



Configuring EoMPLS

The Cisco ASR 901 router supports EoMPLS, a subset of AToM that uses a tunneling mechanism to carry Layer 2 Ethernet traffic. Ethernet Over MPLS (EoMPLS) encapsulates Ethernet frames in MPLS packets and forwards them across the MPLS network. In addition to dot1q, untagged, and default encapsulation support for an Ethernet Virtual Connection (EVC) with cross connect, effective with Cisco IOS Release 15.4(2)S, the Cisco ASR 901 router supports dot1ad encapsulation for the EVC with cross connect.

- [Understanding EoMPLS, on page 1](#)
- [Configuring EoMPLS, on page 2](#)
- [EoMPLS Configuration Example, on page 4](#)
- [Configuring EVC Default Encapsulation with xconnect, on page 5](#)
- [Configuring Pseudowire Redundancy, on page 6](#)
- [Port-Based EoMPLS, on page 8](#)
- [Feature Information for Configuring EoMPLS, on page 9](#)

Understanding EoMPLS

EoMPLS encapsulates Ethernet frames in MPLS packets and forwards them across the MPLS network. Each frame is transported as a single packet, and the PE routers connected to the backbone add and remove labels, as appropriate, for packet encapsulation:

- The ingress PE router receives an Ethernet frame and encapsulates the packet by removing the preamble, the Start Frame Delimiter (SFD), and the frame check sequence (FCS). The rest of the packet header is not changed.
- The ingress PE router adds a point-to-point virtual connection (VC) label and a label-switched path (LSP) tunnel label for normal MPLS routing through the MPLS backbone.
- The network core router uses the LSP tunnel label to move the packet through the MPLS backbone and does not distinguish Ethernet traffic from other types of packets in the MPLS backbone.
- At the other end of the MPLS backbone, the egress PE router receives the packet and de-encapsulates the packet by removing the LSP tunnel label, if present. The PE router also removes the VC label from the packet.
- The PE router updates the header, if necessary, and sends the packet out of the appropriate interface to the destination switch.

The MPLS backbone uses the tunnel labels to transport a packet between the PE routers. The egress PE router uses the VC label to select the outgoing interface for the Ethernet packet. Because EoMPLS tunnels are unidirectional, for bidirectional EoMPLS, you should configure one tunnel in each direction.

The point-to-point VC requires you to configure VC endpoints at the two PE routers. Only the PE routers at the ingress and egress points of the MPLS backbone know about the VCs dedicated to transporting Layer 2 traffic. Other routers do not have table entries for these VCs.

Restrictions for EoMPLS

- When configuring an EoMPLS pseudowire on the Cisco ASR 901 1 Router, you cannot configure an IP address on the same interface as the pseudowire.
- EoMPLS xconnect with VLAN range is not supported.
- EoMPLS xconnect port with double-tagged encapsulation is not supported.
- When port channel is configured on the MPLS core, the encapsulation ID should be equal to the bridge domain.
- To configure cross-connect with dot1ad encapsulation on an EVC, the interface should be a dot1ad NNI port. This means that a service instance with dot1q encapsulation cannot be configured on the port.
- Port-based cross-connect cannot be configured on the dot1ad NNI port interface.
- The encapsulation dot1ad command with cross-connect is not supported on the port channel.
- The dot1ad encapsulation with cross connect is not supported for double tag (QinQ).
- In case of encapsulation dot1ad over cross-connect, push operation at egress is not possible on cross-connect port in scenarios which requires pushing an additional dot1ad tag on the incoming dot1ad tag.
- The maximum number of cross-connect sessions supported on the Cisco ASR 901 Router is 1000. In case of pseudowire redundancy, a maximum of 500 sessions for primary and 500 sessions for backup pseudowire are supported.
- Default EFP under xconnect and untagged EFP under bridge domain on the same interface are not supported.
- Encapsulation is supported only on bridge domain and cross-connect.
- The **rewrite** command in the default EVC encapsulation is rejected.
- Default encapsulation with cross-connect is not supported on the port-channel interface.
- Untagged EFPs are supported only on the port with default encapsulation.
- Layer 3 routing is not supported. Layer 2 VPN is supported on the default encapsulation EFP.
- DSCP based classification for marking is not supported.

Configuring EoMPLS

Complete the following steps to configure EoMPLS:

Procedure

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

	Command or Action	Purpose
	Router# configure terminal	
Step 3	interface <i>interface-id</i> Example: Router(config)# interface gigabitethernet 0/1	Specify the interface, and enter interface configuration mode. Valid interfaces are physical ports. Perform Step 4 if you want to configure dot1ad encapsulation for an EVC with cross-connect. Go to Step 5 if you want to configure dot1q encapsulation for an EVC with cross-connect.
Step 4	ethernet dot1ad nni Example: Router(config-if)# ethernet dot1ad nni	Configures a dot1ad NNI port when you want to configure the dot1ad encapsulation for an EVC with cross-connect.
Step 5	service instance <i>instance-id</i> ethernet Example: Router(config-if)# service instance 101 ethernet	Configure a service instance and enter service instance configuration mode. <ul style="list-style-type: none"> • The <i>instance-id</i> —The service instance identifier, an integer from 1 to 4000. • (Optional) ethernet <i>name</i> —The name of a previously configured EVC. You do not need to use an EVC name in a service instance.
Step 6	encapsulation { dot1q <i>vlan-id</i> dot1ad <i>vlan-id</i> untagged default } Example: Router(config-if-srv)# encapsulation dot1q 51	Configure encapsulation type for the service instance. <ul style="list-style-type: none"> • dot1q—Configures 802.1Q encapsulation. • dot1ad—Configures 802.1ad encapsulation. • untagged—Maps to untagged VLANs. Only one EFP per port can have untagged encapsulation. • default—Configures default encapsulation to match all the ingress frames on the port.
Step 7	rewrite ingress tag pop 1 symmetric Example: Router(config-if-srv)#rewrite ingress tag pop 1 symmetric	(Optional) Specifies that encapsulation modification to occur on packets at ingress. <ul style="list-style-type: none"> • pop 1—Removes the outermost tag. • symmetric—Configures the packet to undergo the reverse of the ingress action at egress. If a tag is removed at ingress, it is added at egress. <p>Note Although the symmetric keyword appears to be optional, you must enter it for rewrite to function correctly.</p>

	Command or Action	Purpose
Step 8	xconnect <i>peer-ip-address</i> <i>vc-id</i> encapsulation mpls Example: Router(config-if-srv)# xconnect 192.168.1.8 101 encapsulation mpls	Configures cross-connect pseudowire by specifying the IP address of remote peer and the virtual circuit ID.

EoMPLS Configuration Example

The following is a sample configuration of dot1q encapsulation with cross-connect:

```

interface Loopback0
  description for_mpls_ldp
  ip address 99.99.99.99 255.255.255.255
!
interface GigabitEthernet0/10
  description Core_facing
  no negotiation auto
  service instance 150 ethernet
  encapsulation dot1q 150
  rewrite ingress tag pop 1 symmetric
  bridge-domain 150
!
interface GigabitEthernet0/11
  description CE_facing
  service instance 501 ethernet
  encapsulation dot1q 501
  rewrite ingress tag pop 1 symmetric
  xconnect 111.0.1.1 501 encapsulation mpls
!
interface FastEthernet0/0
  ip address 10.104.99.74 255.255.255.0
  full-duplex
!
interface Vlan1
!
interface Vlan150
  ip address 150.0.0.1 255.255.255.0
  mpls ip
!
router ospf 7
  network 99.99.99.99 0.0.0.0 area 0
  network 150.0.0.0 0.0.0.255 area 0
!
no ip http server
ip route 10.0.0.0 255.0.0.0 10.104.99.1
!
logging esm config
!
mpls ldp router-id Loopback0 force
!
!
end

```

The following is a sample configuration of dot1ad encapsulation with cross-connect:

```

!
interface GigabitEthernet0/1
  negotiation auto
  ethernet dot1ad nni
service instance 45 ethernet
  encapsulation dot1ad 45
  rewrite ingress tag pop 1 symmetric
  xconnect 13.13.13.13 45 encapsulation mpls
!

```

Configuring EVC Default Encapsulation with xconnect

Complete the following steps to configure EVC default encapsulation for xconnect.



Note When default encapsulation is configured on xconnect, the Cisco ASR 901 router does not support untagged encapsulation on the bridge domain of the same interface.

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Router> enable</pre>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>type number</i> Example: <pre>Router(config)# interface GigabitEthernet0/4</pre>	Specifies an interface type and number, and enters interface configuration mode.
Step 4	service instance <i>instance-id</i> ethernet Example: <pre>Router(config-if)# service instance 10 ethernet</pre>	Creates a service instance on an interface and defines the matching criteria. <ul style="list-style-type: none"> • <i>instance-id</i>—Integer that uniquely identifies a service instance on an interface.
Step 5	encapsulation default Example: <pre>Router(config-if)# encapsulation default</pre>	Configures the default service instance. Configures default encapsulation to match all the ingress frames on the port.

	Command or Action	Purpose
Step 6	xconnect <i>peer-ip-address</i> <i>vc-id</i> encapsulation mpls Example: <pre>Router(config-if-srv)# xconnect 1.1.1.1 100 encapsulation mpls</pre>	<p>Binds an attachment circuit to a pseudowire, and to configure an Any Transport over MPLS (AToM) static pseudowire.</p> <ul style="list-style-type: none"> • <i>peer-ip-address</i>—IP address of the remote provider edge (PE) peer. The remote router ID can be any IP address, as long as it is reachable. • <i>vc-id</i>—The 32-bit identifier of the virtual circuit (VC) between the PE routers. • encapsulation—Specifies the tunneling method to encapsulate the data in the pseudowire. • mpls—Specifies MPLS as the tunneling method.

Verifying EVC Default Encapsulation with xconnect

To verify the configuration of EVC default encapsulation with xconnect, use the **show** command shown below.

```
Router# show running-config interface gigabitEthernet 0/4
Building configuration...
Current configuration : 181 bytes
!
interface GigabitEthernet0/4
no ip address
negotiation auto
no keepalive
service instance 1 ethernet
 encapsulation default
  xconnect 2.2.2.2 100 encapsulation mpls
!
end
```

Configuration Example for EVC Default Encapsulation with Xconnect

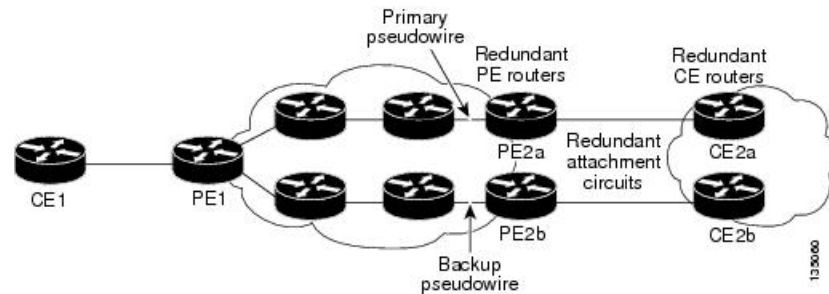
```
!
interface GigabitEthernet0/4
 service instance 10 ethernet
 encapsulation default
  xconnect 1.1.1.1 100 encapsulation mpls
!
```

Configuring Pseudowire Redundancy

Pseudowire (PW) Redundancy 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. Traffic can be switched back to the primary pseudowire after the path is operational again.

You can configure the network with redundant pseudowires and redundant network elements, as shown in the following figure.

Figure 1: Configuring Redundant Pseudowires



Configuration Commands

Complete the following steps to configure pseudowire redundancy:

Procedure

	Command or Action	Purpose
Step 1	<code>configure terminal</code>	Enters global configuration mode.
Step 2	<code>interface GigabitEthernet0/2</code>	Specifies an interface to configure.
Step 3	<code>service instance 101 ethernet</code>	Configures a service instance and enters the service instance configuration mode.
Step 4	<code>encapsulation dot1q 101</code>	Configures the encapsulation type for the service instance.
Step 5	<code>rewrite ingress tag pop 1 symmetric</code>	Specifies the encapsulation modification to be performed on packets at ingress. <ul style="list-style-type: none"> • pop 1—Removes the outermost tag. • symmetric—Configures the packet to undergo the reverse of the ingress action at egress. If a tag is removed at ingress, it is added at egress. <p>Note Although the symmetric keyword seems to be optional, you must enter it for rewrite to function correctly.</p>
Step 6	<code>xconnect 11.205.1.1 141 encapsulation mpls</code>	Binds the VLAN attachment circuit to an AToM pseudowire for EoMPLS.
Step 7	<code>backup peer 13.205.3.3 1141</code>	Specifies a backup peer for redundancy.
Step 8	<code>end</code>	Returns to privileged EXEC mode. <ul style="list-style-type: none"> • <code>show mpls l2t vc id</code>

	Command or Action	Purpose
		<ul style="list-style-type: none"> • show mpls l2t vc detail • show mpls infrastructure lfd pseudowire internal <p>Use these commands to display pseudowire information.</p>

Port-Based EoMPLS

The 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 frame check sequence (FCS) is transported as a single packet. To configure port mode, use the `xconnect` command in the main interface mode and specify the destination address and the VC ID. The syntax and semantics of the `xconnect` command are the same as for all other transport types. Each interface is associated with one unique pseudowire VC label.

Complete the following steps to configure port-based EoMPLS:

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Router> enable</pre>	Enables the privileged EXEC mode. Enter your password, if prompted.
Step 2	configure terminal Example: <pre>Router> configure terminal</pre>	Enters the global configuration mode.
Step 3	interface GigabitEthernet slot/port Example: <pre>Router(config)# interface GigabitEthernet 0/2</pre> Example: <pre>Router(config-if)#</pre>	Specifies an interface to configure.
Step 4	xconnect peer-router-id vcid encapsulation mpls Example: <pre>Router(config)# xconnect 10.0.0.1 123 encapsulation mpls</pre>	Binds the attachment circuit to a pseudowire VC. The syntax for this command is the same as that for all other Layer 2 transports.

What to do next

Feature Information for Configuring EoMPLS

The following table lists the features in this module and provides links to specific configuration information.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn> . An account on Cisco.com is not required.

**Note**

The following 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.

Table 1: Feature Information for EoMPLS

Feature Name	Releases	Feature Information
Configuring EoMPLS	15.1(2)SNI	This feature was introduced on the Cisco ASR 901 Routers.
802.1ad for EVC Cross Connect	15.4(2)S	This feature was introduced on the Cisco ASR 901 Routers.

