

IEEE 802.1Q-in-Q VLAN Tag Termination

Encapsulating IEEE 802.1Q VLAN tags within 802.1Q enables service providers to use a single VLAN to support customers who have multiple VLANs. The IEEE 802.1Q-in-Q VLAN Tag Termination feature on the subinterface level preserves VLAN IDs and keeps traffic in different customer VLANs segregated.

- Information About IEEE 802.1Q-in-Q VLAN Tag Termination, on page 1
- How to Configure IEEE 802.1Q-in-Q VLAN Tag Termination, on page 3
- Configuration Examples for IEEE 802.1Q-in-Q VLAN Tag Termination, on page 8
- Additional References, on page 10
- Feature Information for IEEE 802.1Q-in-Q VLAN Tag Termination, on page 11

Information About IEEE 802.1Q-in-Q VLAN Tag Termination

IEEE 802.1Q-in-Q VLAN Tag Termination on Subinterfaces

IEEE 802.1Q-in-Q VLAN Tag Termination simply adds another layer of IEEE 802.1Q tag (called “metro tag” or “PE-VLAN”) to the 802.1Q tagged packets that enter the network. The purpose is to expand the VLAN space by tagging the tagged packets, thus producing a “double-tagged” frame. The expanded VLAN space allows the service provider to provide certain services, such as Internet access on specific VLANs for specific customers, and yet still allows the service provider to provide other types of services for their other customers on other VLANs.

Generally the service provider’s customers require a range of VLANs to handle multiple applications. Service providers can allow their customers to use this feature to safely assign their own VLAN IDs on subinterfaces because these subinterface VLAN IDs are encapsulated within a service-provider designated VLAN ID for that customer. Therefore there is no overlap of VLAN IDs among customers, nor does traffic from different customers become mixed. The double-tagged frame is “terminated” or assigned on a subinterface with an expanded **encapsulation dot1q** command that specifies the two VLAN ID tags (outer VLAN ID and inner VLAN ID) terminated on the subinterface (see the figure below).

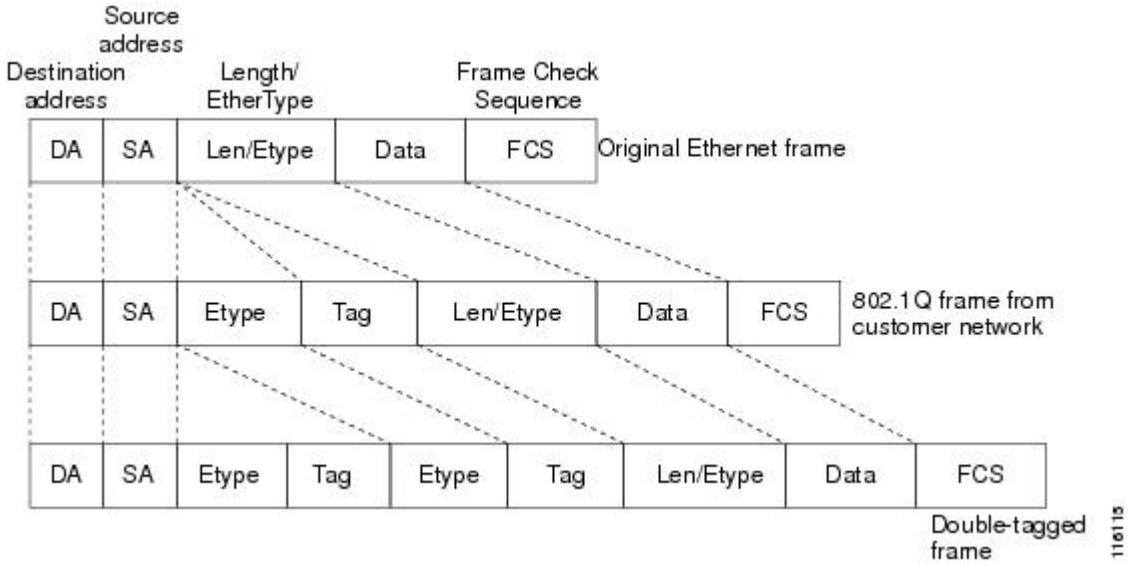
IEEE 802.1Q-in-Q VLAN Tag Termination is generally supported on whichever Cisco IOS XE features or protocols are supported on the subinterface. The only restriction is whether you assign ambiguous or unambiguous subinterfaces for the inner VLAN ID. See the Unambiguous and Ambiguous Subinterfaces section.

The primary benefit for the service provider is reduced number of VLANs supported for the same number of customers. Other benefits of this feature include:

- PPPoE scalability. By expanding the available VLAN space from 4096 to approximately 16.8 million (4096 times 4096), the number of PPPoE sessions that can be terminated on a given interface is multiplied.
- When deploying Gigabyte Ethernet DSL Access Multiplexer (DSLAM) in wholesale model, you can assign the inner VLAN ID to represent the end-customer virtual circuit (VC) and assign the outer VLAN ID to represent the service provider ID.

Whereas switches require IEEE 802.1Q tunnels on interfaces to carry double-tagged traffic, routers need only encapsulate Q-in-Q VLAN tags within another level of 802.1Q tags in order for the packets to arrive at the correct destination.

Figure 1: Untagged, 802.1Q-Tagged, and Double-Tagged Ethernet Frames



Unambiguous and Ambiguous Subinterfaces

The **encapsulation dot1q** command is used to configure Q-in-Q termination on a subinterface. The command accepts an Outer VLAN ID and one or more Inner VLAN IDs. The outer VLAN ID always has a specific value, while inner VLAN ID can either be a specific value or a range of values.

A subinterface that is configured with a single Inner VLAN ID is called an unambiguous Q-in-Q subinterface. In the following example, Q-in-Q traffic with an Outer VLAN ID of 101 and an Inner VLAN ID of 1001 is mapped to the Gigabit Ethernet 1/1/0.100 subinterface:

```
Device(config)# interface gigabitEehernet1/1/0.100
Device(config-subif)# encapsulation dot1q 101 second-dot1q 1001
```

A subinterface that is configured with multiple Inner VLAN IDs is called an ambiguous Q-in-Q subinterface. By allowing multiple Inner VLAN IDs to be grouped together, ambiguous Q-in-Q subinterfaces allow for a smaller configuration, improved memory usage and better scalability.

In the following example, Q-in-Q traffic with an Outer VLAN ID of 101 and Inner VLAN IDs anywhere in the 2001-2100 and 3001-3100 range is mapped to the Gigabit Ethernet 1/1/0.101 subinterface:

```
Device(config)# interface gigabitethernet1/1/0.101
Device(config-subif)# encapsulation dot1q 101 second-dot1q 2001-2100,3001-3100
```

Ambiguous subinterfaces can also use the **any**keyword to specify the inner VLAN ID.

See the Configuration Examples for IEEE 802.1Q-in-Q VLAN Tag Termination section for an example of how VLAN IDs are assigned to subinterfaces, and for a detailed example of how the **any** keyword is used on ambiguous subinterfaces.

Only PPPoE is supported on ambiguous subinterfaces. Standard IP routing is not supported on ambiguous subinterfaces.

IEEE802.1ad Support in Port-channels and Subinterfaces

This enhancement introduces IEEE802.1ad support on port-channel, port-channel subinterfaces, and port-channel member links, with EtherType 0x88a8, 0x9100, and 0x9200, in addition to the existing EtherType 0x8100.

Run the **dot1q tunneling ethertype xxxx** command to configure IEEE802.1ad on port-channel, port-channel subinterface, and port-channel member links.

This configuration is supported on the following platforms:

- Cisco ASR1006-X
- Cisco ASR1009-X: dual RP3, dual ESP 200, dual ESP 200-X, MIP100, and EPA (1X100GE, 10X10GE)
- Cisco ASR 1000 fixed routers

High availability is configured on ASR1006-X, ASR1009-X, and SSO/ISSU.

Restrictions for IEEE802.1ad Support in Port-channels and Subinterfaces

- This feature is not supported with SIP and SPA cards.
- Dot1q tunnel ethertype configuration is not supported in subinterfaces with high-availability configuration.

How to Configure IEEE 802.1Q-in-Q VLAN Tag Termination

Configuring the Interfaces for IEEE 802.1Q-in-Q VLAN Tag Termination

Perform this task to configure the main interface used for the Q-in-Q double tagging and to configure the subinterfaces. An optional step in this task shows you how to configure the EtherType field to be 0x9100 for the outer VLAN tag, if that is required. After the subinterface is defined, the 802.1Q encapsulation is configured to use the double tagging.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **dot1q tunneling ethertype** *ethertype*
5. **interface** *type number* **.** *subinterface-number*
6. **encapsulation dot1q** *vlan-id* **second-dot1q** {**any** | *vlan-id* | *vlan-id* - *vlan-id* [*vlan-id* - *vlan-id*]}

7. **pppoe enable** [**group** *group-name*] [**max-sessions** *max-sessions-number*]
8. **exit**
9. Repeat Step 5 to configure another subinterface.
10. Repeat Step 6 and Step 7 to specify the VLAN tags to be terminated on the subinterface.
11. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Device(config)# interface gigabitethernet 1/0/0`	Configures an interface and enters interface configuration mode.
Step 4	**dot1q tunneling ethertype** *ethertype* **Example:** `Device(config-if)# dot1q tunneling ethertype 0x9100`	(Optional) Defines the Ethertype field type used by peer devices when implementing Q-in-Q VLAN tagging.
Step 5	**interface** *type number* **.** *subinterface-number* **Example:** `Device(config-if)# interface gigabitethernet 1/0/0.1`	Configures a subinterface and enters subinterface configuration mode.
Step 6	**encapsulation dot1q** *vlan-id* **second-dot1q** {**any** | *vlan-id* | *vlan-id* - *vlan-id* [*vlan-id* - *vlan-id*]} **Example:** `Device(config-subif)# encapsulation dot1q 100 second-dot1q 200`	(Required) Enables the 802.1Q encapsulation of traffic on a specified subinterface in a VLAN. • Use the **second-dot1q** keyword and the *vlan-id*argument to specify the VLAN tags to be terminated on the subinterface. • In this example, an unambiguous Q-in-Q subinterface is configured because only one inner VLAN ID is specified.

	Command or Action	Purpose
		• Q-in-Q frames with an outer VLAN ID of 100 and an inner VLAN ID of 200 will be terminated.
Step 7	**pppoe enable** [**group** *group-name*] [**max-sessions** *max-sessions-number*] **Example:** `Device(config-subif)# pppoe enable group vpn1`	Enables PPPoE sessions on a subinterface. The example specifies that the PPPoE profile, vpn1, will be used by PPPoE sessions on the subinterface.
Step 8	**exit** **Example:** `Device(config-subif)# exit`	Exits subinterface configuration mode and returns to interface configuration mode. • Repeat this step one more time to exit interface configuration mode.
Step 9	Repeat Step 5 to configure another subinterface. **Example:** `Device(config-if)# interface gigabitethernet 1/0/0.2`	(Optional) Configures a subinterface and enters subinterface configuration mode.

Step 10	Repeat Step 6 and Step 7 to specify the VLAN tags to be terminated on the subinterface.
	Example:
	`Device(config-subif)# encapsulation dot1q 100 second-dot1q 100-199,201-600`
	Example:
	`Device(config-subif)# pppoe enable group vpn1`
		Step 6 enables the 802.1Q encapsulation of traffic on a specified subinterface in a VLAN.
		• Use the **second-dot1q** keyword and the *vlan-id*argument to specify the VLAN tags to be terminated on the subinterface.
		• In the example, an ambiguous Q-in-Q subinterface is configured because a range of inner VLAN IDs is specified.
		• Q-in-Q frames with an outer VLAN ID of 100 and an inner VLAN ID in the range of 100 to 199 or 201 to 600 will be terminated.
		Step 7 enables PPPoE sessions on the subinterface. The example specifies that the PPPoE profile, vpn1, will be used by PPPoE sessions on the subinterface.

Step 11	**end** **Example:** `Device(config-subif)# end`	Exits subinterface configuration mode and returns to privileged EXEC mode.

Verifying the IEEE 802.1Q-in-Q VLAN Tag Termination

Perform this optional task to verify the configuration of the IEEE 802.1Q-in-Q VLAN Tag Termination feature.

SUMMARY STEPS

1. **enable**
2. **show running-config**
3. **show vlans dot1q** [**internal** *interface-type interface-number* **.***subinterface-number*[**detail**] | **second-dot1q** *inner-id* **any**]] [**detail]**

DETAILED STEPS

Procedure

Step 1 **enable**

Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Device> enable
```

Step 2 **show running-config**

Use this command to show the currently running configuration on the device. You can use delimiting characters to display only the relevant parts of the configuration.

Example:

```
Device# show running-config
```

Step 3 **show vlans dot1q** [**internal** *interface-type interface-number* **.***subinterface-number*[**detail**] | **second-dot1q** *inner-id* **any**]] [**detail]**

Use this command to show the statistics for all the 802.1Q VLAN IDs. In this example, only the outer VLAN ID is displayed.

Example:

```
Router# show vlans dot1q

Total statistics for 802.1Q VLAN 1:
   441 packets, 85825 bytes input
   1028 packets, 69082 bytes output
Total statistics for 802.1Q VLAN 101:
   5173 packets, 510384 bytes input
   3042 packets, 369567 bytes output
Total statistics for 802.1Q VLAN 201:
   1012 packets, 119254 bytes input
   1018 packets, 120393 bytes output
Total statistics for 802.1Q VLAN 301:
   3163 packets, 265272 bytes input
   1011 packets, 120750 bytes output
Total statistics for 802.1Q VLAN 401:
   1012 packets, 119254 bytes input
   1010 packets, 119108 bytes output
```

Configuring IEEE 802.1ad in Port-channels and Subinterfaces

Perform this task to configure IEEE802.1ad in port-channels, port-channel subinterfaces, and port-channel member links.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **dot1q tunneling ethertype** {*0x88A8* | *0x9100* | *0x9200*}
5. **interface** *type number* **.** *subinterface-number*
6. **encapsulation dot1q** *vlan-id* **second-dot1q** {**any** | *vlan-id* | *vlan-id* **-** *vlan-id* [*vlan-id* **-** *vlan-id*]}
7. **ip address** *ip-address*
8. **pppoe enable** [**group** *group name*]
9. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Device(config)# interface port-channel 12`	Configures an interface and enters interface configuration mode.
Step 4	**dot1q tunneling ethertype** {*0x88A8* | *0x9100* | *0x9200*} **Example:** `Device(config-if)# dot1q tunneling ethertype 0x9100`	Defines the EtherType field used by peer devices when implementing Q-in-Q VLAN tagging.
Step 5	**interface** *type number* **.** *subinterface-number* **Example:** `Device(config-if)# interface port-channel 12.1`	Configures a subinterface and enters subinterface configuration mode.
Step 6	**encapsulation dot1q** *vlan-id* **second-dot1q** {**any** | *vlan-id* | *vlan-id* **-** *vlan-id* [*vlan-id* **-** *vlan-id*]}	Enables the 802.1Q encapsulation of traffic on a specified subinterface in a VLAN.

	Command or Action	Purpose
	Example: `Device(config-subif)# encapsulation dot1q 100 second-dot1q 200`	**Note** • Use the **second-dot1q** keyword and the *vlan-id* argument to specify the VLAN tags to be terminated on the subinterface. • In this example, an unambiguous Q-in-Q subinterface is configured because only one inner VLAN ID is specified. • Q-in-Q frames with an outer VLAN ID of 100 and an inner VLAN ID of 200 will be terminated.
Step 7	**ip address** *ip-address* **Example:** `Device(config-subif)# ip address 192.0.2.1.255.255.255.0`	Defines an IP address on the interface.
Step 8	**pppoe enable** [**group** *group name*] **Example:** `Device(config-subif)# pppoe enable group 2`	Enables PPPoE sessions on the subinterface.
Step 9	**end** **Example:** `Device(config-subif)# end`	Exits subinterface configuration mode and returns to privileged EXEC mode.

Configuration Examples for IEEE 802.1Q-in-Q VLAN Tag Termination

Configuring any Keyword on Subinterfaces for IEEE 802.1Q-in-Q VLAN Tag Termination Example

Some ambiguous subinterfaces can use the **any** keyword for the inner VLAN ID specification. The **any** keyword represents any inner VLAN ID that is not explicitly configured on any other interface. In the following example, seven subinterfaces are configured with various outer and inner VLAN IDs.

Note The **any** keyword can be configured on only one subinterface of a physical interface, outer VLAN ID, or a specified port-channel subinterface.

```
interface GigabitEthernet1/0/0.1
 encapsulation dot1q 100 second-dot1q 100
interface GigabitEthernet1/0/0.2
 encapsulation dot1q 100 second-dot1q 200
```

```
interface GigabitEthernet1/0/0.3
 encapsulation dot1q 100 second-dot1q 300-400,500-600
interface GigabitEthernet1/0/0.4
 encapsulation dot1q 100 second-dot1q any
interface GigabitEthernet1/0/0.5
 encapsulation dot1q 200 second-dot1q 50
interface GigabitEthernet1/0/0.6
 encapsulation dot1q 200 second-dot1q 1000-2000,3000-4000
interface GigabitEthernet1/0/0.7
 encapsulation dot1q 200 second-dot1q any
```

The table below shows which subinterfaces are mapped to different values of the outer and inner VLAN ID on Q-in-Q frames that come in on Gigabit Ethernet interface 1/0/0.

Table 1: Subinterfaces Mapped to Outer and Inner VLAN IDs for GE Interface 1/0/0

Outer VLAN ID	Inner VLAN ID	Subinterface mapped to
100	1 through 99	GigabitEthernet1/0/0.4
100	100	GigabitEthernet1/0/0.1
100	101 through 199	GigabitEthernet1/0/0.4
100	200	GigabitEthernet1/0/0.2
100	201 through 299	GigabitEthernet1/0/0.4
100	300 through 400	GigabitEthernet1/0/0.3
100	401 through 499	GigabitEthernet1/0/0.4
100	500 through 600	GigabitEthernet1/0/0.3
100	601 through 4095	GigabitEthernet1/0/0.4
200	1 through 49	GigabitEthernet1/0/0.7
200	50	GigabitEthernet1/0/0.5
200	51 through 999	GigabitEthernet1/0/0.7
200	1000 through 2000	GigabitEthernet1/0/0.6
200	2001 through 2999	GigabitEthernet1/0/0.7
200	3000 through 4000	GigabitEthernet1/0/0.6
200	4001 through 4095	GigabitEthernet1/0/0.7

A new subinterface is now configured:

```
interface GigabitEthernet1/0/0.8
 encapsulation dot1q 200 second-dot1q 200-600,900-999
```

The table below shows the changes made to the table for the outer VLAN ID of 200. Notice that subinterface 1/0/0.7 configured with the **any** keyword now has new inner VLAN ID mappings.

Table 2: Subinterfaces Mapped to Outer and Inner VLAN IDs for GE Interface 1/0/0--Changes Resulting from Configuring GE Subinterface 1/0/0.8

Outer VLAN ID	Inner VLAN ID	Subinterface mapped to
200	1 through 49	GigabitEthernet1/0/0.7
200	50	GigabitEthernet1/0/0.5
200	51 through 199	GigabitEthernet1/0/0.7
200	200 through 600	GigabitEthernet1/0/0.8
200	601 through 899	GigabitEthernet1/0/0.7
200	900 through 999	GigabitEthernet1/0/0.8
200	1000 through 2000	GigabitEthernet1/0/0.6
200	2001 through 2999	GigabitEthernet1/0/0.7
200	3000 through 4000	GigabitEthernet1/0/0.6
200	4001 through 4095	GigabitEthernet1/0/0.7

Additional References

The following sections provide references related to the IEEE 802.1Q-in-Q VLAN Tag Termination feature.

Related Documents

Related Topic	Document Title
Related commands	*Cisco IOS LAN Switching Command Reference*

Standards

Standards	Title
IEEE 802.1Q	--

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 IEEE 802.1Q-in-Q VLAN Tag Termination

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 3: Feature Information for IEEE 802.1Q-in-Q VLAN Tag Termination

Feature Name	Releases	Feature Information
IEEE 802.1Q-in-Q VLAN Tag Termination	Cisco IOS XE Release 2.1	This feature is introduced in the Cisco ASR 1000 Series Aggregation Services Routers. The following commands have been modified for this feature: **dot1q tunneling ethertype**, **encapsulation dot1q**, and **show vlans dot1q**
IEEE802.1ad Support in Port-channels and Subinterfaces	Cisco IOS XE Bengaluru 17.6	This feature is introduced in the Cisco ASR1006-X, Cisco ASR1009-X: dual RP3, dual ESP 200, dual ESP 200-X, MIP100 and EPA (1X100GE, 10X10GE), and Cisco ASR 1000 fixed routers. This feature allows IEEE802.1ad configuration on port-channels, port-channel subinterfaces, and member links with EtherTypes 0x88a8, 0x9100, and 0x9200.