# Ethernet Interfaces Commands

This module describes the Cisco IOS XR software commands used to configure the Ethernet interfaces on the Cisco ASR 9000 Series Router.

**Note** This module does not include the commands for Management Ethernet interfaces and Ethernet OAM. To configure a Management Ethernet interface for routing or modify the configuration of a Management Ethernet interface or to configure Ethernet OAM, use the commands described in the *Interface and Hardware Component Configuration Guide for Cisco ASR 9000 Series Routers*

Refer to the *Interface and Hardware Component Command Reference for Cisco ASR 9000 Series Routers* for more information on the Ethernet Interfaces and Ethernet OAM commands.

# dot1q tunneling ethertype

To configure the Ethertype, used by peer devices when implementing QinQ VLAN tagging, to be 0x9100, use the **dot1q tunneling ethertype** command in the interface configuration mode for an Ethernet interface. To return to the default Ethertype configuration (0x8100), use the **no** form of this command.

**dot1q tunneling ethertype** {**0x9100** | **0x9200**}
**no dot1q tunneling ethertype**

**Syntax Description**

| | |
|---|---|
| **0x9100** | Sets the Ethertype value to 0x9100. |
| **0x9200** | Sets the Ethertype value to 0x9200. |

**Command Default**

The Ethertype field used by peer devices when implementing QinQ VLAN tagging is either 0x8100 or 0x8200.

**Command Modes**

Interface configuration mode

**Command History**

| Release | Modification |
|---|---|
| Release 3.9.0 | This command was introduced. |

**Usage Guidelines**

To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

The **dot1q tunneling ethertype** command can be applied to a main interface. When applied to the main interface, it changes the subinterfaces, that have been configured with an **encapsulation dot1q second-dot1q** command, under that main interface.

This command changes the outer VLAN tag from 802.1q Ethertype 0x8100 to 0x9100 or 0x9200.

**Task ID**

| Task ID | Operations |
|---|---|
| vlan | read, write |

**Examples**

The following example shows how to configure the Ethertype to 0x9100:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/1/5/0
RP/0/RSP0/CPU0:router(config-if)# dot1q tunneling ethertype 0x9100
RP/0/RSP0/CPU0:router(config-if)#
```

The following example shows how to configure the Ethertype to 0x9200:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/1/5/1
```

```
RP/0/RSP0/CPU0:router(config-if)# dot1q tunneling ethertype 0x9200
RP/0/RSP0/CPU0:router(config-if)#
```

| Related Commands | Command | Description |
|---|---|---|
| | encapsulation dot1q, on page 8 | Defines the matching criteria to map 802.1Q frames ingress on an interface to the appropriate service instance. |
| | encapsulation dot1ad dot1q, on page 6 | Defines the matching criteria to be used in order to map single-tagged 802.1ad frames ingress on an interface to the appropriate service instance. |
| | encapsulation dot1q second-dot1q, on page 10 | Defines the matching criteria to map Q-in-Q ingress frames on an interface to the appropriate service instance. |
| | encapsulation untagged, on page 12 | Defines the matching criteria to map untagged ingress Ethernet frames on an interface to the appropriate service instance. |

# encapsulation default

To configure the default service instance on a port, use the **encapsulation default** command in the Interface configuration mode. To delete the default service instance on a port, use the **no** form of this command.

**encapsulation default**
**no encapsulation default**

**Syntax Description**  This command has no keywords or arguments.

**Command Default**  No default service instance is configured on the port.

**Command Modes**  Interface configuration

**Command History**

| Release | Modification |
|---|---|
| Release 3.7.2 | This command was introduced. |

**Usage Guidelines**  To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

If the default service instance is the only one configured on a port, the **encapsulation default** command matches all ingress frames on that port. If the default service instance is configured on a port that has other non-default service instances, the **encapsulation default** command matches frames that are unmatched by those non-default service instances (anything that does not meet the criteria of other services instances on the same physical interface falls into this service instance).

Only a single default service instance can be configured per interface. If you attempt to configure more than one default service instance per interface, the **encapsulation default** command is rejected.

Only one encapsulation command must be configured per service instance.

**Examples**  The following example shows how to configure a service instance on a port:

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

**Related Commands**

| Command | Description |
|---|---|
| encapsulation dot1q, on page 8 | Defines the matching criteria to map 802.1Q frames ingress on an interface to the appropriate service instance. |
| encapsulation dot1ad dot1q, on page 6 | Defines the matching criteria to be used in order to map single-tagged 802.1ad frames ingress on an interface to the appropriate service instance. |
| encapsulation dot1q second-dot1q, on page 10 | Defines the matching criteria to map Q-in-Q ingress frames on an interface to the appropriate service instance. |

| Command | Description |
| --- | --- |
| encapsulation untagged, on page 12 | Defines the matching criteria to map untagged ingress Ethernet frames on an interface to the appropriate service instance. |

# encapsulation dot1ad dot1q

To define the matching criteria to be used in order to map single-tagged 802.1ad frames ingress on an interface to the appropriate service instance, use the **encapsulation dot1ad dot1q** command in subinterface configuration mode. To delete the matching criteria to map single-tagged 802.1ad frames ingress on an interface to the appropriate service instance, use the **no** form of this command.

**encapsulation dot1ad** *vlan-id* **dot1q** {*vlan-id*}
**no encapsulation dot1ad** *vlan-id* **dot1q** {*vlan-id*}

**Syntax Description**

| | |
|---|---|
| **dot1ad** | Indicates that the IEEE 802.1ad provider bridges encapsulation type is used for the outer tag. |
| **dot1q** | Indicates that the IEEE 802.1q standard encapsulation type is used for the inner tag. |
| *vlan-id* | VLAN ID, integer in the range 1 to 4094. |
| | A hyphen must be entered to separate the starting and ending VLAN ID values that are used to define a range of VLAN IDs. (Optional) A comma must be entered to separate each VLAN ID range from the next range. |

**Command Default**
No matching criteria are defined.

**Command Modes**
Subinterface configuration

**Command History**

| Release | Modification |
|---|---|
| Release 3.9.0 | This command was introduced. |

**Usage Guidelines**
To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

The outer VLAN tag is an 802.1ad VLAN tag, instead of an 802.1Q tag. An 802.1ad tag has an ethertype value of 0x88A8, instead of 0x8100 that 802.1Q uses.

Some of the fields in the 802.1ad VLAN header are interpreted differently per 802.1ad standard. A **tunneling ethertype** command applied to the main interface does not apply to an 802.1ad subinterface.

An interface with encapsulation dot1ad causes the router to categorize the interface as an 802.1ad interface. This causes special processing for certain protocols and other features:

- MSTP uses the IEEE 802.1ad MAC STP address instead of the STP MAC address.
- Certain QoS functions may use the Drop Eligibility (DE) bit of the IEEE 802.1ad tag.

**Examples**
The following example shows how to map single-tagged 802.1ad ingress frames to a service instance:

```
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1ad 100 dot1q 20
```

| Related Commands | Command | Description |
| --- | --- | --- |
| | encapsulation default, on page 4 | Configure the default service instance on a port. |
| | encapsulation dot1q, on page 8 | Defines the matching criteria to map 802.1Q frames ingress on an interface to the appropriate service instance. |
| | encapsulation untagged, on page 12 | Defines the matching criteria to map untagged ingress Ethernet frames on an interface to the appropriate service instance. |

# encapsulation dot1q

To define the matching criteria to map 802.1Q frames ingress on an interface to the appropriate service instance, use the **encapsulation dot1q** command in the interface configuration mode. To delete the matching criteria to map 802.1Q frames ingress on an interface to the appropriate service instance, use the **no** form of this command.

**encapsulation dot1q** *vlan-id* [*,vlan-id* [*-vlan-id*]] [{**exact** | **ingress source-mac** *mac-address* | **second-dot1q** *vlan-id*}]
**encapsulation dot1q** *vlan-id,* **untagged**
**no encapsulation dot1q**

| Syntax Description | | |
|---|---|---|
| **vlan-id** | VLAN ID, integer in the range 1 to 4094. | |
| | Hyphen must be entered to separate the starting and ending VLAN ID values that are used to define a range of VLAN IDs. (Optional) Comma must be entered to separate each VLAN ID range from the next range. | |
| **exact** | (Optional) Prevents matching of frames with more than one tag. | |
| **ingress source-mac** | (Optional) Performs MAC-based matching. | |
| **untagged** | (Optional) Allows matches for both the single-tag dot1q frames and untagged frames. | |

**Command Default**  No matching criteria are defined.

**Command Modes**  Interface configuration

**Command History**

| Release | Modification |
|---|---|
| Release 3.7.2 | This command was introduced. |
| Release 3.9.1 | The **ingress source-mac** keyword was added. |
| Release 4.0.1 | This command was supported on l2transport subinterfaces. |

**Usage Guidelines**  To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Only one encapsulation statement can be applied to a subinterface. Encapsulation statements cannot be applied to main interfaces.

A single encapsulation dot1q statement specifies matching for frames with a single VLAN ID; a range of VLAN IDs; or a single VLAN ID or untagged.

**Examples**  The following example shows how to map 802.1Q frames ingress on an interface to the appropriate service instance:

```
RP/0/RSP0/CPU0:router(config-if)# encapsulation dot1q 10
```

| Related Commands | Command | Description |
|---|---|---|
| | encapsulation default, on page 4 | Configure the default service instance on a port. |
| | encapsulation dot1ad dot1q, on page 6 | Defines the matching criteria to be used in order to map single-tagged 802.1ad frames ingress on an interface to the appropriate service instance. |
| | encapsulation dot1q second-dot1q, on page 10 | Defines the matching criteria to map Q-in-Q ingress frames on an interface to the appropriate service instance. |
| | encapsulation untagged, on page 12 | Defines the matching criteria to map untagged ingress Ethernet frames on an interface to the appropriate service instance. |

# encapsulation dot1q second-dot1q

To define the matching criteria to map Q-in-Q ingress frames on an interface to the appropriate service instance, use the **encapsulation dot1q second-dot1q** command in the interface configuration mode. To delete the matching criteria to map Q-in-Q ingress frames on an interface to the appropriate service instance, use the **no** form of this command.

**encapsulation dot1q** {**any** | *vlan-id* [*,vlan-id* [*-vlan-id]*] **second-dot1q** {**any** | *vlan-id* [*,vlan-id* [*-vlan-id]*] [{**exact** | **ingress source-mac** *mac-address*}]}
**no encapsulation dot1q** {**any** | *vlan-id* [*,vlan-id* [*-vlan-id]*] **second-dot1q** {**any** | *vlan-id* [*,vlan-id* [*-vlan-id]*] [{**exact** | **ingress source-mac** *mac-address*}]}

| Syntax Description | | |
|---|---|---|
| *vlan-id* | | VLAN ID, integer in the range 1 to 4094. |
| | | A hyphen must be entered to separate the starting and ending VLAN ID values that are used to define a range of VLAN IDs. |
| | | (Optional) A comma must be entered to separate each VLAN ID range from the next range. |
| | | A maximum of nine ranges or individual values may be specified. The values must not overlap. |
| **second-dot1q** | | (Optional) Specifies IEEE 802.1Q VLAN tagged packets. |
| **any** | | Any second tag in the range 1 to 4094. |
| **exact** | | (Optional) Ensures that frames with more than two tags do not match. |
| **ingress source-mac** | | (Optional) Performs MAC-based matching. |

**Command Default**   No matching criteria are defined.

**Command Modes**   Interface configuration

**Command History**

| Release | Modification |
|---|---|
| Release 3.7.2 | This command was introduced. |
| Release 3.9.1 | The **ingress source-mac** keyword was added. |

**Usage Guidelines**   To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

The criteria for this command are: the outer tag must be unique and the inner tag may be a single VLAN, a range of VLANs or lists of the previous two.

QinQ service instance, allows single, multiple or range on second-dot1q.

Only one encapsulation command must be configured per service instance.

**Examples**

The following example shows how to map ingress frames to a service instance:

```
RP/0/RSP0/CPU0:router(config-if)# encapsulation dot1q second-dot1q 20
```

**Related Commands**

| Command | Description |
|---|---|
| encapsulation default, on page 4 | Configure the default service instance on a port. |
| encapsulation dot1ad dot1q, on page 6 | Defines the matching criteria to be used in order to map single-tagged 802.1ad frames ingress on an interface to the appropriate service instance. |
| encapsulation dot1q, on page 8 | Defines the matching criteria to map 802.1Q frames ingress on an interface to the appropriate service instance. |
| encapsulation untagged, on page 12 | Defines the matching criteria to map untagged ingress Ethernet frames on an interface to the appropriate service instance. |

# encapsulation untagged

To define the matching criteria to map untagged ingress Ethernet frames on an interface to the appropriate service instance, use the **encapsulation untagged** command in the Interface configuration mode. To delete the matching criteria to map untagged ingress Ethernet frames on an interface to the appropriate service instance, use the **no** form of this command.

**encapsulation untagged** [**ingress source-mac** *mac-address*]
**no encapsulation untagged**

| Syntax Description | ingress source-mac | (Optional) Performs MAC-based matching. |
|---|---|---|
| | *mac-address* | Specifies the source MAC address. |

**Command Default**  No matching criteria are defined.

**Command Modes**  Interface configuration

**Command History**

| Release | Modification |
|---|---|
| Release 3.7.2 | This command was introduced. |
| Release 3.9.1 | The **ingress source-mac** keyword was added. |

**Usage Guidelines**  To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Only one service instance per port is allowed to have untagged encapsulation. The reason is to be able to unambiguously map the incoming frames to the service instance. However, it is possible for a port that hosts an service instance matching untagged traffic to host other service instances that match tagged frames. Only one encapsulation command may be configured per service instance.

Only one subinterface may be configured as encapsulation untagged. This interface is referred to as the untagged subinterface or untagged EFP (incase of an L2 interface).

The untagged subinterface has a higher priority than the main interface; all untagged traffic, including L2 protocol traffic, passes through this subinterface rather than the main interface. If the **ethernet filtering** command is applied to a main interface having an untagged subinterface, the filtering is applied to the untagged subinterface.

**Examples**  The following example shows how to map untagged ingress Ethernet frames to a service instance:

Example 1:

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

Example 2:

```
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/1/1/0.100 l2transport
RP/0/RSP0/CPU0:router(config-subif)# encapsulation untagged
```

| | Command | Description |
|---|---|---|
| **Related Commands** | encapsulation default, on page 4 | Configure the default service instance on a port. |
| | encapsulation dot1q, on page 8 | Defines the matching criteria to map 802.1Q frames ingress on an interface to the appropriate service instance. |
| | encapsulation dot1q second-dot1q, on page 10 | Defines the matching criteria to map Q-in-Q ingress frames on an interface to the appropriate service instance. |

# ethernet egress-filter

To enable strict egress filtering on all subinterfaces on the router by default, use the **ethernet egress-filter** command in global configuration mode.

**ethernet  egress-filter  strict**

To enable or disable egress filtering explicitly on any Layer 2 subinterface, use the **ethernet egress-filter** command in Layer 2 subinterface mode.

**ethernet  egress-filter  {strict | disabled}**

| Syntax Description | | |
|---|---|---|
| **strict** | Enables strict egress EFP filtering on the interface. Only packets that pass the ingress EFP filter on the interface can be transmitted out of this interface. Other packets are dropped at the egress filter. | |
| **disabled** | Disables strict egress EFP filtering on the interface. This allows packets that do not match the interface encapsulation to be transmitted out of the interface. | |

**Command Default**   For platforms that support this command, the global default is that subinterface egress encapsulation filtering is disabled.

**Command Modes**   Global configuration and Layer 2 subinterface configuration

**Command History**

| Release | Modification |
|---|---|
| Release 3.7.3 | This command was introduced. |

**Usage Guidelines**   To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

**Task ID**

| Task ID | Operations |
|---|---|
| **interface** | read, write |

**Examples**   The following example shows how to enable strict egress filtering on all subinterfaces in global configuration mode:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# ethernet egress-filter strict
```

The following example shows how to enable the strict egress filtering on any Layer 2 subinterface in Layer 2 subinterface mode:

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

```
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/1/0/1.1
RP/0/RSP0/CPU0:router(config-subif)# ethernet egress-filter strict
```

# ethernet filtering

To enable ethernet filtering on interfaces on the router, use the **ethernet filtering** command in the interface configuration mode. To disable ethernet filtering, use the **no** form of the command.

**ethernet  filtering  {dot1ad | dot1q}**
**no  ethernet  filtering**

**Syntax Description**

| | |
|---|---|
| **dot1ad** | Filters only the Ethernet multicast protocol addresses that are reserved by IEEE 802.1ad, used for C-facing interfaces, to prevent C-network traffic from interfering with the S-network protocols. |
| **dot1q** | Filters all Ethernet multicast protocol addresses. |

**Command Default**
Ethernet filtering is not enabled.

**Command Modes**
interface configuration mode

**Command History**

| Release | Modification |
|---|---|
| Release 3.9.0 | This command was introduced. |

**Usage Guidelines**
To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

The following table lists the DA MAC addresses and specifies the action taken when either the dot1q or the dot1ad keywords are used:

| DA MAC Address | Description | dot1q | dot1ad |
|---|---|---|---|
| 01-80-C2-00-00-00 | STP, RSTP, MSTP, etc. | Discard | Data |
| 01-80-C2-00-00-01 | 802.3X Pause Protocol | Discard | Discard |
| 01-80-C2-00-00-02 | Slow Protocols: 802.3ad LACP, 802.3ah OAM | Discard | Discard |
| 01-80-C2-00-00-03 | 802.1X | Discard | Discard |
| 01-80-C2-00-00-04 | Reserved | Discard | Discard |
| 01-80-C2-00-00-05 | Reserved | Discard | Discard |
| 01-80-C2-00-00-06 | Reserved | Discard | Discard |
| 01-80-C2-00-00-07 | Reserved | Discard | Discard |
| 01-80-C2-00-00-08 | Provider Bridge Group Address (e.g. MSTP BPDU) | Discard | Discard |
| 01-80-C2-00-00-09 | Reserved | Discard | Discard |

| DA MAC Address | Description | dot1q | dot1ad |
|---|---|---|---|
| 01-80-C2-00-00-0A | Reserved | Discard | Discard |
| 01-80-C2-00-00-0B | Reserved | Discard | Data |
| 01-80-C2-00-00-0C | Reserved | Discard | Data |
| 01-80-C2-00-00-0D | Provider Bridge GVRP address | Discard | Data |
| 01-80-C2-00-00-0E | 802.1ab-LLDP | Discard | Data |
| 01-80-C2-00-00-0F | Reserved | Discard | Data |
| 01-80-C2-00-00-10 | All Bridges address | Discard | Data |
| 01-80-C2-00-00-20 | GMRP / MMRP | Discard | Data |
| 01-80-C2-00-00-21 | GVRP / MVRP | Discard | Data |
| 01-80-C2-00-00-22-2F | Other GARP addresses | Discard | Data |
| 01-00-0C-CC-CC-CC | CDP, DTP, VTP, PaGP, UDLD | Discard | Data |

**Task ID**

| Task ID | Operations |
|---|---|
| **interface** | read, write |

**Examples**

The following example shows how to apply ethernet filtering on a main interface:

```
RP/0/RSP0/CPU0:router#configure
RP/0/RSP0/CPU0:router(config)#interface GigabitEthernet0/5/0/1
RP/0/RSP0/CPU0:router(config-if)#ethernet filtering dot1q
RP/0/RSP0/CPU0:router(config-if)#l2transport
RP/0/RSP0/CPU0:router(config-if-l2)#commit

RP/0/RSP0/CPU0:router#show run | begin GigabitEthernet0/5/0/1
Tue Nov 24 12:29:55.718 EST
Building configuration...
interface GigabitEthernet0/5/0/1
 mtu 1500
 ethernet filtering dot1q
 l2transport
 !
!
interface GigabitEthernet0/5/0/2
 shutdown
!
interface GigabitEthernet0/5/0/3
 shutdown
!
interface GigabitEthernet0/5/0/4
 shutdown
!
interface GigabitEthernet0/5/0/5
```

```
 shutdown
!
interface GigabitEthernet0/5/0/6
 shutdown
!
interface GigabitEthernet0/5/0/7
 shutdown
RP/0/RSP0/CPU0:router#
```

The following example shows how to apply ethernet filtering on a subinterface:

```
RP/0/RSP0/CPU0:router#configure
RP/0/RSP0/CPU0:router(config)#interface GigabitEthernet0/5/0/1
RP/0/RSP0/CPU0:router(config-if)#ethernet filtering dot1q
RP/0/RSP0/CPU0:router(config-if)#interface GigabitEthernet0/5/0/1.1 l2transport
RP/0/RSP0/CPU0:router(config-subif)#encapsulation untagged
RP/0/RSP0/CPU0:router(config-subif)#commit
RP/0/RSP0/CPU0:router(config-subif)#end

RP/0/RSP0/CPU0:router#show run | begin GigabitEthernet0/5/0/1
Tue Nov 24 12:26:25.494 EST
Building configuration...
interface GigabitEthernet0/5/0/1
 mtu 1500
 ethernet filtering dot1q
!
interface GigabitEthernet0/5/0/1.1 l2transport
 encapsulation untagged
!
interface GigabitEthernet0/5/0/2
 shutdown
!
interface GigabitEthernet0/5/0/3
 shutdown
!
interface GigabitEthernet0/5/0/4
 shutdown
!
interface GigabitEthernet0/5/0/5
 shutdown
!
interface GigabitEthernet0/5/0/6
 shutdown
!
interface GigabitEthernet0/5/0/7
RP/0/RSP0/CPU0:router#
```

**Note**  Ethernet filtering is configured on the main interface; however, the configuration affects the subinterface and not the main interface.

# ethernet source bypass egress-filter

To mark all ingress packets, received on the interface, to indicate that the packets should bypass any strict egress filter on any egress interface, use the **ethernet source bypass egress-filter** command in the subinterface configuration mode. To allow packets without being marked, use the **no** form of this command.

**ethernet source bypass egress-filter**
**no ethernet source bypass egress-filter**

This command has no keywords or arguments.

| | |
|---|---|
| **Command Default** | None |

| | |
|---|---|
| **Command Modes** | Subinterface configuration |

**Command History**

| Release | Modification |
|---|---|
| Release 3.9.1 | This command was introduced. |

**Usage Guidelines**

To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

**Task ID**

| Task ID | Operations |
|---|---|
| interface | read, write |

**Examples**

The following example shows how to mark all ingress packets received on the interface:

```
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet0/0/0/0/3.1 l2transport
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 1
RP/0/RSP0/CPU0:router(config-subif)# rewrite ingress tag translate 1-to-1 dot1q 4094 symmetric
RP/0/RSP0/CPU0:router(config-subif)# ethernet egress-filter disabled
RP/0/RSP0/CPU0:router(config-subif)# ethernet source-bypass-egress-filter
```

**Related Commands**

| Command | Description |
|---|---|
| encapsulation dot1q, on page 8 | Defines the matching criteria to map 802.1Q frames ingress on an interface to the appropriate service instance. |

# l2protocol (Ethernet)

To configure Layer 2 protocol tunneling and protocol data unit (PDU) filtering on an Ethernet interface, use the **l2protocol** command in Layer 2 transport configuration mode. To disable a Layer 2 protocol tunneling and Layer 2 protocol data units configuration, use the **no** form of this command.

**l2protocol cpsv** {**tunnel** | **reverse-tunnel**}
**no l2protocol**

| Syntax Description | | |
|---|---|---|
| **cpsv** | Enables L2PT for the interface. L2PT is enabled for the following protocols only: | |

**cpsv**      Enables L2PT for the interface. L2PT is enabled for the following protocols only:

- CDP
- STP
- VTP

**Note**      STP includes all Spanning Tree protocol derivatives (RSTP, MSTP, etc.)

**tunnel**      Performs L2PT encapsulation on frames as they enter the interface. Also, performs L2PT de-encapsulation on frames as they exit they interface.

L2PT encapsulation rewrites the destination MAC address with the L2PT destination MAC address. L2PT deencapsulation replaces the L2PT destination MAC address with the original destination MAC address.

**reverse-tunnel**      Performs L2PT encapsulation on frames as they exit the interface. Also, perform L2PT deencapsulation on frames as they enter the interface.

**Command Default**      All Layer 2 protocol data units are forwarded through the network without modification.

**Command Modes**      Layer 2 transport configuration

**Command History**

| Release | Modification |
|---|---|
| Release 3.9.1 | This command was introduced. |

**Usage Guidelines**      To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

**Note**      The **l2protocol** command is available only when Layer 2 transport port mode is enabled on the interface with the **l2transport** command.

**Task ID**

| Task ID | Operations |
|---------|------------|
| l2vpn | read, write |

**Examples**

The following example shows how to configure an Ethernet interface to tunnel in the ingress direction:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface TenGigE 0/0/0/1
RP/0/RSP0/CPU0:router(config-if)# l2transport
RP/0/RSP0/CPU0:router(config-if-l2)# l2protocol cpsv tunnel
```

**Related Commands**

| Command | Description |
|---------|-------------|
| l2transport (Ethernet), on page 22 | Enables Layer 2 transport port mode on an Ethernet interface and enter Layer 2 transport configuration mode. |

# l2transport (Ethernet)

To enable Layer 2 transport port mode on an Ethernet interface and enter Layer 2 transport configuration mode, use the **l2transport** command in interface configuration mode for an Ethernet interface. To disable Layer 2 transport port mode on an Ethernet interface, use the **no** form of this command.

**l2transport**
**no  l2transport**

This command has no keywords or arguments.

**Command Default**   None

**Command Modes**   Interface configuration

**Command History**

| Release | Modification |
|---|---|
| Release 3.7.2 | This command was introduced. |

**Usage Guidelines**   To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

When you issue the **l2transport** command in interface configuration mode, the CLI prompt changes to "config-if-l2," indicating that you have entered the Layer 2 transport configuration submode. In the following sample output, the question mark (**?**) online help function displays all the commands available under Layer 2 transport configuration submode for an Ethernet interface:

```
RP/0/RSP0/CPU0:router#configure
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/1/5/0
RP/0/RSP0/CPU0:router(config-if)# l2transport
RP/0/RSP0/CPU0:router(config-if-l2)# ?
  commit          Commit the configuration changes to running
  describe        Describe a command without taking real actions
  do              Run an exec command
  exit            Exit from this submode
  no              Negate a command or set its defaults
  service-policy  Configure QoS Service policy
  show            Show contents of configuration
RP/0/RSP0/CPU0:router(config-if-l2)#
```

**Note**   The **l2transport** command is mutually exclusive with any Layer 3 interface configuration.

**Task ID**

| Task ID | Operations |
|---|---|
| l2vpn | read, write |

**Examples**

The following example shows how to enable Layer 2 transport port mode on an Ethernet interface and enter Layer 2 transport configuration mode:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface GigabitEther 0/2/0/0
RP/0/RSP0/CPU0:router(config-if)# l2transport
RP/0/RSP0/CPU0:router(config-if-l2)#
```

The following example shows how to use the **l2transport** keyword in the **interface** command:

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface GigabitEther 0/2/0/0 l2transport
RP/0/RSP0/CPU0:router(config-if)# encapsulation dot1q 200
RP/0/RSP0/CPU0:router(config-if-l2)#commit
```

The following example shows how to use the **l2transport** command on an Ethernet subinterface:

**Note** Ensure that the **l2transport** command is applied on the same line as the **interface** command for the Ethernet subinterface.

```
RP/0/RSP0/CPU0:router#configure
RP/0/RSP0/CPU0:router(config)#interface GigabitEthernet 0/5/0/1.1 l2transport
RP/0/RSP0/CPU0:router(config-subif)#encapsulation dot1q 100
RP/0/RSP0/CPU0:router(config-subif)#ethernet egress-filter strict
RP/0/RSP0/CPU0:router(config-subif)#commit
RP/0/RSP0/CPU0:router(config-subif)#end

RP/0/RSP0/CPU0:router#sh run | begin GigabitEthernet0/5/0/1
Thu Dec  3 10:15:40.916 EST Building configuration...
interface GigabitEthernet0/5/0/1
 mtu 1500
 ethernet filtering dot1q
!
interface GigabitEthernet0/5/0/1.1 l2transport
encapsulation dot1q 100
ethernet egress-filter strict !
interface GigabitEthernet0/5/0/2
 shutdown
!
!
```

**Note** To configure l2transport on an Ethernet subinterface, ensure that the main interface is configured as a Layer 3 interface.

**Related Commands**

| Command | Description |
|---|---|
| **show interfaces** | Displays statistics for all interfaces configured on the router or for a specific node. |
| **show l2vpn xconnect** | Displays brief information on configured xconnects. |

# local-traffic default encapsulation

To enable Connectivity Fault Management (CFM) to identify a range of VLAN IDs that are to be used as the default for sourcing CFM packets from the interface, use the **local-traffic default encapsulation** command in the subinterface configuration mode. To return to the default behavior, use the **no** form of this command.

**local-traffic default encapsulation** {**dot1q** *vlan-id* | **dot1q** *vlan-id* **second-dot1q** *vlan-id* | **dot1ad** *vlan-id* | **dot1ad** *vlan-id* **dot1q** *vlan-id*}
**no local-traffic default encapsulation** {**dot1q** *vlan-id* | **dot1q** *vlan-id* **second-dot1q** *vlan-id* | **dot1ad** *vlan-id* | **dot1ad** *vlan-id* **dot1q** *vlan-id*}

**Syntax Description**

| | |
|---|---|
| **dot1q** | Indicates that the IEEE 802.1q standard encapsulation type is used. |
| **second-dot1q** | Indicates that the IEEE 802.1q encapsulation is used. |
| **dot1ad** | Indicates that the IEEE 802.1ad provider bridges encapsulation type is used. |
| *vlan-id* | Specifies the VLAN ID as an integer. The range is 1 to 4094. A hyphen separates the starting and ending VLAN ID values that are used when defining a range of VLAN IDs. |

**Command Default**

Lowest numbered VLAN ID is chosen.

**Command Modes**

Subinterface configuration

**Command History**

| Release | Modification |
|---|---|
| Release 3.9.1 | This command was introduced. |

**Usage Guidelines**

To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

The tag stack configured by the **local-traffic default encapsulation** command must match the encapsulation specified for this interface in the **encapsulation** command.

For packets that are sent as responses to incoming packets, the encapsulation that is to be used may be derived from the incoming packet. This command determines the encapsulation to use when this is not the case.

**Task ID**

| Task ID | Operations |
|---|---|
| interface | read, write |

**Examples**

The following example indicates that the locally sourced frames (not sent in response to another ingress frame) sent out of GigabitEthernet subinterface 0/3/0/1.1 should be tagged with 802.1Q VLAN 50. When the local-traffic is not configured, chooses the lowest value in the range and sends the frames out tagged with 802.1Q VLAN 10.

```
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/3/0/1.1 l2transport
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 10-100
RP/0/RSP0/CPU0:router(config-subif)# local-traffic default encapsulation dot1q 50
```

The followoing example indicates that the locally sourced frames are sent out with an outer VLAN tag of 802.1Q 1000, and an inner VLAN tag of 802.1Q 500. Without configuring the local-traffic, the frames are sent out with an outer VLAN tag of 1000 and an inner VLAN tag of 1:

```
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet0/0/0/0.2 l2transport
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 1000 second-dot1q 1-500
RP/0/RSP0/CPU0:routerr(config-subif)# local-traffic default encapsulation dot1q 1000
second-dot1q 500
```

# rewrite ingress tag

To specify the encapsulation adjustment that is to be performed on the frame ingress to the service instance, use the **rewrite ingress tag** command in the interface configuration mode. To delete the encapsulation adjustment that is to be performed on the frame ingress to the service instance, use the **no** form of this command.

**rewrite ingress tag** {**push** {**dot1q** *vlan-id* | **dot1q** *vlan-id* **second-dot1q** *vlan-id* | **dot1ad** *vlan-id* **dot1q** *vlan-id*} | **pop** {**1** | **2**} | **translate** {**1to1** {**dot1q** *vlan-id* | **dot1ad** *vlan-id*} | **2-to-1 dot1q** *vlan-id* | **dot1ad** *vlan-id*} | **1-to-2** {**dot1q** *vlan-id* **second-dot1q** *vlan-id* | **dot1ad** *vlan-id* **dot1q** *vlan-id*} | **2-to-2** {**dot1q** *vlan-id* **second-dot1q** *vlan-id* | **dot1ad** *vlan-id* **dot1q** *vlan-id*}} [**symmetric**]
**no rewrite ingress tag** {**push** {**dot1q** *vlan-id* | **dot1q** *vlan-id* **second-dot1q** *vlan-id* | **dot1ad** *vlan-id* **dot1q** *vlan-id*} | **pop** {**1** | **2**} | **translate** {**1to1** {**dot1q** *vlan-id* | **dot1ad** *vlan-id*} | **2-to-1 dot1q** *vlan-id* | **dot1ad** *vlan-id*} | **1-to-2** {**dot1q** *vlan-id* **second-dot1q** *vlan-id* | **dot1ad** *vlan-id* **dot1q** *vlan-id*} | **2-to-2** {**dot1q** *vlan-id* **second-dot1q** *vlan-id* | **dot1ad** *vlan-id* **dot1q** *vlan-id*}} [**symmetric**]

**Syntax Description**

| | |
|---|---|
| *vlan-id* | VLAN ID, integer in the range 1 to 4094. |
| **push dot1q** *vlan-id* | Pushes one 802.1Q tag with *vlan-id*. |
| **push dot1q** *vlan-id* **second-dot1q** *vlan-id* | Pushes a pair of 802.1Q tags in the order first, second. |
| **pop {1 | 2}** | One or two tags are removed from the packet. This command can be combined with a push (pop N and subsequent push *vlan-id*). |
| **translate 1-to-1 dot1q** *vlan-id* | Replaces the incoming tag (defined in the encapsulation command) into a different 802.1Q tag at the ingress service instance. |
| **translate 2-to-1 dot1q** *vlan-id* | Replaces a pair of tags defined in the **encapsulation** command by vlan-id. |
| **translate 1-to-2 dot1q** *vlan-id* **second-dot1q** *vlan-id* | Replaces the incoming tag defined by the encapsulation command by a pair of 802.1Q tags. |
| **translate 2-to-2 dot1q** *vlan-id* **second-dot1q** *vlan-id* | Replaces the pair of tags defined by the encapsulation command by a pair of VLANs defined by this rewrite. |
| **symmetric** | (Optional) A rewrite operation is applied on both ingress and egress. The operation on egress is the inverse operation as ingress. |

**Command Default**     The frame is left intact on ingress.

**Command Modes**     Interface configuration

**Command History**

| Release | Modification |
|---|---|
| Release 3.7.2 | This command was introduced. |

**Usage Guidelines**

To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

The **symmetric** keyword is accepted only when a single VLAN is configured in encapsulation. If a list of VLANs or a range VLAN is configured in encapsulation, the **symmetric** keyword is accepted only for push rewrite operations; all other rewrite operations are rejected.

The **pop** command assumes the elements being popped are defined by the encapsulation type. The exception case should be drop the packet.

The **rewrite ingress tag translate**command assume the tags being translated from are defined by the encapsulation type. In the 2-to-1 option, the "2" means "2 tags of a type defined by the **encapsulation** command. The translation operation requires at least "from" tag in the original packet. If the original packet contains more tags than the ones defined in the "from", then the operation should be done beginning on the outer tag. Exception cases should be dropped.

**Examples**

The following example shows how to specify the encapsulation adjustment that is to be performed on the frame ingress to the service instance:

```
RP/0/RSP0/CPU0:router(config-if)# rewrite ingress push dot1q 200
```

**Related Commands**

| Command | Description |
|---|---|
| encapsulation default, on page 4 | Configure the default service instance on a port. |
| encapsulation dot1ad dot1q, on page 6 | Defines the matching criteria to be used in order to map single-tagged 802.1ad frames ingress on an interface to the appropriate service instance. |
| encapsulation dot1q, on page 8 | Defines the matching criteria to map 802.1Q frames ingress on an interface to the appropriate service instance. |
| encapsulation dot1q second-dot1q, on page 10 | Defines the matching criteria to map Q-in-Q ingress frames on an interface to the appropriate service instance. |
| encapsulation untagged, on page 12 | Defines the matching criteria to map untagged ingress Ethernet frames on an interface to the appropriate service instance. |

**rewrite ingress tag**