



## CHAPTER 22

# Configuring 802.1Q Tunneling

---

Virtual private networks (VPNs) provide enterprise-scale connectivity on a shared infrastructure, often Ethernet-based, with the same security, prioritization, reliability, and manageability requirements of private networks. Tunneling is a feature designed for service providers who carry traffic of multiple customers across their networks and who are required to maintain the VLAN configurations of each customer without impacting the traffic of other customers. The Catalyst 4500 series switch supports IEEE 802.1Q tunneling.

**Note**

---

Be aware that 802.1Q requires WS-C4948, WS-C4948-10GE, ME-4924-10GE, WS-C4928-10GE, WS-C4900M, WS-X4013+10GE, WS-X4516, or WS-X4516-10GE.

---

This chapter contains these sections:

- [About 802.1Q Tunneling, page 22-2](#)
- [Configuring 802.1Q Tunneling, page 22-3](#)
- [Monitoring and Maintaining Tunneling Status, page 22-7](#)
- [Configuring Layer 2 Protocol Tunneling, page 22-9](#)
- [Monitoring and Maintaining Tunneling Status, page 22-7](#)

**Note**

---

For complete syntax and usage information for the switch commands used in this chapter, look at the *Cisco Catalyst 4500 Series Switch Command Reference* and related publications at this location:

<http://www.cisco.com/en/US/products/hw/switches/ps4324/index.html>

If the command is not found in the Catalyst 4500 Command Reference, it is located in the larger Cisco IOS library. Refer to the *Catalyst 4500 Series Switch Cisco IOS Command Reference* and related publications at this location:

<http://www.cisco.com/en/US/products/ps6350/index.html>

---

## About 802.1Q Tunneling

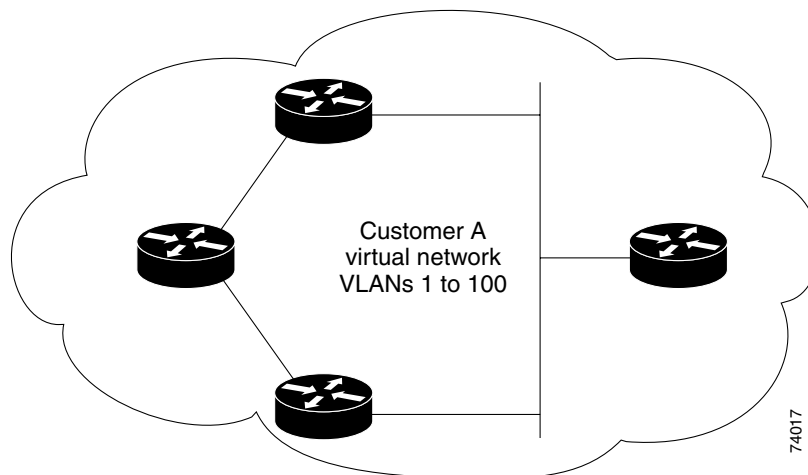
The VLAN ranges required by different customers in the same Service Provider network might overlap, and customer traffic through the infrastructure might be mixed. Assigning a unique range of VLAN IDs to each customer would restrict customer configurations and could easily exceed the VLAN limit (4096) of the 802.1Q specification.

802.1Q tunneling enables Service Providers to use a single VLAN to support customers who have multiple VLANs, while preserving customer VLAN IDs and keeping traffic in different customer VLANs segregated.

A port configured to support 802.1Q tunneling is called a tunnel port. When you configure tunneling, you assign a tunnel port to a VLAN ID that is dedicated to tunneling. Each customer requires a separate Service Provider VLAN ID, but that Service Provider VLAN ID supports VLANs of all the customers.

Customer traffic tagged in the normal way with appropriate VLAN IDs comes from an 802.1Q trunk port on the customer device and into a tunnel port on the Service Provider edge switch. The link between the customer device and the edge switch is asymmetric because one end is configured as an 802.1Q trunk port, and the other end is configured as a tunnel port. You assign the tunnel port interface to an access VLAN ID that is unique to each customer. See [Figure 22-1](#).

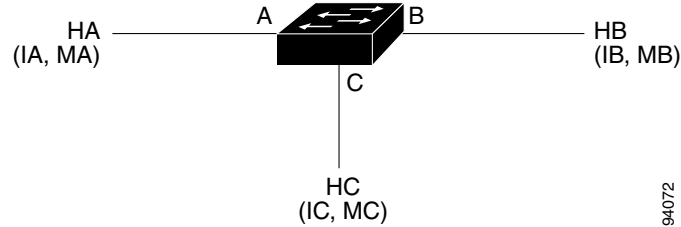
**Figure 22-1** 802.1Q Tunnel Ports in a Service Provider Network



Packets coming from the customer trunk port into the tunnel port on the Service Provider edge switch are normally 802.1Q-tagged with the appropriate VLAN ID. When the tagged packets exit the trunk port into the Service Provider network, they are encapsulated with another layer of an 802.1Q tag (called the *metro tag*) that contains the VLAN ID that is unique to the customer. The original customer 802.1Q tag is preserved in the encapsulated packet. Therefore, packets entering the Service Provider network are double-tagged, with the metro tag containing the customer's access VLAN ID, and the inner VLAN ID being that of the incoming traffic.

When the double-tagged packet enters another trunk port in a Service Provider core switch, the metro tag is stripped as the switch processes the packet. When the packet exits another trunk port on the same core switch, the same metro tag is again added to the packet. [Figure 22-2](#) shows the tag structures of the Ethernet packets starting with the original, or normal, frame.

**Figure 22-2 Original (Normal), 802.1Q, and Double-Tagged Ethernet Packet Formats**



When the packet enters the trunk port of the Service Provider egress switch, the metro tag is again stripped as the switch processes the packet. However, the metro tag is not added when the packet is sent out the tunnel port on the edge switch into the customer network. The packet is sent as a normal 802.1Q-tagged frame to preserve the original VLAN numbers in the customer network.

All packets entering the Service Provider network through a tunnel port on an edge switch are treated as untagged packets, whether they are untagged or already tagged with 802.1Q headers. The packets are encapsulated with the metro tag VLAN ID (set to the access VLAN of the tunnel port) when they are sent through the Service Provider network on an 802.1Q trunk port. The priority field on the metro tag is set to the interface class of service (CoS) priority configured on the tunnel port. (The default is zero if none is configured.)

In [Figure 22-1](#), Customer A was assigned VLAN 30, and Customer B was assigned VLAN 40. Packets entering the edge-switch tunnel ports with 802.1Q tags are double-tagged when they enter the Service Provider network, with the metro tag containing VLAN ID 30 or 40, appropriately, and the inner tag containing the original customer VLAN number, for example, VLAN 100. Even if Customers A and B both have VLAN 100 in their networks, the traffic remains segregated within the Service Provider network because the metro tag is different. Each customer controls its own VLAN numbering space, which is independent of the VLAN numbering space used by other customers and the VLAN numbering space used by the Service Provider network.

## Configuring 802.1Q Tunneling

These sections describe 802.1Q tunneling configuration:

- [802.1Q Tunneling Configuration Guidelines, page 22-3](#)
- [802.1Q Tunneling and Other Features, page 22-5](#)
- [Configuring an 802.1Q Tunneling Port, page 22-6](#)



### Note

By default, 802.1Q tunneling is disabled because the default switch port mode is dynamic auto. Tagging of 802.1Q native VLAN packets on all 802.1Q trunk ports is also disabled.

## 802.1Q Tunneling Configuration Guidelines

When you configure 802.1Q tunneling, you should always use asymmetrical links for traffic going through a tunnel and should dedicate one VLAN for each tunnel. You should also be aware of configuration requirements for native VLANs and maximum transmission units (MTUs). For more information about MTUs, see the [“System MTU” section on page 22-5](#).

## Native VLANs

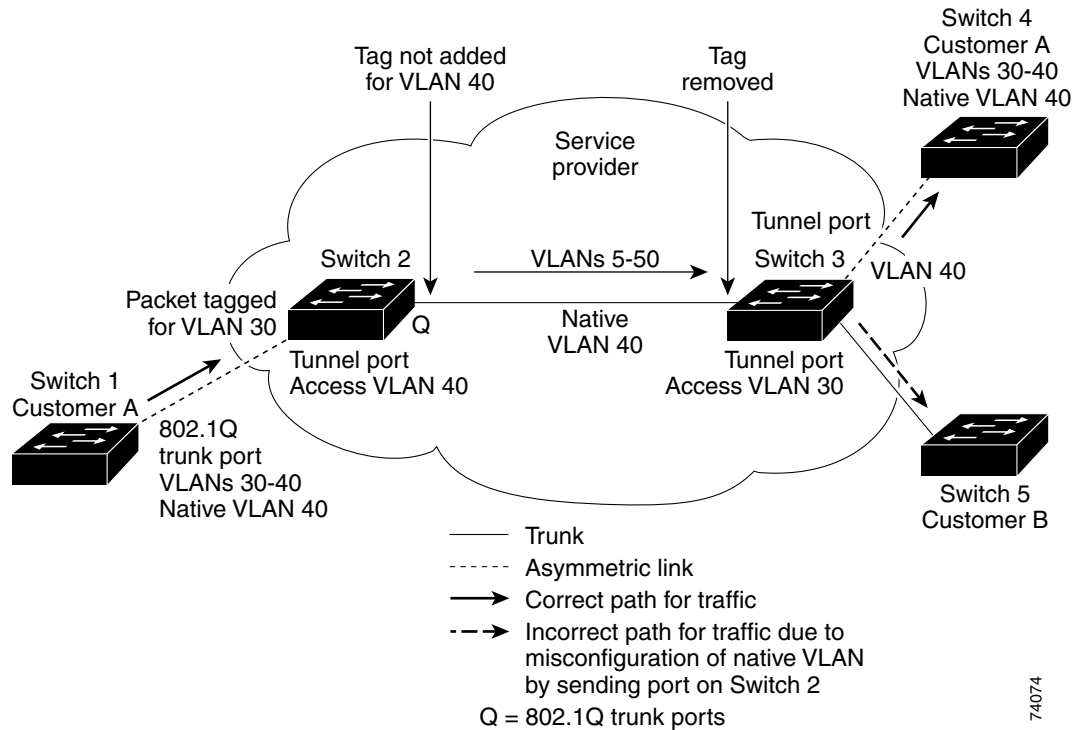
When configuring 802.1Q tunneling on an edge switch, you must use 802.1Q trunk ports for sending packets into the Service Provider network. However, packets going through the core of the Service Provider network can be carried through 802.1Q trunks, ISL trunks, or nontrunking links. When 802.1Q trunks are used in these core switches, the native VLANs of the 802.1Q trunks must not match any native VLAN of the nontrunking (tunneling) port on the same switch because traffic on the native VLAN would not be tagged on the 802.1Q sending trunk port.

See [Figure 22-3](#). VLAN 40 is configured as the native VLAN for the 802.1Q trunk port from Customer A at the ingress edge switch in the Service Provider network (Switch 2). Switch 1 of Customer A sends a tagged packet on VLAN 30 to the ingress tunnel port of Switch 2 in the Service Provider network, which belongs to access VLAN 40. Because the access VLAN of the tunnel port (VLAN 40) is the same as the native VLAN of the edge-switch trunk port (VLAN 40), the metro tag is not added to tagged packets received from the tunnel port. The packet carries only the VLAN 30 tag through the Service Provider network to the trunk port of the egress-edge switch (Switch 3) and is misdirected through the egress switch tunnel port to Customer B.

These are some ways to solve this problem:

- Use ISL trunks between core switches in the Service Provider network. Although customer interfaces connected to edge switches must be 802.1Q trunks, we recommend using ISL trunks for connecting switches in the core layer.
- Use the **switchport trunk native vlan tag** per-port command and the **vlan dot1q tag native** global configuration command to configure the edge switch so that all packets going out an 802.1Q trunk, including the native VLAN, are tagged. If the switch is configured to tag native VLAN packets on all 802.1Q trunks, the switch ensures that all packets exiting the trunk are tagged and prevents the reception of untagged packets on the trunk port.
- Ensure that the native VLAN ID on the edge-switch trunk port is not within the customer VLAN range. For example, if the trunk port carries traffic of VLANs 100 to 200, assign the native VLAN a number outside that range.

Figure 22-3 Potential Problem with 802.1Q Tunneling and Native VLANs



74074

## System MTU

The default system MTU for traffic on the Catalyst 4500 series switch is 1500 bytes. You can configure the switch to support larger frames by using the **system mtu** global configuration command. Because the 802.1Q tunneling feature increases the frame size by 4 bytes when the metro tag is added, you must configure all switches in the Service Provider network to be able to process larger frames by increasing the switch system MTU size to at least 1504 bytes. The maximum allowable system MTU for Catalyst 4500 Gigabit Ethernet switches is 9198 bytes; the maximum system MTU for Fast Ethernet switches is 1552 bytes.

## 802.1Q Tunneling and Other Features

Although 802.1Q tunneling works well for Layer 2 packet switching, there are incompatibilities between some Layer 2 features and Layer 3 switching.

- A tunnel port cannot be a routed port.
- IP routing is not supported on a VLAN that includes 802.1Q ports. Packets received from a tunnel port are forwarded based only on Layer 2 information. If routing is enabled on a switch virtual interface (SVI) that includes tunnel ports, untagged IP packets received from the tunnel port are recognized and routed by the switch. Customers can access the Internet through the native VLAN. If this access is not needed, you should not configure SVIs on VLANs that include tunnel ports.
- Tunnel ports do not support IP access control lists (ACLs).
- Layer 3 quality of service (QoS) ACLs and other QoS features related to Layer 3 information are not supported on tunnel ports. MAC-based QoS is supported on tunnel ports.

- EtherChannel port groups are compatible with tunnel ports as long as the 802.1Q configuration is consistent within an EtherChannel port group.
- Port Aggregation Protocol (PAgP), Link Aggregation Control Protocol (LACP), and UniDirectional Link Detection (UDLD) are supported on 802.1Q tunnel ports.
- Dynamic Trunking Protocol (DTP) is not compatible with 802.1Q tunneling because you must manually configure asymmetric links with tunnel ports and trunk ports.
- Loopback detection is supported on 802.1Q tunnel ports.
- When a port is configured as an 802.1Q tunnel port, spanning-tree bridge protocol data unit (BPDU) filtering is automatically enabled on the interface. Cisco Discovery Protocol (CDP) is automatically disabled on the interface.

## Configuring an 802.1Q Tunneling Port

To configure a port as an 802.1Q tunnel port, perform this task:

	Command	Purpose
Step 1	Switch# <b>configure terminal</b>	Enters global configuration mode.
Step 2	Switch(config)# <b>interface</b> <i>interface-id</i>	Enters interface configuration mode and the interface to be configured as a tunnel port. This should be the edge port in the Service Provider network that connects to the customer switch. Valid interfaces include physical interfaces and port-channel logical interfaces (port channels 1 to 64).
Step 3	Switch(config-if)# <b>switchport</b> <b>access vlan</b> <i>vlan-id</i>	Specifies the default VLAN, which is used if the interface stops trunking. This VLAN ID is specific to the particular customer.
Step 4	Switch(config-if)# <b>switchport mode</b> <b>dot1q-tunnel</b>	Sets the interface as an 802.1Q tunnel port.
Step 5	Switch(config-if)# <b>exit</b>	Returns to global configuration mode.
Step 6	Switch(config)# <b>vlan dot1q tag</b> <b>native</b>	(Optional) Sets the switch to enable tagging of native VLAN packets on all 802.1Q trunk ports. When not set, and a customer VLAN ID is the same as the native VLAN, the trunk port does not apply a metro tag, and packets could be sent to the wrong destination.
Step 7	Switch(config)# <b>end</b>	Returns to privileged EXEC mode.
Step 8	Switch# <b>show dot1q-tunnel</b>	Displays the tunnel ports on the switch.
Step 9	Switch# <b>show vlan dot1q tag native</b>	Displays 802.1Q native-VLAN tagging status.
Step 10	Switch# <b>copy running-config</b> <b>startup-config</b>	(Optional) Saves your entries in the configuration file.

Use the **no vlan dot1q tag native** global command and the **no switchport mode dot1q-tunnel** interface configuration command to return the port to the default state of dynamic auto. Use the **no vlan dot1q tag native** global configuration command to disable tagging of native VLAN packets.

This example shows how to configure an interface as a tunnel port, enable tagging of native VLAN packets, and verify the configuration. In this configuration, the VLAN ID for the customer connected to Gigabit Ethernet interface 2/7 is VLAN 22.

```
Switch(config)# interface gigabitethernet2/7
Switch(config-if)# switchport access vlan 22
% Access VLAN does not exist. Creating vlan 22
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# exit
Switch(config)# vlan dot1q tag native
Switch(config)# end
Switch# show dot1q-tunnel interface gigabitethernet2/7
Port
-----
LAN Port(s)
-----
Gi2/7
Switch# show vlan dot1q tag native
dot1q native vlan tagging is enabled globally
```

## Monitoring and Maintaining Tunneling Status

Table 22-1 shows the commands for monitoring and maintaining 802.1Q and Layer 2 protocol tunneling.

**Table 22-1** Commands for Monitoring and Maintaining Tunneling

Command	Purpose
Switch# <code>show dot1q-tunnel</code>	Displays 802.1Q tunnel ports on the switch.
Switch# <code>show dot1q-tunnel interface interface-id</code>	Verifies if a specific interface is a tunnel port.
Switch# <code>show errdisable recovery</code>	Verifies if the recovery timer from a Layer 2 protocol-tunnel error disable state is enabled.
Switch# <code>show vlan dot1q native</code>	Displays the status of native VLAN tagging on the switch.



### Note

With Cisco IOS Release 12.2(20)EW, the BPDU filtering configuration for both dot1q and Layer 2 protocol tunneling is no longer visible in the running configuration as "spanning-tree bpdupfilter enable." Instead, it is visible in the output of the `show spanning tree int detail` command as shown below.

```
Switch# show spann int f6/1 detail
Port 321 (FastEthernet6/1) of VLAN0001 is listening
Port path cost 19, Port priority 128, Port Identifier 128.321.
Designated root has priority 32768, address 0008.e341.4600
Designated bridge has priority 32768, address 0008.e341.4600
Designated port id is 128.321, designated path cost 0
Timers: message age 0, forward delay 2, hold 0
Number of transitions to forwarding state: 0
Link type is point-to-point by default
** Bpdu filter is enabled internally **
BPDU: sent 0, received 0
```

