 information is exchanged between ASBRs.

For more information about the L2VPN VPLS Inter-AS Option B feature, see L2VPN VPLS Inter-AS Option B.

How to Configure L2VPN Multisegment Pseudowires

- Configuring L2VPN Multisegment Pseudowires, page 230
- Displaying Information About the L2VPN Multisegment Pseudowires, page 233
- Verifying Multisegment Pseudowires with ping mpls and trace mpls Commands, page 234
- Verifying L2VPN VPLS Inter-AS Option B with ping mpls and trace mpls Commands, page 236

Configuring L2VPN Multisegment Pseudowires

Perform the following steps on the S-PE routers to create L2VPN multisegment pseudowires.

- Cisco 7600 Router-Specific Instructions, page 230

Cisco 7600 Router-Specific Instructions

If the Cisco 7600 router is the penultimate hop router connected to the S-PE or T-PE router, issue the following commands on the S-PE or T-PE routers:

- **mpls ldp explicit-null**
- **no mls mpls explicit-null propagate-ttl**

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mpls label protocol ldp**
4. **mpls ldp router-id** *interface* **force**
5. **pseudowire-class** *name*
6. **encapsulation mpls**
7. **switching tlv**
8. **exit**
9. **l2 vfi** *name* **point-to-point**
10. **description** *string*
11. **neighbor** *ip-address vcid {* **encapsulation mpls** | **pw-class** *pw-class-name }*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**mpls label protocol ldp** **Example:** `Router(config)# mpls label protocol ldp`	Configures the use of Label Distribution Protocol (LDP) on all interfaces.
Step 4	**mpls ldp router-id** *interface* **force** **Example:** `Router(config)# mpls ldp router-id loopback0 force`	Specifies the preferred interface for determining the LDP router ID.

	Command or Action	Purpose
Step 5	**pseudowire-class** *name* **Example:** `Router(config)# pseudowire-class atom`	Establishes a pseudowire class with a name that you specify, and enters pseudowire class configuration mode.
Step 6	**encapsulation mpls** **Example:** `Router(config-pw-class)# encapsulation mpls`	Specifies the tunneling encapsulation. • For MPLS L2VPNs, the encapsulation type is **mpls**.
Step 7	**switching tlv** **Example:** `Router(config-pw-class)# switching tlv`	(Optional) Enables the advertisement of the switching point type-length variable (TLV) in the label binding. • This command is enabled by default.
Step 8	**exit** **Example:** `Router(config-pw-class)# exit`	Exits pseudowire class configuration mode.
Step 9	**l2 vfi** *name* **point-to-point** **Example:** `Router(config)# l2 vfi atomtunnel point-to-point`	Creates a point-to-point Layer 2 virtual forwarding interface (VFI) and enters VFI configuration mode.
Step 10	**description** *string* **Example:** `Router(config-vfi)# description segment1`	Provides a description of the switching provider edge router for a multisegment pseudowire.

Step 11	**neighbor** *ip-address vcid {* **encapsulation mpls** | **pw-class** *pw-class-name }*
	Example:
	`Router(config-vfi)# neighbor 10.0.0.1 100 pw-class mpls`
		Sets up an emulated VC.
		• Specify the IP address and the VC ID of the peer router. Also specify the pseudowire class to use for the emulated VC.
		Note Only two **neighbor**commands are allowed for each **l2 vfi point-to-point** command.

Displaying Information About the L2VPN Multisegment Pseudowires

Perform the following task to display the status of L2VPN multisegment pseudowires.

SUMMARY STEPS

1. **show mpls l2transport binding**
2. **show mpls l2transport vc detail**

DETAILED STEPS

Step 1 **show mpls l2transport binding**

Use the **show mpls l2transport binding** command to display information about the pseudowire switching point, as shown in bold in the output. (In the following examples PE1 and PE4 are the T-PE routers.)

Example:

```
Router# show mpls l2transport binding

  Destination Address: 10.1.1.1,  VC ID: 102
    Local Label:  17
        Cbit: 1,    VC Type: Ethernet,    GroupID: 0
        MTU: 1500,   Interface Desc: n/a
        VCCV: CC Type: CW [1], RA [2], TTL [3]
              CV Type: LSPV [2]
    Remote Label: 16
        Cbit: 1,    VC Type: Ethernet,    GroupID: 0
        MTU: 1500,   Interface Desc: n/a
        VCCV: CC Type: CW [1], RA [2], TTL [3]
              CV Type: LSPV [2]
        PW Switching Point:
             Vcid   local IP addr      remote IP addr      Description
             101        10.11.11.11    10.20.20.20         PW Switching Point PE3
             100        10.20.20.20    10.11.11.11            PW Switching Point PE2
```

Step 2 **show mpls l2transport vc detail**

Use the **show mpls l2transport vc detail** command to display status of the pseudowire switching point. In the following example, the output (shown in bold) displays the segment that is the source of the fault of the multisegment pseudowire:

Example:

```
Router# show mpls l2transport vc detail
Local interface: Se3/0 up, line protocol up, HDLC up
  Destination address: 12.1.1.1, VC ID: 100, VC status: down
    Output interface: Se2/0, imposed label stack {23}
    Preferred path: not configured
    Default path: active
    Next hop: point2point
  Create time: 00:03:02, last status change time: 00:01:41
  Signaling protocol: LDP, peer 10.1.1.1:0 up
    Targeted Hello: 10.1.1.4(LDP Id) -> 10.1.1.1, LDP is UP
    Status TLV support (local/remote)   : enabled/supported
      LDP route watch                   : enabled
      Label/status state machine        : established, LruRrd
      Last local dataplane   status rcvd: No fault
      Last local SSS circuit status rcvd: No fault
      Last local SSS circuit status sent: DOWN(PW-tx-fault)
      Last local  LDP TLV    status sent: No fault
      Last remote LDP TLV    status rcvd: DOWN(PW-tx-fault)
       PW Switching Point:
       Fault type  Vcid   local IP addr   remote IP addr   Description
       PW-tx-fault  101   10.1.1.1        10.1.1.1         S-PE2
      Last remote LDP ADJ    status rcvd: No fault
    MPLS VC labels: local 19, remote 23
    Group ID: local 0, remote 0
    MTU: local 1500, remote 1500
    Remote interface description:
  Sequencing: receive disabled, send disabled
  VC statistics:
    packet totals: receive 16, send 27
    byte totals:   receive 2506, send 3098
    packet drops:  receive 0, seq error 0, send 0
```

Verifying Multisegment Pseudowires with ping mpls and trace mpls Commands

You can use **ping mpls** and **trace mpls** commands to verify connectivity in multisegment pseudowires.

Note

Some **ping mpls** and **trace mpls**keywords that are available with IPv4 LDP or traffic engineering (TE) are not available with pseudowire.

The following keywords are not available with the **ping mpls pseudowire** command:

- **dsmap**
- **flags**
- **force-explicit-null**
- **output**
- **revision**
- **ttl**

The following keywords are not available with the **trace mpls pseudowire** command:

- **flags**
- **force-explicit-null**
- **output**
- **revision**
- **ttl**

>

SUMMARY STEPS

1. **ping mpls pseudowire** *destination-address vc-id* [**segment** *segment-number*]
2. **trace mpls pseudowire** *destination-address vc-id* **segment** *segment-number* [*segment-number]*

DETAILED STEPS

Step 1

ping mpls pseudowire *destination-address vc-id* [**segment** *segment-number*]
Where:

- *destination-address* is the address of the S-PE router, which is the end of the segment from the direction of the source.
- *vc-id* is the VC ID of the segment from the source to the next PE router.
- **segment** *segment-number is optional and specifies the segment you want to ping.*

The following examples use the topology shown in the second figure above:

- To perform an end-to-end ping operation from T-PE1 to T-PE2, enter the following command. *destination-address* is S-PE1 and *vc-id*is the VC between T-PE1 and S-PE1.

ping mpls pseudowire *destination-address vc-id*

- To perform a ping operation from T-PE1 to segment 2, enter the following command. destination-*address* is S-PE1 and *vc-id*is the VC between T-PE1 and S-PE1.

ping mpls pseudowire *destination-address vc-id* **segment 2**

Example:

Step 2

trace mpls pseudowire *destination-address vc-id* **segment** *segment-number* [*segment-number]*
Where:

- *destination-address is the address of the next S-PE router from the origin of the trace.*
- *vc-id* is the VC ID of the segment from which the **trace** command is issued.
- *segment-number* indicates the segment upon which the trace operation will act. If you enter two segment numbers, the traceroute operation will perform a trace on that range of routers.

The following examples use the topology shown in the second figure above:

- To perform a trace operation from T-PE1 to segment 2 of the multisegment pseudowire, enter the following command. *destination-address* is S-PE1 and *vc-id*is the VC between T-PE1 and S-PE1.

trace mpls pseudowire *destination-address vc-id* **segment 2**

This example performs a trace from T-PE1 to S-PE2.

- To perform a trace operation on a range of segments, enter the following command. This example performs a trace from S-PE2 to T-PE2. *destination-address* is S-PE1 and *vc-id* is the VC between T-PE1 and S-PE1.

trace mpls pseudowire *destination-address vc-id* **segment 2 4**

The following commands perform trace operations on S-PE router 10.10.10.9, first on segment 1, then on segment 2.

Segment 1 trace:

Example:

```
Router# trace mpls pseudowire 10.10.10.9 220 segment 1
Tracing MS-PW segments within range [1-1] peer address 10.10.10.9 and timeout 2 seconds
Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
  'L' - labeled output interface, 'B' - unlabeled output interface,
  'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
  'M' - malformed request, 'm' - unsupported tlvs, 'N' - no label entry,
  'P' - no rx intf label prot, 'p' - premature termination of LSP,
  'R' - transit router, 'I' - unknown upstream index,
  'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
L 1 10.10.9.9 0 ms [Labels: 18 Exp: 0]
    local 10.10.10.22 remote 10.10.10.9 vc id 220
Segment 2 trace:
Router# trace mpls pseudowire 10.10.10.9 220 segment 2
Tracing MS-PW segments within range [1-2] peer address 10.10.10.9 and timeout 2 seconds
Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
  'L' - labeled output interface, 'B' - unlabeled output interface,
  'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
  'M' - malformed request, 'm' - unsupported tlvs, 'N' - no label entry,
  'P' - no rx intf label prot, 'p' - premature termination of LSP,
  'R' - transit router, 'I' - unknown upstream index,
  'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
L 1 10.10.9.9 4 ms [Labels: 18 Exp: 0]
    local 10.10.10.22 remote 10.10.10.9 vc id 220
! 2 10.10.3.3 4 ms [Labels: 16 Exp: 0]
    local 10.10.10.9 remote 10.10.10.3 vc id 220
```

Verifying L2VPN VPLS Inter-AS Option B with ping mpls and trace mpls Commands

You can use **ping mpls**and **trace mpls** commands to verify connectivity in configurations using the L2VPN VPLS Inter-AS Option B feature. For end-to-end ping and trace operations, you enter the destination address of the T-PE router at the other end of the pseudowire.

Note Some **ping mpls**and **trace mpls**keywords that are available with IPv4 LDP or traffic engineering (TE) are not available with pseudowire.

The following keywords are not available with the **ping mpls pseudowire** command:

- **dsmap**
- **flags**
- **force-explicit-null**
- **output**
- **revision**
- **ttl**

The following keywords are not available with the **trace mpls pseudowire** command:

- **flags**
- **force-explicit-null**
- **output**
- **revision**
- **ttl**

>

SUMMARY STEPS

1. **ping mpls pseudowire** *destination-address vc-id* [**segment** *segment-number*]
2. **trace mpls pseudowire** *destination-address vc-id* **segment** *segment-number* [*segment-number]*

DETAILED STEPS

Step 1 **ping mpls pseudowire** *destination-address vc-id* [**segment** *segment-number*]
Where:

- *destination-address* is the address of the T-PE2 router at the other end of the pseudowire.
- *vc-id* is the VC ID between T-PE1 and S-PE1.
- **segment** *segment-number is optional and specifies the segment you want to ping.*

The following examples use the topology shown in the second figure above:

- To perform an end-to-end ping operation from T-PE1 to T-PE2, enter the following command. destination-*address* is T-PE2 and *vc-id*is the VC between T-PE1 and S-PE1.

ping mpls pseudowire *destination-address vc-id*

Example:

Step 2 **trace mpls pseudowire** *destination-address vc-id* **segment** *segment-number* [*segment-number]*
Where:

- *destination-address* is the address of the T-PE2 router at the other end of the pseudowire.
- *vc-id* is the VC ID between T-PE1 and S-PE1.

- *segment-number* indicates the segment upon which the trace operation will act. If you enter two segment numbers, the traceroute operation will perform a trace on that range of routers.

The following examples use the topology shown in the second figure above:

- To perform a trace operation from T-PE1 to T-PE2, enter the following command. *destination-address*is T-PE2 and *vc-id*is the VC between T-PE1 and S-PE1.

trace mpls pseudowire *destination-address vc-id* **segment 2**

This example performs a trace from T-PE1 to T-PE2.

- To perform a trace operation on a range of segments, enter the following command. This example performs a trace from S-PE2 to T-PE2. *destination-address*is S-PE1 and *vc-id* is the VC between T-PE1 and S-PE1.

trace mpls pseudowire *destination-address vc-id* **segment 2 4**

Configuration Examples for L2VPN Multisegment Pseudowires

- Example Configuring an L2VPN Multisegment Pseudowire, page 238

Example Configuring an L2VPN Multisegment Pseudowire

The following example does not include all the commands. Unconfigured interfaces are not shown. Portions of the example relevant to L2VPN Multisegment Pseudowires are shown in bold.

T-PE1 Configuration

```
no ipv6 cef
multilink bundle-name authenticated
frame-relay switching
mpls traffic-eng tunnels
mpls ldp discovery targeted-hello accept
no mpls ip propagate-ttl forwarded
mpls label protocol ldp
!
policy-map exp2
!
interface Loopback0
 ip address 10.131.191.252 255.255.255.255
 no clns route-cache
!
interface Ethernet0/0
 ip address 10.131.191.230 255.255.255.252
 mpls label protocol ldp
 mpls ip
 no clns route-cache
 ip rsvp bandwidth 1500 1500
 ip rsvp signalling dscp 0
!
interface Ethernet1/0
 ip address 10.131.159.246 255.255.255.252
shutdown
 no clns route-cache
```

```
!
interface Ethernet2/0
 no ip address
 no cdp enable
!
interface Ethernet2/0.1
 encapsulation dot1Q 1000
 xconnect 10.131.191.251 333 encapsulation mpls
!
router ospf 1
 log-adjacency-changes
 passive-interface Loopback0
 network 10.131.159.244 0.0.0.3 area 0
 network 10.131.191.228 0.0.0.3 area 0
 network 10.131.191.232 0.0.0.3 area 0
 network 10.131.191.252 0.0.0.0 area 0
 network 11.0.0.0 0.0.0.3 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
ip classless
!
no ip http server
!
mpls ldp router-id Loopback0 force
end
```

S-PE1 Configuration

```
no ipv6 cef
multilink bundle-name authenticated
mpls traffic-eng tunnels
no mpls traffic-eng auto-bw timers
mpls ldp discovery targeted-hello accept
no mpls ip propagate-ttl forwarded
mpls label protocol ldp
!
policy-map exp2
!
l2 vfi sam-sp point-to-point
 neighbor 10.131.191.252 333 encapsulation mpls
 neighbor 10.131.159.251 222 encapsulation mpls
!
interface Tunnel3
 ip unnumbered Loopback0
 shutdown
 mpls label protocol ldp
 mpls accounting experimental input
 mpls ip
 tunnel mode mpls traffic-eng
 tunnel destination 10.131.159.252
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 2 2
 tunnel mpls traffic-eng bandwidth 512
 tunnel mpls traffic-eng path-option 1 dynamic
 no clns route-cache
 service-policy output exp2
!
interface Loopback0
 ip address 10.131.191.251 255.255.255.255
 no clns route-cache
!
interface Ethernet0/0
 ip address 10.131.191.229 255.255.255.252
 mpls traffic-eng tunnels
 mpls label protocol ldp
 mpls ip
 no clns route-cache
 ip rsvp bandwidth 1500 1500
 ip rsvp signalling dscp 0
!
interface Ethernet1/0
```

```
 ip address 10.131.159.226 255.255.255.252
 mpls traffic-eng tunnels
 mpls ip
 no clns route-cache
service-policy output exp2
 ip rsvp bandwidth 1500 1500
 ip rsvp signalling dscp 0
!
interface Serial2/0
 ip unnumbered Loopback0
 mpls ip
 no fair-queue
 no keepalive
 serial restart-delay 0
 no clns route-cache
!
router ospf 1
 log-adjacency-changes
 passive-interface Loopback0
 network 10.131.159.224 0.0.0.3 area 0
 network 10.131.191.228 0.0.0.3 area 0
 network 10.131.191.251 0.0.0.0 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
ip classless
!
end
```

T-PE2 Configuration

```
no ipv6 cef
no l2tp congestion-control
multilink bundle-name authenticated
frame-relay switching
mpls traffic-eng tunnels
no mpls traffic-eng auto-bw timers frequency 0
mpls ldp discovery targeted-hello accept
no mpls ip propagate-ttl forwarded
mpls label protocol ldp
!
interface Loopback0
 ip address 10.131.159.252 255.255.255.255
 no clns route-cache
!
interface Ethernet0/0
 ip address 10.131.159.230 255.255.255.252
interface Ethernet0/0
 ip address 10.131.159.230 255.255.255.252
 mpls traffic-eng tunnels
 mpls ip
 no clns route-cache
 ip rsvp bandwidth 1500 1500
 ip rsvp signalling dscp 0
!
interface Ethernet1/0
 ip address 10.131.159.245 255.255.255.252
 shutdown
 mpls ip
 no clns route-cache
!
interface Ethernet3/0.1
 encapsulation dot1Q 1000
 xconnect 10.131.159.251 111 encapsulation mpls
!
router ospf 1
 log-adjacency-changes
 passive-interface Loopback0
 network 10.131.122.0 0.0.0.3 area 0
 network 10.131.159.228 0.0.0.3 area 0
 network 10.131.159.232 0.0.0.3 area 0
 network 10.131.159.244 0.0.0.3 area 0
 network 10.131.159.252 0.0.0.0 area 0
 network 11.0.0.0 0.0.0.3 area 0
 network 19.0.0.0 0.0.0.255 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
end
```

S-PE2 configuration

```
no ipv6 cef
no l2tp congestion-control
multilink bundle-name authenticated
mpls traffic-eng tunnels
no mpls traffic-eng auto-bw timers frequency 0
mpls ldp discovery targeted-hello accept
no mpls ip propagate-ttl forwarded
mpls label protocol ldp
!
l2 vfi sam-sp point-to-point
 neighbor 10.131.159.252 111 encapsulation mpls
 neighbor 10.131.191.251 222 encapsulation mpls
!
!
interface Loopback0
 ip address 10.131.159.251 255.255.255.255
!
interface Ethernet0/0
interface Ethernet0/0
 ip address 10.131.159.229 255.255.255.252
 mpls traffic-eng tunnels
 mpls accounting experimental input
 mpls ip
 ip rsvp bandwidth 1500 1500
 ip rsvp signalling dscp 0
!
interface Ethernet1/0
 ip address 10.131.159.225 255.255.255.252
 mpls traffic-eng tunnels
 mpls ip
 ip rsvp bandwidth 1500 1500
 ip rsvp signalling dscp 0
!
router ospf 1
 log-adjacency-changes
 passive-interface Loopback0
 network 10.131.159.224 0.0.0.3 area 0
 network 10.131.159.228 0.0.0.3 area 0
 network 10.131.159.251 0.0.0.0 area 0
 network 19.0.0.0 0.0.0.255 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
end
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases

Related Topic	Document Title
MPLS commands	*Cisco IOS Multiprotocol Label Switching Command Reference*
Layer 2 VPNS	• Any Transport over MPLS • L2VPN Pseudowire Switching • MPLS LSP Ping/Traceroute for LDP/TE, and LSP Ping for VCCV
L2VPN VPLS Inter-AS Option B	L2VPN VPLS Inter-AS Option B

Standards

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

MIBs

MIB	MIBs Link
No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
RFC 4379	http://tools.ietf.org/html/rfc4379 Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures
RFC 4447	Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP)
RFC 5085	Pseudowire Virtual Circuit Connectivity Verification (VCCV)

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for L2VPN Multisegment Pseudowires

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 25 Feature Information for L2VPN Multisegment Pseudowires

Feature Name	Releases	Feature Information
L2VPN Multisegment Pseudowires	12.2(33)SRE	This feature enables you to configure two or more Layer 2 pseudowire segments that function as a single pseudowire. The feature spans multiple cores or autonomous systems of the same or different carrier networks.
MPLS OAM Support for Multisegment Pseudowires	12.2(33)SRE	This feature enables you to use the **ping mpls** and **trace mpls**commands to verify that all the segments of the MPLS multisegment pseudowire are operating.

Feature Name	Releases	Feature Information
MPLS OAM Support for L2VPN VPLS Inter-AS Option B	15.1(1)S	This feature is an enhancement to the MPLS OAM Support for Multisegment Pseudowires feature. This feature allows you to use the **ping mpls** and **trace mpls**commands to verify the pseudowire used in a L2VPN VPLS Inter-AS Option B configuration.

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.

QoS Policy Support for L2VPN ATM PVPs

This document explains how to configure Quality of Service (QoS) Policy Support for Layer 2 Virtual Private Network (L2VPN) ATM permanent virtual paths (PVPs). That is, it explains how to configure QoS policies in ATM PVP mode for L2VPNs.

- Finding Feature Information, page 245
- Prerequisites for QoS Policy Support for L2VPN ATM PVPs, page 245
- Restrictions for QoS Policy Support for L2VPN ATM PVPs, page 245
- Information About QoS Policy Support for L2VPN ATM PVPs, page 246
- How to Configure QoS Policy Support for L2VPN ATM PVPs, page 247
- Configuration Examples for QoS Policy Support for L2VPN ATM PVPs, page 252
- Additional References, page 252
- Feature Information for QoS Policy Support for L2VPN ATM PVPs, page 254

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for QoS Policy Support for L2VPN ATM PVPs

Before configuring QoS policies on L2VPN ATM PVPs, you should understand the concepts and configuration instructions in the following document:

- Any Transport over MPLS

Restrictions for QoS Policy Support for L2VPN ATM PVPs

The following restrictions apply to the QoS Policy Support for L2VPN ATM PVPs feature:

- The Cisco 7600 series router does not support any queueing features in ATM PVP mode.
- When you enable a policy in PVP mode, do not configure ATM rates on the VCs that are part of the PVP. The VCs should be unspecified bit rate (UBR) VCs only.

- If VCs are part of a PVP that has a policy configured, you cannot configure ATM VC traffic shaping.
- Cisco IOS Release 12.2(33)SRE does not support cell-based ATM shaping per PVP.
- You cannot configure a queueing policy on an ATM PVP with UBR.
- You cannot configure queueing-based policies with UBR traffic shaping.

Information About QoS Policy Support for L2VPN ATM PVPs

- MQC Structure, page 246
- Elements of a Traffic Class, page 246
- Elements of a Traffic Policy, page 246

MQC Structure

The modular QoS command-line interface (CLI) (MQC) structure allows you to define a traffic class, create a traffic policy, and attach the traffic policy to an interface.

The MQC structure is the result of the following these three high-level steps.

1. Define a traffic class by using the **class-map**command. A traffic class is used to classify traffic.
2. Create a traffic policy by using the **policy-map** command. (The terms traffic policy and policy map are often synonymous.) A traffic policy (policy map) contains a traffic class and one or more QoS features that will be applied to the traffic class. The QoS features in the traffic policy determine how to treat the classified traffic.
3. Attach the traffic policy (policy map) to the interface by using the **service-policy** command.

Elements of a Traffic Class

A traffic class contains three major elements: a traffic class name, a series of match commands, and, if more than one match command is used in the traffic class, instructions on how to evaluate these match commands.

The match commands are used for classifying packets. Packets are checked to determine whether they meet the criteria specified in the match commands; if a packet meets the specified criteria, that packet is considered a member of the class. Packets that fail to meet the matching criteria are classified as members of the default traffic class.

Elements of a Traffic Policy

A traffic policy contains three elements: a traffic policy name, a traffic class (specified with the **class** command), and the command used to enable the QoS feature.

The traffic policy (policy map) applies the enabled QoS feature to the traffic class once you attach the policy map to the interface (by using the **service-policy** command).

Note A packet can match only one traffic class within a traffic policy. If a packet matches more than one traffic class in the traffic policy, the first traffic class defined in the policy will be used.

How to Configure QoS Policy Support for L2VPN ATM PVPs

- Enabling a Service Policy in ATM PVP Mode, page 247
- Enabling Traffic Shaping in ATM PVP Mode, page 248
- Enabling Matching of ATM VCIs, page 251

Enabling a Service Policy in ATM PVP Mode

You can enable a service policy in ATM PVP mode. You can also enable a service policy on PVP on a multipoint subinterface.

Note

- The Cisco 7600 series router does not support a service policy that uses the **match atm-vci**command in the egress direction.
- The **show policy-map interface** command does not display service policy information for ATM interfaces.
- Cisco IOS Release 12.2(33)SRE does not support cell-based ATM shaping per PVP.

>

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. interface atm slot/port
4. **atm pvp** *vpi* **l2transport**
5. **service-policy** [**input** | **output**] *policy-map-name*
6. xconnect peer-router-id vcid encapsulation mpls
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	interface atm slot/port **Example:** `Router(config)# interface atm 1/0`	Defines the interface and enters interface configuration mode.
Step 4	**atm pvp** *vpi* **l2transport** **Example:** `Router(config-if)# atm pvp 1 l2transport`	Specifies that the PVP is dedicated to transporting ATM cells and enters l2transport PVP configuration mode. • The l2transport keyword indicates that the PVP is for cell relay. This mode is for Layer 2 transport only; it is not for regular PVPs.
Step 5	**service-policy** [**input** | **output**] *policy-map-name* **Example:** `Router(config-if-atm-l2trans-pvp)# service policy input pol1`	Enables a service policy on the specified PVP.
Step 6	xconnect peer-router-id vcid encapsulation mpls **Example:** `Router(config-if-atm-l2trans-pvp)# xconnect 10.0.0.1 123 encapsulation mpls`	Binds the attachment circuit to a pseudowire VC. • The syntax for this command is the same as for all other Layer 2 transports.

Step 7	**end**
	Example:
	`Router(config-if-atm-l2trans-pvp)#`
	`end`
		Exits l2transport PVP configuration mode and returns to privileged EXEC mode.

Enabling Traffic Shaping in ATM PVP Mode

Traffic shaping commands are supported in ATM PVP mode. For egress VP shaping, one configuration command is supported for each ATM service category. The supported service categories are constant bit rate (CBR), UBR, variable bit rate-nonreal time (VBR-NRT), and variable bit rate real-time (VBR-RT).

Note

- The Cisco 7600 series router does not support traffic shaping.
- Cisco IOS Release 12.2(33)SRE does not support cell-based ATM shaping per PVP.

>

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. interface atm slot/port
4. **atm pvp** *vpi* **l2transport**
5. Do one of the following:
   - **ubr** *pcr*
   -
   - **cbr** *pcr*
   - or
   - **vbr-nrt** *pcr scr mbs*
   - or
   - **vbr-rt** *pcr scr mbs*
6. xconnect peer-router-id vcid encapsulation mpls

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	interface atm slot/port **Example:** `Router(config)# interface atm 1/0`	Defines the interface and enters interface configuration mode.
Step 4	**atm pvp** *vpi* **l2transport** **Example:** `Router(config-if)# atm pvp 1 l2transport`	Specifies that the PVP is dedicated to transporting ATM cells, and enters l2transport PVP configuration mode. • The l2transport keyword indicates that the PVP is for cell relay. This mode is for Layer 2 transport only; it is not for regular PVPs.

	Command or Action	Purpose

Step 5	Do one of the following:
	• **ubr** *pcr*
	•
	• **cbr** *pcr*
	• or
	• **vbr-nrt** *pcr scr mbs*
	• or
	• **vbr-rt** *pcr scr mbs*
	Example:
	`Router(config-if-atm-l2trans-pvp)# cbr 1000`
	Example:
	Example:
	`cbr 56`
	Example:
	Example:
	`vbr-nrt 11760 11760 1`
	Example:
	Example:
	`vbr-rt 640 320 80`
		Enables traffic shaping in ATM PVP mode.
		• *pcr* = peak cell rate
		• *scr* = sustain cell rate
		• *mbs* = maximum burst size

	Command or Action	Purpose
Step 6	xconnect peer-router-id vcid encapsulation mpls **Example:** `Router(config-if-atm-l2trans-pvp)# xconnect 10.0.0.1 123 encapsulation mpls`	Binds the attachment circuit to a pseudowire VC. • The syntax for this command is the same as for all other Layer 2 transports.

Enabling Matching of ATM VCIs

You can enable packet matching on an ATM VCI or range of VCIs using the **match atm-vci** command in class map configuration mode.

Note

- When you configure the **match atm-vci**command in class map configuration mode, you can add this class map to a policy map that can be attached only to an ATM VP.
- On the Cisco 7600 series router, the **match atm-vci**command is supported only in the ingress direction on an ATM VP.
- Cisco IOS Release 12.2(33)SRE does not support cell-based ATM shaping per PVP.

>

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map** *class-map-name* [**match-all** | **match-any**]
4. **match atm-vci** *vc-id* **[- vc-id]**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose

Step 3	**class-map** *class-map-name* [**match-all** | **match-any**]
	Example:
	`Router(config)# class-map class1`
		Creates a class map to be used for matching traffic to a specified class, and enters class map configuration mode.

Step 4	**match atm-vci** *vc-id* **[- vc-id]** **Example:** `Router(config-cmap)# match atm-vci 50`	Enables packet matching on an ATM VCI or range of VCIs. • The range is 32 to 65535. **Note** You can use the **match not** command to match any VC except those you specify in the command.

Step 5	**end**
	Example:
	`Router(config-cmap)# end`
		(Optional) Returns to privileged EXEC mode.

Configuration Examples for QoS Policy Support for L2VPN ATM PVPs

- Enabling Traffic Shaping in ATM PVP Mode Example, page 252

Enabling Traffic Shaping in ATM PVP Mode Example

The following example enables traffic shaping in ATM PMP mode.

```
interface atm 1/0
 atm pvp 100 l2transport
  ubr 1000
  xconnect 10.11.11.11 777 encapsulation mpls
 atm pvp 101 l2transport
  cbr 1000
  xconnect 10.11.11.11 888 encapsulation mpls
 atm pvp 102 l2transport
  vbr-nrt 1200 800 128
  xconnect 10.11.11.11 999 encapsulation mpls
```

Additional References

The following sections provide references related to the QoS Policy Support for L2VPN ATM PVPs feature.

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
MPLS commands	*Cisco IOS Multiprotocol Label Switching Command Reference*
Any Transport over MPLS	Any Transport over MPLS

Standards

Standard	Title
None	—

MIBs

MIB	MIBs Link
• None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
None	—

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/public/support/tac/home.shtml http://www.cisco.com/cisco/web/support/index.html

Feature Information for QoS Policy Support for L2VPN ATM PVPs

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 26 Feature Information for QoS Policy Support for L2VPN ATM PVPs

Feature Name	Releases	Feature Information
QoS Policy Support for L2VPN ATM PVPs	12.2(33)SRE	This feature enables you to configure QoS policies in ATM PVP mode for L2VPNs. The following commands were introduced or modified by this feature: **cbr**, **match atm-vci**, **service-policy**, **ubr, vbr-nrt**, **vbr-rt**.

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.

L2VPN Pseudowire Preferential Forwarding

The L2VPN: Pseudowire Preferential Forwarding feature allows you to configure the pseudowires so that you can use ping and show commands to find status information of the pseudowires before, during, and after a switchover.

- Finding Feature Information, page 255
- Prerequisites for L2VPN--Pseudowire Preferential Forwarding, page 255
- Restrictions for L2VPN--Pseudowire Preferential Forwarding, page 256
- Information About L2VPN--Pseudowire Preferential Forwarding, page 256
- How to Configure L2VPN--Pseudowire Preferential Forwarding, page 257
- Configuration Examples for L2VPN--Pseudowire Preferential Forwarding, page 258
- Additional References, page 260
- Feature Information for L2VPN: Pseudowire Preferential Forwarding, page 262

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for L2VPN--Pseudowire Preferential Forwarding

- Before configuring the L2VPN: Pseudowire Preferential Forwarding feature, you should understand the concepts in the following documents:
    - Preferential Forwarding Status Bit Definition (draft-ietf-pwe3-redundancy-bit-xx.txt)
    - MPLS Pseudowire Status Signaling
    - L2VPN Pseudowire Redundancy
    - NSF/SSO--Any Transport over MPLS and AToM Graceful Restart
    - MPLS LSP Ping/Traceroute for LDP/TE, and LSP Ping for VCCV
- The PE routers must be configured with the following features:

  - L2VPN Pseudowire Redundancy
  - NSF/SSO--Any Transport over MPLS and AToM Graceful Restart
- The L2VPN: Pseudowire Preferential Forwarding feature requires that the following mechanisms be in place to enable you to detect a failure in the network:
  - Label switched paths (LSP) Ping/Traceroute and Any Transport over MPLS Virtual Circuit Connection Verification (AToM VCCV)
  - Local Management Interface (LMI)
  - Operation, Administration, and Maintenance (OAM)

Restrictions for L2VPN--Pseudowire Preferential Forwarding

- Only ATM attachment circuits are supported.
- The following features are not supported:
  - Port mode cell relay
  - Any Transport over MPLS: AAL5 over MPLS
  - VC cell packing
  - OAM emulation
  - ILMI/PVC-D
  - Permanent virtual circuit (PVC) Range
  - L2TPv3 Pseudowire Redundancy
  - Local switching
  - Multiple backup pseudowires
  - Static pseudowires

Information About L2VPN--Pseudowire Preferential Forwarding

- Overview of L2VPN--Pseudowire Preferential Forwarding, page 256

Overview of L2VPN--Pseudowire Preferential Forwarding

The L2VPN: Pseudowire Preferential Forwarding feature allows you to configure pseudowires so that you can use ping, traceroute, and show commands to find status information before, during, and after a switchover. The implementation of this feature is based on *Preferential Forwarding Status Bit Definition* (draft-ietf-pwe3-redundancy-bit-xx.txt). The L2VPN: Pseudowire Preferential Forwarding feature provides these enhancements for displaying information about the pseudowires:

- You can issue **ping mpls**commands on the backup pseudowires.
- You can display status of the pseudowires before, during, and after a switchover, using the **show xconnect** and s**how mpls l2transport vc**commands.

Note In a single-segment pseudowire, the PE routers at each end of the pseudowire serve as the termination points. In multisegment pseudowires, the terminating PE routers serve as the termination points.

How to Configure L2VPN--Pseudowire Preferential Forwarding

- Configuring the Pseudowire Connection Between PE Routers, page 257

Configuring the Pseudowire Connection Between PE Routers

You set up a connection, called a pseudowire, between the routers to transmit Layer 2 frames between PE routers.

As part of the pseudowire configuration, issue the **status redundancy master**command to make it the master. This enables the L2VPN: Pseudowire Preferential Forwarding feature to display the status of the active and backup pseudowires. By default, the PE router is in slave mode.

Note One pseudowire must be the master and the other must be assigned the slave. You cannot configure both pseudowires as master or slave.

Note You must specify the encapsulation mpls command as part of the pseudowire class for the AToM VCs to work properly. If you omit the encapsulation mpls command, you receive the following error: % Incomplete command.

The PE routers must be configured for the L2VPN Pseudowire Redundancy and NSF/SSO--Any Transport over MPLS and AToM Graceful Restart features. See the following documents for configuration instructions.

- L2VPN Pseudowire Redundancy
- NSF/SSO--Any Transport over MPLS and AToM Graceful Restart

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. pseudowire-class name
4. **encapsulation mpls**
5. **status redundancy** {**master**| **slave**}
6. **interworking** {**ethernet** | **ip**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	pseudowire-class name **Example:** `Router(config)# pseudowire-class atom`	Establishes a pseudowire class with a name that you specify, and enters pseudowire class configuration mode.
Step 4	**encapsulation mpls** **Example:** `Router(config-pw)# encapsulation mpls`	Specifies the tunneling encapsulation. • For AToM, the encapsulation type is mpls.

Step 5	**status redundancy** {**master**| **slave**}
	Example:
	`Router(config-pw)# status redundancy master`
		Specifies the pseudowire as the master or slave. This enables the L2VPN: Pseudowire Preferential Forwarding feature to display the status of the active and backup pseudowires.
		• By default, the PE router is in slave mode.
		Note One pseudowire must be the master and the other must be assigned the slave. You cannot configure both pseudowires as master or slave.

Step 6	**interworking** {**ethernet** | **ip**}
	Example:
	`Router(config-pw)# interworking ip`
		(Optional) Enables the translation between the different Layer 2 encapsulations.

Configuration Examples for L2VPN--Pseudowire Preferential Forwarding

- L2VPN--Pseudowire Preferential Forwarding Configuration Example, page 259
- Displaying the Status of the Pseudowires Example, page 259

L2VPN--Pseudowire Preferential Forwarding Configuration Example

The following commands configure a PE router with the L2VPN: Pseudowire Preferential Forwarding feature:

```
mpls ldp graceful-restart
mpls ip
mpls label protocol ldp
mpls ldp router-id Loopback0 force
mpls ldp advertise-labels
!
pseudowire-class mpls
 encapsulation mpls
 status redundancy master
interface ATM0/2/0.1 multipoint
 logging event subif-link-status
 atm pvp 50 l2transport
  xconnect 10.1.1.2 100 encap mpls
   backup peer 10.1.1.3 100 encap mpls
end
```

Displaying the Status of the Pseudowires Example

The following examples show the status of the active and backup pseudowires before, during, and after a switchover.

The **show mpls l2transport vc** command on the active PE router displays the status of the pseudowires:

```
Router# show mpls l2transport vc
Local intf     Local circuit              Dest address    VC ID      Status
-------------  -------------------------- --------------- ---------- ----------
AT0/2/0/0.1    ATM VPC CELL 50            10.1.1.2         100        UP
AT0/2/0/0.1    ATM VPC CELL 50            10.1.1.3         100        STANDBY
```

The **show mpls l2transport vc** command on the backup PE router displays the status of the pseudowires. The active pseudowire on the backup PE router has the HOTSTANDBY status.

```
Router1-standby# show mpls l2transport vc

Local intf     Local circuit              Dest address    VC ID      Status
-------------  -------------------------- --------------- ---------- ----------
AT0/2/0/0.1    ATM VPC CELL 50            10.1.1.2         100        HOTSTANDBY
AT0/2/0/0.1    ATM VPC CELL 50            10.1.1.3         100        DOWN
```

During a switchover, the status of the active and backup pseudowires changes:

```
Router# show mpls l2transport vc
Local intf     Local circuit              Dest address    VC ID      Status
-------------  -------------------------- --------------- ---------- ----------
AT0/2/0/0.1    ATM VPC CELL 50            10.1.1.2         100        RECOVERING
AT0/2/0/0.1    ATM VPC CELL 50            10.1.1.3         100        DOWN
```

After the switchover is complete, the recovering pseudowire shows a status of UP:

```
Router# show mpls l2transport vc
Local intf     Local circuit              Dest address    VC ID      Status
-------------  -------------------------- --------------- ---------- ----------
AT0/2/0/0.1    ATM VPC CELL 50            10.1.1.2         100        UP
AT0/2/0/0.1    ATM VPC CELL 50            10.1.1.3         100        STANDBY
```

The **show xconnect** command displays the standby (SB) state for the backup pseudowire, which is independent of the stateful switchover mode of the router:

```
Router# show xconnect all
Legend:    XC ST=Xconnect State  S1=Segment1 State  S2=Segment2 State
  UP=Up       DN=Down            AD=Admin Down      IA=Inactive
  SB=Standby  HS=Hot Standby     RV=Recovering      NH=No Hardware
XC ST        Segment 1                                  S1 Segment
2                                 S2
------+---------------------------------+--+---------------------------------+---------
UP pri ac   AT1/1/0/0.1/1/1:220/220(ATM V  UP mpls 10.193.193.3:330            UP
IA sec ac   AT1/1/0/0.1/1/1:220/220(ATM V  UP mpls 10.193.193.3:331            SB
```

The **ping mpls** and **traceroute mpls** commands show that the dataplane is active on the backup pseudowire:

```
Router# ping mpls pseudowire 10.193.193.22 331
%Total number of MS-PW segments is less than segment number; Adjusting the segment number
to 1
Sending 5, 100-byte MPLS Echos to 10.193.193.22,
     timeout is 2 seconds, send interval is 0 msec:
Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
  'L' - labeled output interface, 'B' - unlabeled output interface,
  'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
  'M' - malformed request, 'm' - unsupported tlvs, 'N' - no label entry,
  'P' - no rx intf label prot, 'p' - premature termination of LSP,
  'R' - transit router, 'I' - unknown upstream index,
  'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms

Router# traceroute mpls pseudowire 10.193.193.22 331 segment 1
Tracing MS-PW segments within range [1-1] peer address 10.193.193.22 and timeout 2 seconds
Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
  'L' - labeled output interface, 'B' - unlabeled output interface,
  'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
  'M' - malformed request, 'm' - unsupported tlvs, 'N' - no label entry,
  'P' - no rx intf label prot, 'p' - premature termination of LSP,
  'R' - transit router, 'I' - unknown upstream index,
  'X' - unknown return code, 'x' - return code 0
Type escape sequence to abort.
! 1 10.193.33.22 4 ms [Labels: 23 Exp: 0]
    local 10.193.193.3 remote 10.193.193.22 vc id 331
```

Additional References

The following sections provide references related to the L2VPN: Pseudowire Preferential Forwarding feature.

Related Documents

Related Topic	Document Title
Description of commands associated with MPLS and MPLS applications	*Cisco IOS Multiprotocol Label Switching Command Reference*
L2VPN Pseudowires	• L2VPN Pseudowire Redundancy • MPLS Pseudowire Status Signaling

Related Topic	Document Title
NSF/SSO for L2VPNs	NSF/SSO--Any Transport over MPLS and AToM Graceful Restart
Ping and Traceroute for L2VPNs	MPLS LSP Ping/Traceroute for LDP/TE, and LSP Ping for VCCV

Standards

Standard	Title
draft-ietf-pwe3-redundancy-bit-xx.txt	Preferential Forwarding Status Bit Definition

MIBs

MIB	MIBs Link
No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco IOS XE software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

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

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

Feature Information for L2VPN: Pseudowire Preferential Forwarding

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 27 Feature Information for L2VPN: Pseudowire Preferential Forwarding

Feature Name	Releases	Feature Information
L2VPN: Pseudowire Preferential Forwarding	12.2(33)SRE	This feature allows you to configure the pseudowires so that you can use ping and show commands to find status information of the pseudowires before, during, and after a switchover. The following commands were introduced or modified: **show mpls l2transport vc**, **show xconnect**, **status redundancy**.

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.

MPLS Pseudowire Status Signaling

The MPLS Pseudowire Status Signaling feature enables you to configure the router so it can send pseudowire status to a peer router, even when the attachment circuit is down. In releases prior to Cisco IOS 12.2(33)SRC, if the attachment circuit was down, the pseudowire status messages were not sent to the peer.

- Finding Feature Information, page 263
- Prerequisites for MPLS Pseudowire Status Signaling, page 263
- Restrictions for MPLS Pseudowire Status Signaling, page 263
- Information About MPLS Pseudowire Status Signaling, page 264
- How to Configure MPLS Pseudowire Status Signaling, page 266
- Configuration Examples for MPLS Pseudowire Status Signaling, page 268
- Additional References, page 269

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for MPLS Pseudowire Status Signaling

- Before configuring this feature, make sure that both peer routers are capable of sending and receiving pseudowire status messages. Specifically, both routers should be running Cisco IOS Release 12.2(33)SRC and have the supported hardware installed.

Restrictions for MPLS Pseudowire Status Signaling

- Both peer routers must support the ability to send and receive pseudowire status messages in label advertisement and label notification messages. If both peer routers do not support pseudowire status messages, Cisco recommends that you disable the messages with the **no status** command.

- This feature is not integrated with Any Transport over MPLS (AToM) Virtual Circuit Connection Verification (VCCV).
- This feature is not integrated with Bidirectional Forwarding Detection (BFD).
- The standby and required switchover values from IETF draft-muley-pwe3-redundancy-02.txt are not supported.
- For a list of supported hardware for this feature, see the release notes for your platform.

Information About MPLS Pseudowire Status Signaling

- How MPLS Pseudowire Status Signaling Works, page 264

How MPLS Pseudowire Status Signaling Works

In releases prior to Cisco IOS Release 12.2(33)SRC, the control plane for AToM does not have the ability to provide pseudowire status. Therefore, when an attachment circuit (AC) associated with a pseudowire is down (or is forced down as part of the Pseudowire Redundancy functionality), labels advertised to peers are withdrawn. In Cisco IOS Release 12.2(33)SRC, the MPLS Pseudowire Status Signaling feature enables the AC status to be sent to the peer through the Label Distribution Protocol.

The pseudowire status messages are sent in label advertisement and label notification messages if the peer also supports the MPLS Pseudowire Status Signaling feature. You can issue the **show mpls l2transport vc detail** command to show that both the local and remote routers support pseudowire status messages. The following example shows the line of output to look for:

```
Router# show mpls l2transport vc detail
.
.
.
status TLV support (local/remote): enabled/supported
```

- When One Router Does Not Support MPLS Pseudowire Status Signaling, page 264
- Status Messages Indicating That the Attachment Circuit Is Down, page 265
- Message Codes in the Pseudowire Status Messages, page 265

When One Router Does Not Support MPLS Pseudowire Status Signaling

The peer routers must support the ability to send and receive pseudowire status messages in label advertisement and label notification messages. If one router does not support pseudowire status messages, Cisco recommends that you disable the messages with the **no status** command. This returns the router to label withdraw mode.

If the peer does not support the MPLS Pseudowire Status Signaling feature, the local router changes its mode of operation to label withdraw mode. You can issue the **show mpls l2transport vc detail** command to show that the remote router does not support pseudowire status messages. The following example shows the line of output to look for:

```
Router# show mpls l2transport vc detail

.

.

.

status TLV support (local/remote): enabled/not supported
```

When you issue the following **debug mpls l2transport vc**commands, the messages show that the peer router does not supportthe MPLS Pseudowire Status Signaling feature and that the local router is changing to withdraw mode, as shown in bold in the following example:

Router# debug mpls l2transport vc event Router# **debug mpls l2transport vc status event** Router# **debug mpls l2transport vc status fsm** Router# **debug mpls l2transport vc ldp**

*Feb 26 13:41:40.707: AToM LDP [110.1.1.2]: Sending label withdraw msg *Feb 26 13:41:40.707: AToM LDP [110.1.1.2]: VC Type 5, mtu 1500 *Feb 26 13:41:40.707: AToM LDP [110.1.1.2]: VC ID 100, label 18 *Feb 26 13:41:40.707: AToM LDP [110.1.1.2]: Status 0x0000000A [PW Status NOT supported]

Status Messages Indicating That the Attachment Circuit Is Down

When the attachment circuit is down between the two routers, the output of the **show mpls l2transport vc detail** command shows the following status:

```
Router# show mpls l2transport vc detail

.

.

.

Last remote LDP TLV    status rcvd: AC DOWN(rx,tx faults)
```

The debug messages also indicate that the attachment circuit is down, as shown in bold in the command output:

Router# debug mpls l2transport vc event Router# **debug mpls l2transport vc status event** Router# **debug mpls l2transport vc status fsm** Router# **debug mpls l2transport vc ldp**

```
*Feb 26 11:51:42.427: AToM LDP [10.1.1.1]: Received notif msg, id 88
*Feb 26 11:51:42.427: AToM LDP [10.1.1.1]:   Status    0x00000007 [PW Status]
*Feb 26 11:51:42.427: AToM LDP [10.1.1.1]:   PW Status 0x00000006 [AC DOWN(rx,tx faults)]
```

Other pseudowire status messages include not-forwarding, pw-tx-fault, and pw-rx-fault.

Message Codes in the Pseudowire Status Messages

The **debug mpls l2transport vc**and the **show mpls l2transport vc detail** commands show output that contains message codes. For example:

```
    Label/status state machine: established, LruRru
```

```
AToM MGR [10.9.9.9, 100]: S:Evt local up, LndRru->LnuRru
```

The message codes (LruRru, LndRru, and LnuRru) indicate the status of the local and remote routers. You can use the following key to interpret the message codes:

L—local router

R—remote router

r or n—ready (r) or not ready (n)

u or d—up (u) or down (d) status

The output also includes other values:

D—Dataplane

S—Local shutdown

How to Configure MPLS Pseudowire Status Signaling

- Enabling MPLS Pseudowire Status Signaling, page 266

Enabling MPLS Pseudowire Status Signaling

Perform the following task to enable the router to send pseudowire status to a peer router even when the attachment circuit is down. If both routers do not support pseudowire status messages, then disable the messages with the **no status** command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **pseudowire-class** *name*
4. **status**
5. **encapsulation mpls**
6. **exit**
7. **exit**
8. **show mpls l2transport vc detail**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**pseudowire-class** *name* **Example:** `Router(config)# pseudowire-class atom`	Establishes a pseudowire class with a name that you specify and enters pseudowire class configuration mode.
Step 4	**status** **Example:** `Router(config-pw)# status`	(Optional) Enables the router to send pseudowire status messages to the peer router through label advertisement and label notification messages. **Note** By default, status messages are enabled. This step is included only in case status messages have been disabled. If you need to disable status messages because both peer routers do not support this functionality, enter the **no status**command.
Step 5	**encapsulation mpls** **Example:** `Router(config-pw)# encapsulation mpls`	Specifies the tunneling encapsulation.
Step 6	**exit** **Example:** `Router(config-pw)# exit`	Exits pseudowire class configuration mode.
Step 7	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode.

	Command or Action	Purpose
Step 8	**show mpls l2transport vc detail** **Example:** `Router# show mpls l2transport vc detail`	Validates that pseudowire messages can be sent and received.

Configuration Examples for MPLS Pseudowire Status Signaling

- MPLS Pseudowire Status Signaling Example, page 268
- Verifying That Both Routers Support Pseudowire Status Messages Example, page 269

MPLS Pseudowire Status Signaling Example

The following example configures the MPLS Pseudowire Status Signaling feature on two PE routers. By default, status messages are enabled. The **status** command is included in this example in case status messages have been disabled.

PE1

```
interface Loopback0
 ip address 10.1.1.1 255.255.255.255
!
pseudowire-class atomstatus
 encapsulation mpls
 status
!
interface GigabitEthernet10/5
 xconnect 10.1.1.2 123 pw-class atomstatus
```

PE2

```
interface Loopback0
 ip address 10.1.1.2 255.255.255.255
!
pseudowire-class atomstatus
 encapsulation mpls
 status
!
interface GigabitEthernet3/3
 xconnect 10.1.1.1 123 pw-class atomstatus
```

Verifying That Both Routers Support Pseudowire Status Messages Example

You can issue the **show mpls l2transport vc detail** command to show that both the local and remote routers support pseudowire status messages. The following example shows the line of output to look for:

```
Router# show mpls l2transport vc detail
.
.
.
status TLV support (local/remote): enabled/supported
```

Additional References

The following sections provide references related to the MPLS Pseudowire Status Signaling feature.

Related Documents

Related Topic	Document Title
Any Transport over MPLS	Any Transport over MPLS
Virtual Private LAN Services	Virtual Private LAN Services on the Optical Services Modules

Standards

Standard	Title
draft-ietf-pwe3-control-protocol-15.txt	Pseudowire Setup and Maintenance Using LDP
draft-ietf-pwe3-iana-allocation-08.txt	IANA Allocations for Pseudo Wire Edge to Edge Emulation (PWE3)
draft-martini-pwe3-pw-switching-03.txt	Pseudo Wire Switching

MIBs

MIB	MIBs Link
Pseudowire Emulation Edge-to-Edge MIBs for Ethernet, Frame Relay, and ATM Services	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
None	—

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

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.