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.

Feature History for the IEEE 802.1Q-in-Q VLAN Tag Termination Feature

Release	Modification
12.3(7)T	This feature was introduced.
12.3(7)XI1	This feature was integrated into Cisco IOS Release 12.3(7)XI1. Support for PPPoEoQ-in-Q, PPPoE packets that are double-tagged for Q-in-Q VLAN tag termination, was implemented on the Cisco 10000 series routers.
12.3(7)XI7	Support for IPoQ-in-Q, IP packets that are double-tagged for Q-in-Q VLAN tag termination, was added on the Cisco 10000 series routers.
12.2(28)ZV	This feature was integrated into Cisco IOS Release 12.2(28)ZV.
12.2(34)SB	This feature was integrated into Cisco IOS Release12.2(34)SB.

Finding Support Information for Platforms and Cisco IOS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS software image support. Access Cisco Feature Navigator at http://www.cisco.com/go/fn. You must have an account on Cisco.com. If you do not have an account or have forgotten your username or password, click **Cancel** at the login dialog box and follow the instructions that appear.

Contents

Prerequisites for IEEE 802.1Q-in-Q VLAN Tag Termination

- You have checked Feature Navigator to verify that your Cisco device and software image support this feature.
- You must be connected to an Ethernet device that supports double VLAN tag imposition/disposition or switching.

Restrictions for IEEE 802.1Q-in-Q VLAN Tag Termination

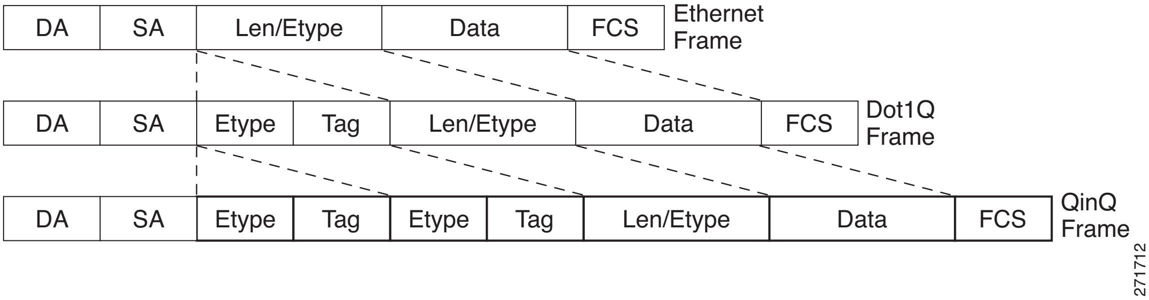
Cisco 10000 Series Router Restrictions

- Supported on Ethernet, FastEthernet, or Gigabit Ethernet interfaces.
- Supports only Point-to-Point Protocol over Ethernet (PPPoE) and IP packets that are double-tagged for Q-in-Q VLAN tag termination. Specifically PPPoEoQ-in-Q and IPoQ-in-Q are supported.
- PPPoEoQ-in-Q supports a maximum of 32,000 PPPoE sessions per interface for a maximum of 61,500 PPPoE sessions for the router. Note that the number of supported PPPoE sessions per interface can be limited by the cap of 32,000 sessions of any type that can run on the router.
- PPPoEoQ-in-Q supports 4094 outer VLAN IDs and 4094 inner VLAN IDs if only PPPoE is enabled and IP is not enabled on the subinterface.
- IPoQ-in-Q supports a maximum of 16,000 IPoQ-in-Q subinterfaces per interface.
- IPoQ-in-Q supports a maximum of 447 outer VLAN IDs and 4094 inner VLAN IDs.
- Multiprotocol Label Switching (MPLS) is **not** supported on PPPoEoQ-in-Q and IPoQ-in-Q subinterfaces.
- Layer 2 Ethernet over MPLS (EoMPLS) tunneling using the **xconnect** command on PPPoEoQ-in-Q and IPoQ-in-Q subinterfaces is *not* supported.
- Modular QoS CLI (MQC) can be applied to unambiguous subinterfaces only.
- Limited ACL support for PPPoEoQ-in-Q subinterfaces.

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

This Cisco IOS feature adds support for IP over IEEE 802.1QinQ (VLAN Tag stacking). QinQ encapsulation processing is an extension to 802.1q encapsulation processing. A QinQ frame looks like a VLAN 802.1Q frame, only it has two 802.1Q tags instead of one. Figure 1 demonstrates a QinQ frame.

Figure 1 QinQ frame



In releases prior to Cisco IOS Release 12.2(31)SB2, PRE2 supported PPPoEoQinQ encapsulation only. This capability has been retained and support is added for QinQ encapsulated IP traffic.

This section lists the concepts that the user should understand in order to perform the tasks in the "How to Configure IEEE 802.1Q-in-Q VLAN Tag Termination" section on page 7. The following concepts are described in this section:

- IEEE 802.1Q-in-Q VLAN Tag Termination on Subinterfaces, page 3
- Cisco 10000 Series Router Application, page 5
- Unambiguous and Ambiguous Subinterfaces, page 6

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 Figure 2 on page 4.

IEEE 802.1Q-in-Q VLAN Tag Termination is generally supported on whichever Cisco IOS features or protocols are supported on the subinterface. For example, if you can run PPPoE on the subinterface, you can configure a double-tagged frame for PPPoE. IPoQ-in-Q supports IP packets that are double-tagged for Q-in-Q VLAN tag termination by forwarding IP traffic with the double-tagged (also known as *stacked*) 802.1Q headers.

A primary consideration is whether you assign ambiguous or unambiguous subinterfaces for the inner VLAN ID. See the "Unambiguous and Ambiguous Subinterfaces" section on page 6.

For information on supported PPPoE sessions, number of supported inner and outer VLAN IDs, and general restrictions on the Cisco 10000 series routers, see Restrictions for IEEE 802.1Q-in-Q VLAN Tag Termination, page 2.

Note The Cisco 10000 series router supports PPPoEoQ-in-Q in Cisco IOS Release 12.3(7)XI1 and later, and IP over Q-in-Q (IPoQ-in-Q) in Cisco IOS Release 12.3(7)XI7 and later.

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:

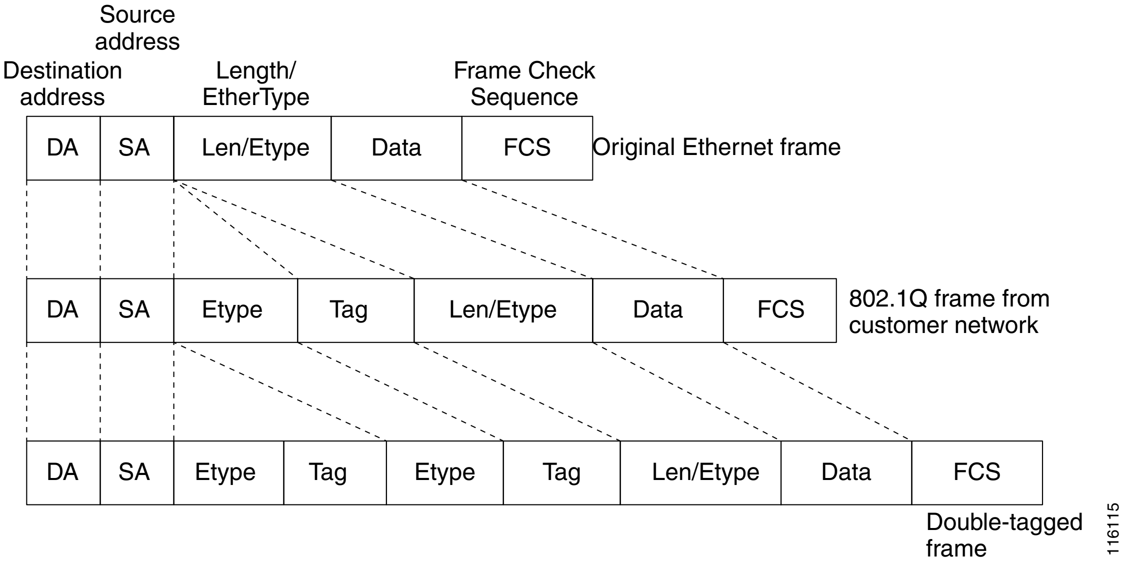
- PPPoE scalability. By expanding the available VLAN space from 4096 to approximately 16.8 million (4096 times 4096), the number of PPPoE sessions that can be terminated on a given interface is multiplied.

  Note The Cisco 10000 series router supports up to 32,000 PPPoE sessions per interface for a maximum of 61,500 PPPoE sessions for the router. These sessions may be over PPPoEoQ-in-Q ambiguous or unambiguous subinterfaces.

- When deploying Gigabyte Ethernet DSL Access Multiplexer (DSLAM) in a 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.

The Q-in-Q VLAN tag termination feature is simpler than the IEEE 802.1Q tunneling feature deployed for the Catalyst 6500 series switches or the Catalyst 3550 and Catalyst 3750 switches. 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.

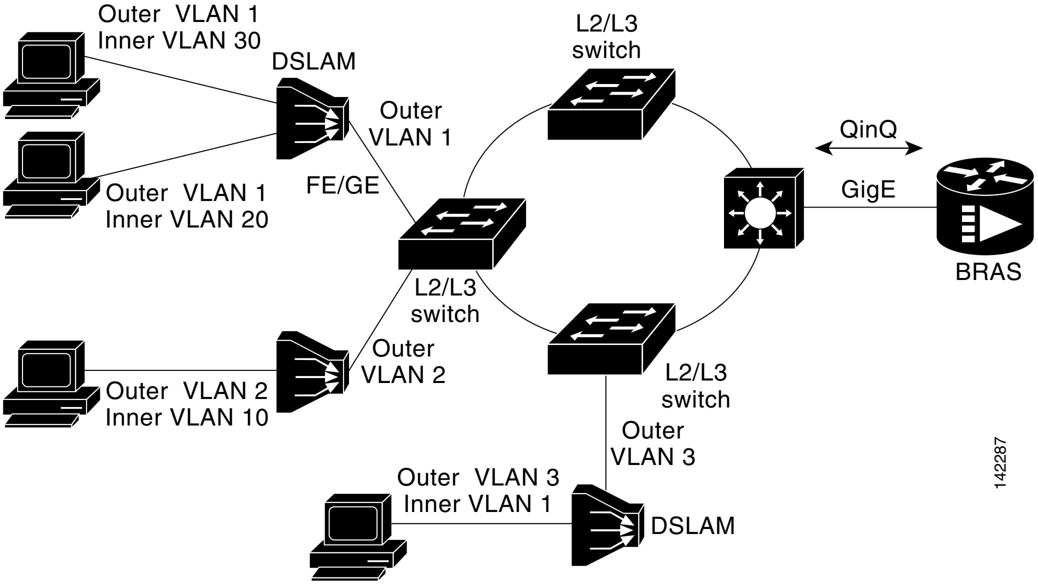
Figure 2 Untagged, 802.1Q-Tagged, and Double-Tagged Ethernet Frames



Cisco 10000 Series Router Application

For the emerging broadband Ethernet-based DSLAM market, the Cisco 10000 series router supports Q-in-Q encapsulation. With the Ethernet-based DSLAM model shown in Figure 3, customers typically get their own VLAN and all these VLANs are aggregated on a DSLAM.

Figure 3 Broadband Ethernet-based DSLAM Model of Q-in-Q VLANs



VLAN aggregation on a DSLAM will result in a lot of aggregate VLANs that at some point need to be terminated on the broadband remote access servers (BRAS). Although the model could connect the DSLAMs directly to the BRAS, a more common model uses the existing Ethernet-switched network where each DSLAM VLAN ID is tagged with a second tag (Q-in-Q) as it connects into the Ethernet-switched network.

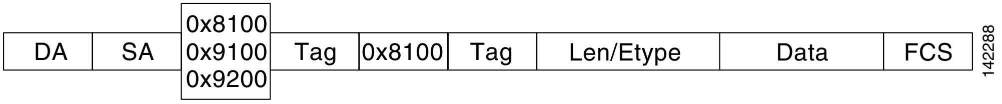
The Cisco 10000 series router supports PPPoEoQ-in-Q in Cisco IOS Release 12.3(7)XI1 and later, and IP over Q-in-Q (IPoQ-in-Q) in Cisco IOS Release 12.3(7)XI7 and later. Both PPPoE sessions and IP can be enabled on a subinterface. For information on supported PPPoE sessions, number of supported inner and outer VLAN IDs, and general restrictions on the Cisco 10000 series router, see the "Restrictions for IEEE 802.1Q-in-Q VLAN Tag Termination" section on page 2.

The PPPoEoQ-in-Q model is a PPP-terminated session.

PPPoQ-in-Q and IPoQ-in-Q encapsulation processing is an extension to 802.1Q encapsulation processing. A Q-in-Q frame looks like a VLAN 802.1Q frame, only it has two 802.1Q tags instead of one. See Figure 2 on page 4.

Q-in-Q encapsulation supports configurable outer tag Ethertype. The configurable Ethertype field values are 0x8100 (default), 0x9100, and 0x9200. See Figure 4.

Figure 4 Supported Configurable Ethertype Field Values



Security ACL Application on the Cisco 10000 Series Router

The IEEE 802.1Q-in-Q VLAN Tag Termination feature provides limited security access control list (ACL) support for PPPoEoQ-in-Q subinterfaces for the Cisco 10000 series router. There are no ACL restrictions on subinterfaces configured with IPoQ-in-Q.

If you apply an ACL to PPPoE traffic on a Q-in-Q subinterface in a VLAN, apply the ACL directly on the PPPoE session, using virtual access interfaces (VAIs) or RADIUS attribute 11 or 242.

You can apply ACLs to virtual access interfaces by configuring them under virtual template interfaces. You can also configure ACLs by using RADIUS attribute 11 or 242. When you use attribute 242, a maximum of 30,000 sessions can have ACLs.

ACLs that are applied to the VLAN Q-in-Q subinterface have no effect and are silently ignored. In the following example, ACL 1 that is applied to the VLAN Q-in-Q subinterface level will be ignored:

```
Router(config)# interface FastEthernet3/0/0.100
Router(config-subif)# encapsulation dot1q 100 second-dot1q 200
Router(config-subif)# ip access-group 1
```

Unambiguous and Ambiguous Subinterfaces

Note Only PPPoE is supported on ambiguous subinterfaces. Standard IP routing is not supported on 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 the 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 1/0.100 subinterface:

```
Router(config)# interface gigabitEehernet1/0.100
Router(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 1/0.101 subinterface.:

```
Router(config)# interface gigabitethernet1/0.101
Router(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 on page 11 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.

Note The **any** keyword in the **second-dot1q** command is not supported on a subinterface configured for IPoQ-in-Q because IP routing is not supported on ambiguous subinterfaces. Therefore, multiple values and ranges for the inner VLAN ID are not supported on IPoQ-in-Q.

Note On the Cisco 10000 series router, Modular QoS CLI (MQC) are only supported on unambiguous subinterfaces.

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

This section contains the following tasks:

- Configuring the Interfaces for IEEE 802.1Q-in-Q VLAN Tag Termination, page 7 (required)
- Verifying the IEEE 802.1Q-in-Q VLAN Tag Termination, page 9 (optional)

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.

Prerequisites

Cisco 10000 Series Router Prerequisites:

- PPPoE or IP is already configured.
- Virtual private dial-up network (VPDN) is enabled.

SUMMARY STEPS

Steps to configure Ethertype field for outer VLAN tag (Optional):

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **dot1q tunneling ethertype** *ethertype*

Steps to configure the Q-in-Q subinterface:

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 enabled [group *group-name*]

8. ip address *ip-address mask* [secondary]
9. Repeat Step 5 to configure another subinterface.
10. Repeat Step 6, Step 7, and Step 8, as required, to specify the VLAN tags to be terminated on the subinterface, to enable PPPoE sessions or IP on the subinterface.
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`** *`type number`*
	Example:
	`Router(config)# interface gigabitethernet 1/0/0`
		Configures an interface and enters interface configuration mode.

Step 4	**`dot1q tunneling ethertype`** *`ethertype`*
	Example:
	`Router(config-if)# dot1q tunneling ethertype 0x9100`
		(Optional) Defines the Ethertype field type used by peer devices when implementing Q-in-Q VLAN tagging.
		• Use this command if the Ethertype of peer devices is 0x9100 or 0x9200.
		• Cisco 10000 series router supports both the 0x9100 and 0x9200 Ethertype field types.

Step 5	**`interface`** *`type number.subinterface-number`*
	Example:
	`Router(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:
	`Router(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.
		• Q-in-Q frames with an outer VLAN ID of 100 and an inner VLAN ID of 200 will be terminated.

	Command or Action	Purpose

Step 7	**`pppoe enable`** `[`**`group`** *`group-name`*`]`
	Example:
	`Router(config-subif)# pppoe enable group vpn1`
		(Optional) This step is only required for PPPoEoQ-in-Q.
		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	**`ip address`** *`ip-address mask`* `[`**`secondary`**`]`
	Example:
	`Router(config-subif)# ip address 192.168.1.2 255.255.255.0`
		(Optional) This step is only required for IPoQ-in-Q.
		Sets a primary or secondary IP address for a subinterface. The example enables IP on the subinterface specified by the IP address, 192.168.1.2, and mask, 255.255.255.0.

Step 9	Repeat Step 5 to configure another subinterface. **Example:** `Router(config-if)# interface gigabitethernet 1/0/0.2`	(Optional) Configures a subinterface and enters subinterface configuration mode.

Step 10	Repeat Step 6, Step 7, and Step 8, as required, to specify the VLAN tags to be terminated on the subinterface.
	Example:
	`Router(config-subif)# encapsulation dot1q 100 second-dot1q 100-199,201-600`
	Example:
	`Router(config-subif)# pppoe enable group vpn1`
	Example:
	`Router(config-subif)# ip address 192.168.1.2 255.255.255.0`
		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 8 enables IP on a subinterface specified by the IP address and mask. The example enables IP on the subinterface specified by the IP address, 192.168.1.2, and mask, 255.255.255.0.
		Note Both PPPoE sessions and IP can be enabled on a subinterface.

Step 11	**`end`**
	Example:
	`Router(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**] | *outer-id* [*interface-type interface-number* | **second-dot1q** [*inner-id* | **any**]] [**detail**]]

DETAILED STEPS

Step 1 **enable**

Enables privileged EXEC mode. Enter your password if prompted.

```
Router> 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.

The following shows the currently running PPPoEoQ-in-Q and IPoQ-in-Q configurations on a Cisco 10000 series router:

```
Router# show running-config

.
.
.
interface FastEthernet0/0.201
 encapsulation dot1Q 201
 ip address 10.7.7.5 255.255.255.252
!
interface FastEthernet0/0.401
 encapsulation dot1Q 401
 ip address 10.7.7.13 255.255.255.252
!
interface FastEthernet0/0.201999
 encapsulation dot1Q 201 second-dot1q any
 pppoe enable
!
interface FastEthernet0/0.2012001
 encapsulation dot1Q 201 second-dot1q 2001
 ip address 10.8.8.9 255.255.255.252
!
interface FastEthernet0/0.2012002
 encapsulation dot1Q 201 second-dot1q 2002
 ip address 10.8.8.13 255.255.255.252
 pppoe enable
!
interface FastEthernet0/0.4019999
 encapsulation dot1Q 401 second-dot1q 100-900,1001-2000
 pppoe enable
!
interface GigabitEthernet5/0.101
 encapsulation dot1Q 101
 ip address 10.7.7.1 255.255.255.252
!
interface GigabitEthernet5/0.301
 encapsulation dot1Q 301
 ip address 10.7.7.9 255.255.255.252
!
```

```
interface GigabitEthernet5/0.301999
 encapsulation dot1Q 301 second-dot1q any
 pppoe enable
!
interface GigabitEthernet5/0.1011001
 encapsulation dot1Q 101 second-dot1q 1001
 ip address 10.8.8.1 255.255.255.252
!
interface GigabitEthernet5/0.1011002
 encapsulation dot1Q 101 second-dot1q 1002
 ip address 10.8.8.5 255.255.255.252
!
interface GigabitEthernet5/0.1019999
 encapsulation dot1Q 101 second-dot1q 1-1000,1003-2000
 pppoe enable

.
.
.
```

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

Note The **any** keyword is not supported on a subinterface configured for IPoQ-in-Q because IP routing is not supported on ambiguous subinterfaces.

```
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
```

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

This section contains the following example:

- Configuring any Keyword on Subinterfaces for IEEE 802.1Q-in-Q VLAN Tag Termination: Example, page 12

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 specified physical interface and outer VLAN ID.

Note The **any** keyword in the **second-dot1q** command is not supported on a subinterface configured for IPoQ-in-Q because IP routing is not supported on ambiguous subinterfaces. Therefore, multiple values and ranges for the inner VLAN ID are not supported on IPoQ-in-Q.

```
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
```

Table 1 shows which subinterfaces are mapped to different values of the outer and inner VLAN IDs 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

Table 1 Subinterfaces Mapped to Outer and Inner VLAN IDs for GE Interface 1/0/0 (continued)

Outer VLAN ID	Inner VLAN ID	Subinterface mapped to
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
```

Table 2 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
Interface commands: complete command syntax, command mode, defaults, usage guidelines, and examples	*Cisco IOS Interface and Hardware Component Command Reference*, *Release 12.3T*
Interface configuration examples	*Cisco IOS Interface and Hardware Component Configuration Guide*

Standards

Standards	Title
802.1Q	IEEE 802.1Q standard for local and metropolitan area networks.

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 Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Command Reference

This section documents the following commands.

- **dot1q tunneling ethertype**

- **encapsulation dot1q**
- **show vlans dot1q**

dot1q tunneling ethertype

To define the Ethertype field type used by peer devices when implementing Q-in-Q VLAN tagging, use the **dot1q tunneling ethertype** command in interface configuration mode. To remove the VLAN tag Ethertype, use the **no** form of this command.

dot1q tunneling ethertype {*0x8100* | *0x9100* | *0x9200*}

no dot1q tunneling ethertype {*0x8100* | *0x9100* | *0x9200*}

Syntax Description

ethertype	Type of Ethertype field.

Defaults

The Ethertype field used by peer devices when implementing Q-in-Q VLAN tagging is 0x8100.

Command Modes

Interface configuration

Command History

Release	Modification
12.3(7)T	This command was introduced.
12.3(7)XI1	This command was implemented on the Cisco 10000 series routers.
12.2(31)SB2	This command was integrated into Cisco IOS Release 12.2(31)SB2.
12.2(33)SRC	This command was integrated into Cisco IOS Release 12.2(33)SRC.

Usage Guidelines

Use the **dot1q tunneling ethertype** command if the peer switching devices are using an Ethertype field value of 0x9100 or 0x9200. All Cisco switching devices use the default Ethertype field value of 0x8100. The Cisco 10000 series router additionally supports the 0x9200 Ethertype field value.

Note On the Cisco 10000 series router, the Ethertype field for the outer VLAN ID can be changed, but the Ethertype field for the inner VLAN ID cannot be changed.

This command is used with the IEEE 802.1Q-in-Q VLAN Tag Termination feature in which double VLAN tagging is configured using the **encapsulation dot1q** command. 802.1Q double tagging allows a service provider to use a single VLAN to support customers who have multiple VLANs.

Examples

The following example shows how to configure an Ethertype field as 0x9100:

```
Router(config)# interface gigabitethernet 1/0/0
Router(config-if)# dot1q tunneling ethertype 0x9100
```

The following example shows how to configure an Ethertype field as 0x9200 on a Cisco 10000 series router:

```
Router(config)# interface gigabitethernet 1/0/0
Router(config-if)# dot1q tunneling ethertype 0x9200
```

Related Commands

Command	Description
encapsulation dot1q	Enables 802.1Q encapsulation of traffic on a specified subinterface or range of subinterfaces.
interface	Configures an interface and enters interface configuration mode.

encapsulation dot1q

To enable IEEE 802.1Q encapsulation of traffic on a specified subinterface in a VLAN, use the **encapsulation dot1q** command in interface range configuration mode or subinterface configuration mode. To disable IEEE 802.1Q encapsulation, use the **no** form of this command.

Interface Range Configuration Mode

encapsulation dot1q *vlan-id* [**native**]

no encapsulation dot1q

Subinterface Configuration Mode

encapsulation dot1q *vlan-id* **second-dot1q** {**any** | *vlan-id* | *vlan-id*-*vlan-id*[,*vlan-id*-*vlan-id*]}

no encapsulation dot1q *vlan-id* **second-dot1q** {**any** | *vlan-id* | *vlan-id*-*vlan-id*[,*vlan-id*-*vlan-id*]}

Syntax Description

vlan-id	Virtual LAN identifier. The allowed range is from 1 to 4094. For the IEEE 802.1Q-in-Q VLAN Tag Termination feature, the first instance of this argument defines the outer VLAN ID, and the second and subsequent instances define the inner VLAN ID.
native	(Optional) Sets the VLAN ID value of the port to the value specified by the *vlan-id* argument. **Note** This keyword is not supported by the IEEE 802.1Q-in-Q VLAN Tag Termination feature.
second-dot1q	Supports the IEEE 802.1Q-in-Q VLAN Tag Termination feature by allowing an inner VLAN ID to be configured.
any	Sets the inner VLAN ID value to a number that is not configured on any other subinterface. **Note** The **any** keyword in the **second-dot1q** command is not supported on a subinterface configured for IP over Q-in-Q (IPoQ-in-Q) because IP routing is not supported on ambiguous subinterfaces.
-	Separates the inner and outer VLAN ID values in the range to be defined. The hyphen is required.
,	Separates each VLAN ID range from the next range. The comma is required. Do not insert spaces between the values.

Defaults

IEEE 802.1Q encapsulation is disabled.

Command Modes

Interface range configuration
Subinterface configuration

Command History

Release	Modification
12.0(1)T	This command was introduced.
12.1(3)T	The **native** keyword was added.
12.2(2)DD	Support was added for this command in interface range configuration mode.
12.2(4)B	This command was integrated into Cisco IOS Release 12.2(4)B.
12.2(8)T	This command was integrated into Cisco IOS Release 12.2(8)T.
12.3(7)T	The **second-dot1q** keyword was added to support the IEEE 802.1Q-in-Q VLAN Tag Termination feature.
12.3(7)XI1	This command was integrated into Cisco IOS Release 12.3(7)XI and implemented on the Cisco 10000 series routers.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(31)SB2	This command was integrated into Cisco IOS Release 12.2(31)SB2.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.
12.2(33)SRC	This command was integrated into Cisco IOS Release 12.2(33)SRC.

Usage Guidelines

Interface Range Configuration Mode

IEEE 802.1Q encapsulation is configurable on Fast Ethernet interfaces. IEEE 802.1Q is a standard protocol for interconnecting multiple switches and routers and for defining VLAN topologies.

Use the **encapsulation dot1q** command in interface range configuration mode to apply a VLAN ID to each subinterface within the range specified by the **interface range** command. The VLAN ID specified by the *vlan-id* argument is applied to the first subinterface in the range. Each subsequent interface is assigned a VLAN ID, which is the specified *vlan-id* value plus the subinterface number minus the first subinterface number (VLAN ID + subinterface number – first subinterface number).

Note The Cisco 10000 series router does not support the **interface range** command nor the interface range configuration mode.

Do not configure encapsulation on the native VLAN of an IEEE 802.1Q trunk without using the **native** keyword. (Always use the **native** keyword when *vlan-id* is the ID of the IEEE 802.1Q native VLAN.)

Subinterface Configuration Mode

Use the **second-dot1q** keyword to configure the IEEE 802.1Q-in-Q VLAN Tag Termination feature. 802.1Q in 802.1Q (Q-in-Q) VLAN tag termination adds another layer of 802.1Q tag (called “metro tag” or “PE-VLAN”) to the 802.1Q tagged packets that enter the network. Double tagging expands the VLAN space, allowing service providers to offer certain services such as Internet access on specific VLANs for some customers and other types of services on other VLANs for other customers.

After a subinterface is defined, use the **encapsulation dot1q** command to add outer and inner VLAN ID tags to allow one VLAN to support multiple VLANs. You can assign a specific inner VLAN ID to the subinterface; that subinterface is unambiguous. Or you can assign a range or ranges of inner VLAN IDs to the subinterface; that subinterface is ambiguous.

Examples

The following example shows how to create the subinterfaces within the range 0.11 and 0.60 and apply VLAN ID 101 to the Fast Ethernet0/0.11 subinterface, VLAN ID 102 to Fast Ethernet0/0.12 (*vlan-id* = 101 + 12 – 11 = 102), and so on up to VLAN ID 150 to Fast Ethernet0/0.60 (*vlan-id* = 101 + 60 – 11 = 150):

```
Router(config)# interface range fastethernet0/0.11 - fastethernet0/0.60
Router(config-int-range)# encapsulation dot1q 101
```

The following example shows how to terminate a Q-in-Q frame on an unambiguous subinterface with an outer VLAN ID of 100 and an inner VLAN ID of 200:

```
Router(config)# interface gigabitethernet1/0/0.1
Router(config-subif)# encapsulation dot1q 100 second-dot1q 200
```

The following example shows how to terminate a Q-in-Q frame on an ambiguous subinterface with an outer VLAN ID of 100 and an inner VLAN ID in the range from 100 to 199 or from 201 to 600:

```
Router(config)# interface gigabitethernet1/0/0.1
Router(config-subif)# encapsulation dot1q 100 second-dot1q 100-199,201-600
```

Related Commands

Command	Description
encapsulation isl	Enables the ISL, which is a Cisco proprietary protocol for interconnecting multiple switches and maintaining VLAN information as traffic goes between switches.
encapsulation sde	Enables IEEE 802.10 encapsulation of traffic on a specified subinterface in VLANs.
interface range	Specifies multiple subinterfaces on which subsequent commands are executed at the same time.
show vlans dot1q	Displays information about 802.1Q VLAN subinterfaces.

show vlans dot1q

To display statistics about 802.1Q VLAN subinterfaces, use the **show vlans dot1q** command in privileged EXEC mode.

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

Syntax Description

internal	(Optional) Displays internal QinQ VLAN tag termination information. Used for troubleshooting purposes. The QinQ VLAN Tag Termination feature on the subinterface level preserves VLAN IDs and keeps traffic in different customer VLANs segregated.
interface-type	(Optional) Interface type.
interface-number	(Optional) Interface number.
.*subinterface-number*	(Optional) Subinterface number in the range 1 to 4294967293. A period (**.**) must be entered between the *interface-number* argument and the *subinterface-number* argument.
detail	(Optional) Displays detailed information.
outer-id	(Optional) Outer VLAN identifier. The allowed range is from 1 to 4095.
second-dot1q	(Optional) Displays inner VLAN subinterface information.
inner-id	(Optional) Inner VLAN identifier. The allowed range is from 1 to 4095.
any	(Optional) Displays information for all the inner VLAN subinterfaces configured as "any." **Note** The **any** keyword is not supported on a subinterface configured for IPoQinQ because IP routing is not supported on ambiguous subinterfaces.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.3(7)T	This command was introduced.
12.3(7)XI7	This command was integrated into Cisco IOS Release 12.3(7)XI7 and implemented on the Cisco 10000 series routers.
12.2(31)SB2	This command was integrated into Cisco IOS Release 12.2(31)SB2.
12.2(33)SRC	This command was integrated into Cisco IOS Release 12.2(33)SRC.

Usage Guidelines

If no arguments or keywords are entered, statistics for all of the 802.1Q VLAN IDs are displayed.

The **any** keyword is not supported for IPoQinQ because IP routing is not supported on ambiguous subinterfaces. However, the **second-dot1q** *inner-id* keyword and argument can be used on IPoQinQ for a specific inner VLAN ID that is not an ambiguous subinterface.

Note On the Cisco 10000 series router, the following is an implementation limitation—when a service policy is applied to a PPPoEoQinQ or IPoQinQ subinterface and the service policy drops some packets, the packets dropped are still displayed in the outgoing packet counters as output.

Examples

The output from the **show vlans dot1q** command displays the statistics for all the 802.1Q VLAN IDs. Only the outer VLAN IDs are displayed here.

```
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
```

Table 3 describes the significant fields shown in the display.

Table 3 show vlans dot1q Field Descriptions

Field	Description
Total statistics for 802.1Q VLAN 1	Statistics are shown for the VLAN ID with the specified outer ID.
packets	Number of packets encapsulated by the 802.1Q QinQ VLAN.
bytes input	Number of bytes input.
bytes output	Number of bytes output.

The following sample output from the **show vlans dot1q** command displays the statistics for the 802.1Q VLAN subinterface configured on Gigabit Ethernet interface 5/0:

```
Router# show vlans dot1q GigabitEthernet 5/0.1011001

GigabitEthernet5/0.1011001 (101/1001)
   1005 packets, 122556 bytes input
   1023 packets, 125136 bytes output
```

Table 4 describes the significant fields shown in the display.

Table 4 show vlans dot1q (subinterface) Field Descriptions

Field	Description
GigabitEthernet5/0.1011001 (101/1001)	Statistics are shown for subinterface Gigabit Ethernet 5/0.1011001 with an outer VLAN ID of 101 and an inner VLAN ID of 1001.
packets	Number of packets encapsulated by the 802.1Q QinQ VLAN.
bytes input	Number of bytes input.
bytes output	Number of bytes output.

The following sample output from the **show vlans dot1q** command displays the summary statistics for all of the VLAN subinterfaces under the physical interface Gigabit Ethernet 5/0 that have an outer VLAN ID of 101:

```
Router# show vlans dot1q 101 GigabitEthernet 5/0

Total statistics for 802.1Q VLAN 101 on GigabitEthernet5/0:
   5218 packets, 513444 bytes input
   3042 packets, 369567 bytes output
```

The following sample output from the **show vlans dot1q** command displays the individual subinterface statistics and summary statistics for all the VLAN subinterfaces under the physical interface Gigabit Ethernet 5/0 that have an outer VLAN ID of 101:

```
Router# show vlans dot1q 101 GigabitEthernet 5/0 detail

GigabitEthernet5/0.101 (0)
   3220 packets, 269148 bytes input
   1008 packets, 119622 bytes output
GigabitEthernet5/0.1019999 (101/1-1000,1003-2000)
   0 packets, 0 bytes input
   3 packets, 1143 bytes output
GigabitEthernet5/0.1011001 (101/1001)
   1005 packets, 122556 bytes input
   1023 packets, 125136 bytes output
GigabitEthernet5/0.1011002 (101/1002)
   1005 packets, 122556 bytes input
   1008 packets, 123666 bytes output
Total statistics for 802.1Q VLAN 101 on GigabitEthernet5/0:
   5230 packets, 514260 bytes input
   3042 packets, 369567 bytes output
```

The following sample output from the **show vlans dot1q** command displays the statistics for an outer VLAN and inner VLAN ID combination. This is a summary that displays the total for all the subinterfaces on the router that are configured with the specified IDs.

Note When multiple inner VLANs are used, the statistics displayed are at subinterface-level granularity, not VLAN-ID granularity. For example, when a range of inner VLAN IDs is assigned to a subinterface, the statistics are reported only at the subinterface level. Statistics are not available for each inner VLAN ID.

```
Router# show vlans dot1q 101 second-dot1q 1001 detail

GigabitEthernet5/0.1011001 (101/1001)
   1005 packets, 122556 bytes input
   1023 packets, 125136 bytes output
```

```
Total statistics for Outer/Inner VLAN 101/1001:
   1005 packets, 122556 bytes input
   1023 packets, 125136 bytes output
```

The following sample output from the **show vlans dot1q** command displays the statistics for a specific outer VLAN ID of 301 and an inner VLAN ID of any. This is a summary that displays the total for all of the subinterfaces on the router that are configured with the specified IDs.

```
Router# show vlans dot1q 301 second-dot1q any

GigabitEthernet5/0.301999 (301/any)
   0 packets, 0 bytes input
   3 packets, 1128 bytes output
Total statistics for Outer/Inner VLAN 301/"any":
   0 packets, 0 bytes input
   3 packets, 1128 bytes output
```

The following sample output from the **show vlans dot1q** command displays some internal information about the QinQ subsystem and is used for troubleshooting purposes (typically by Cisco engineers):

```
Router# show vlans dot1q internal

Internal VLAN representation on FastEthernet0/0:
VLAN Id: 1    (.1Q, Fa0/0)
VLAN Id: 201  (.1Q-in-.1Q tree, 3 elements)
  Inner VLAN Id: (0   -0   ) Fa0/0.201
  dot1q software subblock bitlist missing
  Inner VLAN Id: (2001-2001) Fa0/0.2012001
  2001
  Inner VLAN Id: (2002-2002) Fa0/0.2012002
  2002
  "any" Fa0/0.201999
VLAN Id: 401  (.1Q-in-.1Q tree, 3 elements)
  Inner VLAN Id: (0   -0   ) Fa0/0.401
  dot1q software subblock bitlist missing
  Inner VLAN Id: (100 -900 ) Fa0/0.4019999
  100-900,1001-2000
  Inner VLAN Id: (1001-2000) Fa0/0.4019999
  100-900,1001-2000
Internal VLAN representation on GigabitEthernet5/0:
VLAN Id: 1    (.1Q, Gi5/0)
VLAN Id: 101  (.1Q-in-.1Q tree, 5 elements)
  Inner VLAN Id: (0   -0   ) Gi5/0.101
  dot1q software subblock bitlist missing
  Inner VLAN Id: (1   -1000) Gi5/0.1019999
  1-1000,1003-2000
  Inner VLAN Id: (1001-1001) Gi5/0.1011001
  1001
  Inner VLAN Id: (1002-1002) Gi5/0.1011002
  1002
  Inner VLAN Id: (1003-2000) Gi5/0.1019999
  1-1000,1003-2000
VLAN Id: 301  (.1Q-in-.1Q tree, 1 elements)
  Inner VLAN Id: (0   -0   ) Gi5/0.301
  dot1q software subblock bitlist missing
  "any" Gi5/0.301999
```

Related Commands

Command	Description
encapsulation dot1q	Enables IEEE 802.1Q encapsulation of traffic on a specified subinterface in a VLAN.
vlan (VLAN configuration mode)	Configures a specific VLAN.
vlan database	Enters VLAN configuration mode.