

Contents

Configuring QinQ	1
About QinQ	1
QinQ benefits	1
How QinQ works	1
QinQ implementations	2
Protocols and standards	3
Restrictions and guidelines: QinQ configuration	3
Enabling QinQ	3
Configuring transmission for transparent VLANs	4
Configuring the TPID for VLAN tags	5
About TPID	5
Restrictions and guidelines	5
Configuring the TPID for CVLAN tags	6
Configuring the TPID for SVLAN tags	6
Setting the 802.1p priority in SVLAN tags	6
About the 802.1p priority in SVLAN tags	6
Prerequisites for setting the 802.1p priority in SVLAN tags	6
Tasks at a glance	7
Creating a traffic class and configuring CVLAN match criteria	7
Creating a traffic behavior and configuring a priority marking action for SVLAN tags	7
Creating a QoS policy	7
Applying the QoS policy	8
Display and maintenance commands for QinQ	8
QinQ configuration examples	8
Example: Configuring basic QinQ	8
Example: Configuring VLAN transparent transmission	10

Configuring QinQ

This document uses the following terms:

- **CVLAN**—Customer network VLANs, also called inner VLANs, refer to VLANs that a customer uses on the private network.
- **SVLAN**—Service provider network VLANs, also called outer VLANs, refer to VLANs that a service provider uses to transmit VLAN tagged traffic for customers.

About QinQ

802.1Q-in-802.1Q (QinQ) adds an 802.1Q tag to 802.1Q tagged customer traffic. It enables a service provider to extend Layer 2 connections across an Ethernet network between customer sites.

QinQ benefits

QinQ provides the following benefits:

- Enables a service provider to use a single SVLAN to convey multiple CVLANs for a customer.
- Enables customers to plan CVLANs without conflicting with SVLANs.
- Enables customers to keep their VLAN assignment schemes unchanged when the service provider changes its VLAN assignment scheme.
- Allows different customers to use overlapping CVLAN IDs. Devices in the service provider network make forwarding decisions based on SVLAN IDs instead of CVLAN IDs.

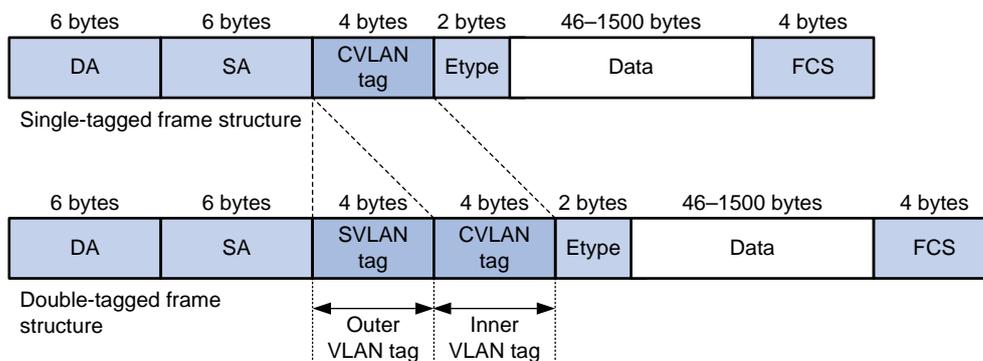
How QinQ works

As shown in [Figure 1](#), a QinQ frame transmitted over the service provider network carries the following tags:

- **CVLAN tag**—Identifies the VLAN to which the frame belongs when it is transmitted in the customer network.
- **SVLAN tag**—Identifies the VLAN to which the QinQ frame belongs when it is transmitted in the service provider network. The service provider allocates the SVLAN tag to the customer.

The devices in the service provider network forward a tagged frame according to its SVLAN tag only. The CVLAN tag is transmitted as part of the frame's payload.

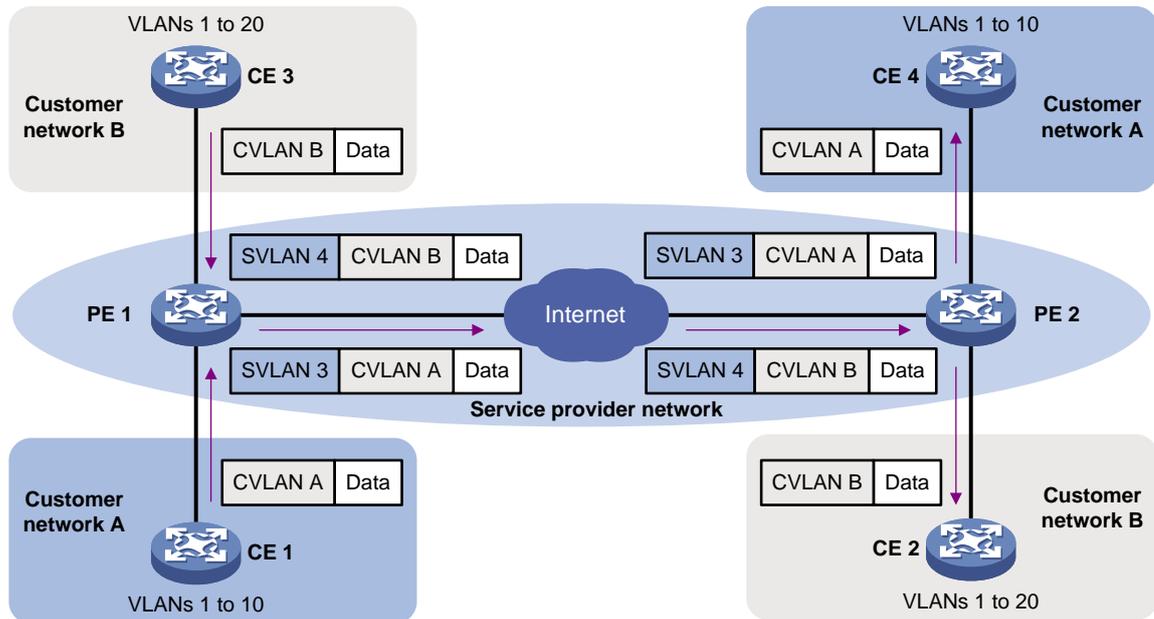
Figure 1 Single-tagged Ethernet frame header and double-tagged Ethernet frame header



As shown in [Figure 2](#), customer A has remote sites CE 1 and CE 4. Customer B has remote sites CE 2 and CE 3. The CVLANs of the two customers overlap. The service provider assigns SVLANs 3 and 4 to customers A and B, respectively.

When a tagged Ethernet frame from CE 1 arrives at PE 1, the PE tags the frame with SVLAN 3. The double-tagged Ethernet frame travels over the service provider network until it arrives at PE 2. PE 2 removes the SVLAN tag of the frame, and then sends the frame to CE 4.

Figure 2 Typical QinQ application scenario



QinQ implementations

QinQ is enabled on a per-port basis. The link type of a QinQ-enabled port can be access, hybrid, or trunk. The QinQ tagging behaviors are the same across these types of ports.

A QinQ-enabled port tags all incoming frames (tagged or untagged) with the PVID tag.

- If an incoming frame already has one tag, it becomes a double-tagged frame.
- If the frame does not have any 802.1Q tags, it becomes a frame tagged with the PVID.

QinQ provides the most basic VLAN manipulation method to tag all incoming frames (tagged or untagged) with the PVID tag. To perform advanced VLAN manipulations, use VLAN mappings or QoS policies as follows:

- To add different SVLANs for different CVLAN tags, use one-to-two VLAN mappings.
- To replace the SVLAN ID, CVLAN ID, or both IDs for an incoming double-tagged frame, use two-to-two VLAN mappings.
- To use criteria other than the CVLAN ID to match packets for SVLAN tagging, use the QoS nest action. The QoS nest action can also be used with other actions in the same traffic behavior.
- To set the 802.1p priority in SVLAN tags, use the priority marking action as described in "[Setting the 802.1p priority in SVLAN tags.](#)"

For more information about VLAN mappings, see "Configuring VLAN mapping." For more information about QoS, see *ACL and QoS Configuration Guide*.

Protocols and standards

- IEEE 802.1Q, *IEEE Standard for Local and Metropolitan Area Networks-Virtual Bridged Local Area Networks*
- IEEE 802.1ad, *IEEE Standard for Local and Metropolitan Area Networks-Virtual Bridged Local Area Networks-Amendment 4: Provider Bridges*

Restrictions and guidelines: QinQ configuration

When you configure QinQ, follow these restrictions and guidelines:

- The inner 802.1Q tag of QinQ frames is treated as part of the payload. As a best practice to ensure correct transmission of QinQ frames, set the MTU to a minimum of 1504 bytes for each port on their forwarding path. This value is the sum of the default Ethernet interface MTU (1500 bytes) and the length (4 bytes) of a VLAN tag.
- You can use a QoS policy, a VLAN mapping, and QinQ on a port for VLAN tag manipulation. If their settings conflict, the QoS policy has the highest priority, the VLAN mapping has the medium priority, and QinQ has the lowest priority.
- QinQ and two-to-two mappings are mutually exclusive. The device does not support adding an SVLAN tag on a QinQ-enabled port and then modifying the CVLAN and SVLAN IDs.

Enabling QinQ

About enabling QinQ

Enable QinQ on customer-side ports of PEs. A QinQ-enabled port tags an incoming frame with its PVID.

Restrictions and guidelines

Before you enable or disable QinQ on a port, you must remove any VLAN mappings on the port. For more information about VLAN mapping, see *Layer 2—LAN Switching Configuration Guide*.

Procedure

1. Enter system view.
system-view
2. Enter Layer 2 Ethernet interface view or Layer 2 aggregate interface view.
interface *interface-type interface-number*
3. Set the port link type.
port link-type { **access** | **hybrid** | **trunk** }
By default, the link type of a port is **access**.
4. Configure the port to allow packets from its PVID to pass through.
 - Assign the access port to the specified VLAN.
port access vlan *vlan-id*
By default, all access ports belong to VLAN 1.
The PVID of an access port is the VLAN to which the port belongs. The port sends packets from the VLAN untagged.
 - Configure the hybrid port to send packets from its PVID untagged.
port hybrid vlan *vlan-id-list* **untagged**
By default, the hybrid port is an untagged member of the VLAN to which the port belongs when its link type is access.

- Configure trunk port to allow packets from its PVID to pass through.
`port trunk permit vlan { vlan-id-list | all }`
 By default, a trunk port allows packets only from VLAN 1 to pass through.
- 5. Enable QinQ on the port.
`qinq enable`
 By default, QinQ is disabled on the port.

Configuring transmission for transparent VLANs

About transparent VLAN

You can exclude a VLAN (for example, the management VLAN) from the QinQ tagging action on a customer-side port. This VLAN is called a transparent VLAN.

Restrictions and guidelines

- Do not configure any other VLAN manipulation actions for the transparent VLAN on the port.
- Make sure all ports on the traffic path permit the transparent VLAN to pass through.
- If you use both transparent VLANs and VLAN mappings on an interface, the transparent VLANs cannot be the following VLANs:
 - Original or translated VLANs of one-to-one, one-to-two, and many-to-one VLAN mappings.
 - Original or translated outer VLANs of two-to-two VLAN mappings.

Procedure

1. Enter system view.
`system-view`
2. Enter Layer 2 Ethernet interface view or Layer 2 aggregate interface view.
`interface interface-type interface-number`
3. Set the port link type.
`port link-type { hybrid | trunk }`
 By default, the link type of a port is **access**.
4. Configure the port to allow packets from the transparent VLANs to pass through.
 - Configure the hybrid port to allow packets from the transparent VLANs to pass through.
`port hybrid vlan vlan-id-list { tagged | untagged }`
 By default, a hybrid port is an untagged member of the VLAN to which the port belongs when its link type is **access**.
 - Configure the trunk port to allow packets from the transparent VLANs to pass through.
`port trunk permit vlan { vlan-id-list | all }`
 By default, a trunk port allows packets only from VLAN 1 to pass through.
5. Specify transparent VLANs for the port.
`qinq transparent-vlan vlan-id-list`
 By default, transparent transmission is not configured for any VLANs.

Configuring the TPID for VLAN tags

About TPID

TPID identifies a frame as an 802.1Q tagged frame. The TPID value varies by vendor. On an H3C device, the TPID in the 802.1Q tag added on a QinQ-enabled port is 0x8100 by default, in compliance with IEEE 802.1Q. In a multi-vendor network, make sure the TPID setting is the same between directly connected devices so 802.1Q tagged frames can be identified correctly.

TPID settings include CVLAN TPID and SVLAN TPID.

A QinQ-enabled port uses the CVLAN TPID to match incoming tagged frames. An incoming frame is handled as untagged if its TPID is different from the CVLAN TPID.

SVLAN TPIDs are configurable on a per-port basis. A service provider-side port uses the SVLAN TPID to replace the TPID in outgoing frames' SVLAN tags and match incoming tagged frames. An incoming frame is handled as untagged if the TPID in its outer VLAN tag is different from the SVLAN TPID.

For example, a PE device is connected to a customer device that uses the TPID 0x8200 and to a provider device that uses the TPID 0x9100. For correct packet processing, you must set the CVLAN TPID and SVLAN TPID to 0x8200 and 0x9100 on the PE, respectively.

The TPID field is at the same position as the EtherType field in an untagged Ethernet frame. To ensure correct packet type identification, do not set the TPID value to any of the values listed in [Table 1](#).

Table 1 Reserved EtherType values

Protocol type	Value
ARP	0x0806
PUP	0x0200
RARP	0x8035
IP	0x0800
IPv6	0x86dd
PPPoE	0x8863/0x8864
MPLS	0x8847/0x8848
IPX/SPX	0x8137
IS-IS	0x8000
LACP	0x8809
LLDP	0x88cc
802.1X	0x888e
802.1ag	0x8902
Cluster	0x88a7
Reserved	0xfffd/0xfffe/0xffff

Restrictions and guidelines

The TPID value in CVLAN tags is typically configured on PEs. The TPID value in SVLAN tags is typically configured on the service provider-side ports of PEs.

Configuring the TPID for CVLAN tags

1. Enter system view.
system-view
2. Set the TPID for CVLAN tags.
qinq ethernet-type customer-tag *hex-value*
By default, the TPID is 0x8100 for CVLAN tags.

Configuring the TPID for SVLAN tags

1. Enter system view.
system-view
2. Enter Layer 2 Ethernet interface view or Layer 2 aggregate interface view.
interface *interface-type interface-number*
3. Set the TPID for SVLAN tags.
qinq ethernet-type service-tag *hex-value*
By default, the TPID is 0x8100 for SVLAN tags.

Setting the 802.1p priority in SVLAN tags

About the 802.1p priority in SVLAN tags

By default, the 802.1p priority in the SVLAN tag added by a QinQ-enabled port depends on the priority trust mode on the port.

- If the 802.1p priority in frames is trusted, the device copies the 802.1p priority in the CVLAN tag to the SVLAN tag.
- If port priority is trusted, the port priority (0 by default) is used as the 802.1p priority in the SVLAN tag.

You can configure a QoS policy to modify the 802.1p priority in SVLAN tags as follows:

- Modify the 802.1p priority in the SVLAN tag based on the 802.1p priority in the CVLAN tag or the CVLAN ID.
- Copy the 802.1p priority in the CVLAN tag to the SVLAN tag.

For more information about QoS policies and priority trust mode, see *ACL and QoS Configuration Guide*.

Prerequisites for setting the 802.1p priority in SVLAN tags

1. Enable QinQ. For more information, see "[Enabling QinQ](#)."
To use the CVLAN ID or 802.1p priority of the CVLAN tag to set the 802.1p priority of the SVLAN tag, you must first enable QinQ on the port.
2. Use the **qos trust dot1p** command to configure the port to trust the 802.1p priority in incoming frames. For more information, see *ACL and QoS Configuration Guide*.
This setting is required if the **remark dot1p** command is configured. It is optional if the **remark dot1p customer-dot1p-trust** command is configured.

Tasks at a glance

To use QoS policies to set the 802.1p priority in SVLAN tags, perform the following tasks:

1. [Creating a traffic class and configuring CVLAN match criteria](#)
2. [Creating a traffic behavior and configuring a priority marking action for SVLAN tags](#)
3. [Creating a QoS policy](#)
4. [Applying the QoS policy](#)

Creating a traffic class and configuring CVLAN match criteria

1. Enter system view.
system-view
2. Create a traffic class and enter its view.
traffic classifier *classifier-name* [**operator** { **and** | **or** }]
3. Configure CVLAN match criteria.
Choose one option as needed:
 - o Match CVLAN IDs.
if-match customer-vlan-id *vlan-id-list*
 - o Match 802.1p priority.
if-match customer-dot1p *dot1p-value*<1-8>

Creating a traffic behavior and configuring a priority marking action for SVLAN tags

1. Enter system view.
system-view
2. Create a traffic behavior and enter its view.
traffic behavior *behavior-name*
3. Configure a priority marking action for SVLAN tags.
Choose one option as needed:
 - o Replace the priority in the SVLAN tags of matching frames with the configured priority.
remark dot1p *dot1p-value*
 - o Copy the 802.1p priority in the CVLAN tag to the SVLAN tag.
remark dot1p customer-dot1p-trust

Creating a QoS policy

1. Enter system view.
system-view
2. Create a QoS policy and enter its view.
qos policy *policy-name*
3. Specify the traffic behavior for the traffic class in the QoS policy.
classifier *classifier-name* **behavior** *behavior-name*

Applying the QoS policy

1. Enter system view.
`system-view`
2. Enter Layer 2 Ethernet interface view.
`interface interface-type interface-number`
3. Apply the QoS policy to the inbound direction of the port.
`qos apply policy policy-name inbound`

Display and maintenance commands for QinQ

Execute `display` commands in any view.

Task	Command
Display QinQ-enabled ports.	<code>display qinq [interface interface-type interface-number]</code>

QinQ configuration examples

Example: Configuring basic QinQ

Network configuration

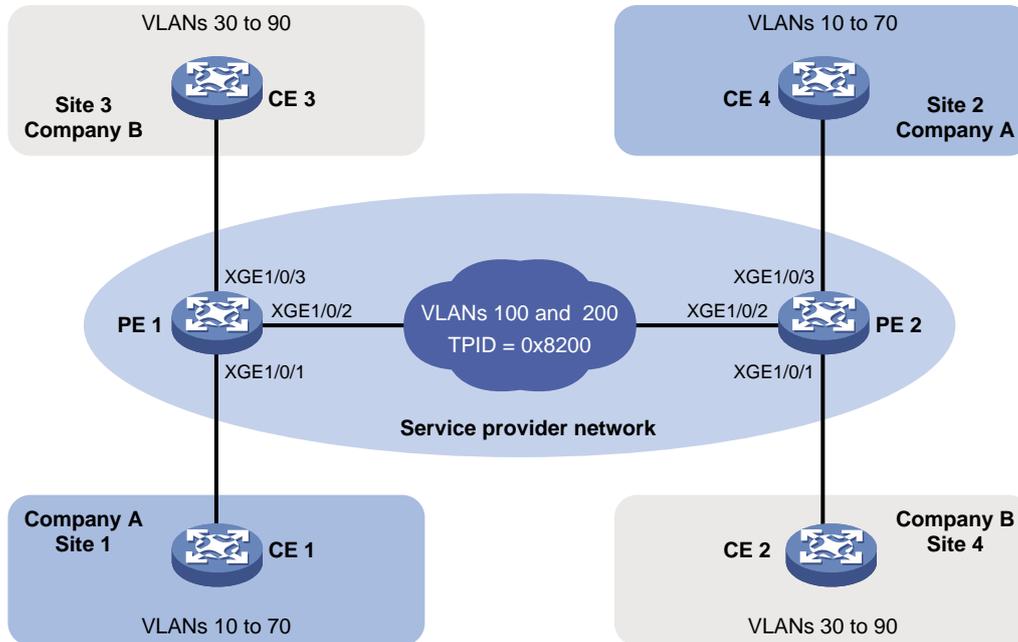
As shown in [Figure 3](#):

- The service provider assigns VLAN 100 to Company A's VLANs 10 through 70.
- The service provider assigns VLAN 200 to Company B's VLANs 30 through 90.
- The devices between PE 1 and PE 2 in the service provider network use a TPID value of 0x8200.

Configure QinQ on PE 1 and PE 2 to transmit traffic in VLANs 100 and 200 for Company A and Company B, respectively.

For the QinQ frames to be identified correctly, set the SVLAN TPID to 0x8200 on the service provider-side ports of PE 1 and PE 2.

Figure 3 Network diagram



Procedure

1. Configure PE 1:

Configure Ten-GigabitEthernet 1/0/1 as a trunk port, and assign it to VLAN 100.

```
<PE1> system-view
[PE1] interface ten-gigabitethernet 1/0/1
[PE1-Ten-GigabitEthernet1/0/1] port link-type trunk
[PE1-Ten-GigabitEthernet1/0/1] port trunk permit vlan 100
```

Set the PVID of Ten-GigabitEthernet 1/0/1 to VLAN 100.

```
[PE1-Ten-GigabitEthernet1/0/1] port trunk pvid vlan 100
```

Enable QinQ on Ten-GigabitEthernet 1/0/1.

```
[PE1-Ten-GigabitEthernet1/0/1] qinq enable
[PE1-Ten-GigabitEthernet1/0/1] quit
```

Configure Ten-GigabitEthernet 1/0/2 as a trunk port, and assign it to VLANs 100 and 200.

```
[PE1] interface ten-gigabitethernet 1/0/2
[PE1-Ten-GigabitEthernet1/0/2] port link-type trunk
[PE1-Ten-GigabitEthernet1/0/2] port trunk permit vlan 100 200
```

Set the TPID value in the SVLAN tags to 0x8200 on Ten-GigabitEthernet 1/0/2.

```
[PE1-Ten-GigabitEthernet1/0/2] qinq ethernet-type service-tag 8200
[PE1-Ten-GigabitEthernet1/0/2] quit
```

Configure Ten-GigabitEthernet 1/0/3 as a trunk port, and assign it to VLAN 200.

```
[PE1] interface ten-gigabitethernet 1/0/3
[PE1-Ten-GigabitEthernet1/0/3] port link-type trunk
[PE1-Ten-GigabitEthernet1/0/3] port trunk permit vlan 200
```

Set the PVID of Ten-GigabitEthernet 1/0/3 to VLAN 200.

```
[PE1-Ten-GigabitEthernet1/0/3] port trunk pvid vlan 200
```

Enable QinQ on Ten-GigabitEthernet 1/0/3.

```
[PE1-Ten-GigabitEthernet1/0/3] qinq enable
```

- ```
[PE1-Ten-GigabitEthernet1/0/3] quit
```
2. Configure PE 2:
    - # Configure Ten-GigabitEthernet 1/0/1 as a trunk port, and assign it to VLAN 200.
 

```
<PE2> system-view
[PE2] interface ten-gigabitethernet 1/0/1
[PE2-Ten-GigabitEthernet1/0/1] port link-type trunk
[PE2-Ten-GigabitEthernet1/0/1] port trunk permit vlan 200
```
    - # Set the PVID of Ten-GigabitEthernet 1/0/1 to VLAN 200.
 

```
[PE2-Ten-GigabitEthernet1/0/1] port trunk pvid vlan 200
```
    - # Enable QinQ on Ten-GigabitEthernet 1/0/1.
 

```
[PE2-Ten-GigabitEthernet1/0/1] qinq enable
[PE2-Ten-GigabitEthernet1/0/1] quit
```
    - # Configure Ten-GigabitEthernet 1/0/2 as a trunk port, and assign it to VLANs 100 and 200.
 

```
[PE2] interface ten-gigabitethernet 1/0/2
[PE2-Ten-GigabitEthernet1/0/2] port link-type trunk
[PE2-Ten-GigabitEthernet1/0/2] port trunk permit vlan 100 200
```
    - # Set the TPID value in the SVLAN tags to 0x8200 on Ten-GigabitEthernet 1/0/2.
 

```
[PE2-Ten-GigabitEthernet1/0/2] qinq ethernet-type service-tag 8200
[PE2-Ten-GigabitEthernet1/0/2] quit
```
    - # Configure Ten-GigabitEthernet 1/0/3 as a trunk port, and assign it to VLAN 100.
 

```
[PE2] interface ten-gigabitethernet 1/0/3
[PE2-Ten-GigabitEthernet1/0/3] port link-type trunk
[PE2-Ten-GigabitEthernet1/0/3] port trunk permit vlan 100
```
    - # Set the PVID of Ten-GigabitEthernet 1/0/3 to VLAN 100.
 

```
[PE2-Ten-GigabitEthernet1/0/3] port trunk pvid vlan 100
```
    - # Enable QinQ on Ten-GigabitEthernet 1/0/3.
 

```
[PE2-Ten-GigabitEthernet1/0/3] qinq enable
[PE2-Ten-GigabitEthernet1/0/3] quit
```
  3. Configure the devices between PE 1 and PE 2:
    - # Set the MTU to a minimum of 1504 bytes for each port on the path of QinQ frames. (Details not shown.)
    - # Configure all ports on the forwarding path to allow frames from VLANs 100 and 200 to pass through without removing the VLAN tag. (Details not shown.)

## Example: Configuring VLAN transparent transmission

### Network configuration

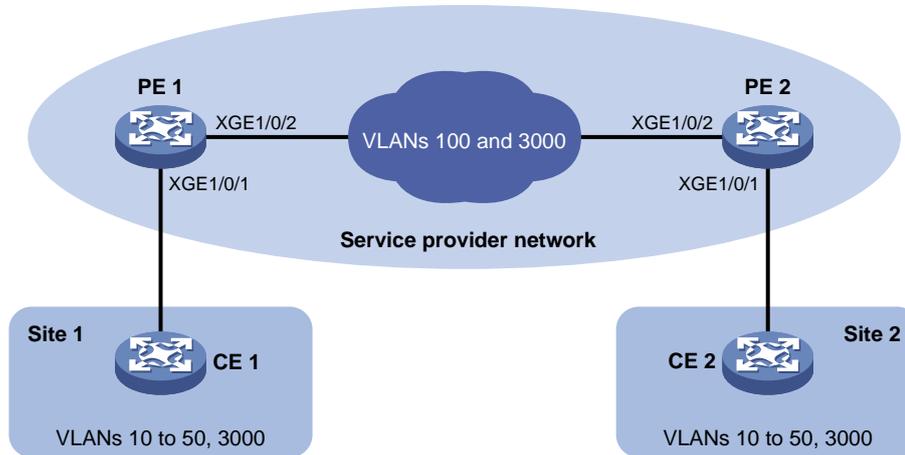
As shown in [Figure 4](#):

- The service provider assigns VLAN 100 to a company's VLANs 10 through 50.
- VLAN 3000 is the dedicated VLAN of the company on the service provider network.

Configure QinQ on PE 1 and PE 2 to provide Layer 2 connectivity for CVLANs 10 through 50 over the service provider network.

Configure VLAN transparent transmission for VLAN 3000 on PE 1 and PE 2 to enable the hosts in VLAN 3000 to communicate without using an SVLAN.

Figure 4 Network diagram



## Procedure

### 1. Configure PE 1:

# Configure Ten-GigabitEthernet 1/0/1 as a trunk port, and assign it to VLAN 100 and VLAN 3000.

```
<PE1> system-view
[PE1] interface ten-gigabitethernet 1/0/1
[PE1-Ten-GigabitEthernet1/0/1] port link-type trunk
[PE1-Ten-GigabitEthernet1/0/1] port trunk permit vlan 100 3000
```

# Set the PVID of Ten-GigabitEthernet 1/0/1 to VLAN 100.

```
[PE1-Ten-GigabitEthernet1/0/1] port trunk pvid vlan 100
```

# Enable QinQ on Ten-GigabitEthernet 1/0/1.

```
[PE1-Ten-GigabitEthernet1/0/1] qinq enable
```

# Enable transparent transmission for VLAN 3000 on Ten-GigabitEthernet 1/0/1.

```
[PE1-Ten-GigabitEthernet1/0/1] qinq transparent-vlan 3000
```

```
[PE1-Ten-GigabitEthernet1/0/1] quit
```

# Configure Ten-GigabitEthernet 1/0/2 as a trunk port, and assign it to VLANs 100 and 3000.

```
[PE1] interface ten-gigabitethernet 1/0/2
[PE1-Ten-GigabitEthernet1/0/2] port link-type trunk
[PE1-Ten-GigabitEthernet1/0/2] port trunk permit vlan 100 3000
[PE1-Ten-GigabitEthernet1/0/2] quit
```

### 2. Configure PE 2:

# Configure Ten-GigabitEthernet 1/0/1 as a trunk port, and assign it to VLAN 100 and VLAN 3000.

```
<PE2> system-view
[PE2] interface ten-gigabitethernet 1/0/1
[PE2-Ten-GigabitEthernet1/0/1] port link-type trunk
[PE2-Ten-GigabitEthernet1/0/1] port trunk permit vlan 100 3000
```

# Set the PVID of Ten-GigabitEthernet 1/0/1 to VLAN 100.

```
[PE1-Ten-GigabitEthernet1/0/1] port trunk pvid vlan 100
```

# Enable QinQ on Ten-GigabitEthernet 1/0/1.

```
[PE2-Ten-GigabitEthernet1/0/1] qinq enable
```

# Enable transparent transmission for VLAN 3000 on Ten-GigabitEthernet 1/0/1.

```
[PE2-Ten-GigabitEthernet1/0/1] qinq transparent-vlan 3000
```

```
[PE2-Ten-GigabitEthernet1/0/1] quit
```

**# Configure Ten-GigabitEthernet 1/0/2 as a trunk port, and assign it to VLANs 100 and 3000.**

```
[PE2] interface ten-gigabitethernet 1/0/2
```

```
[PE2-Ten-GigabitEthernet1/0/2] port link-type trunk
```

```
[PE2-Ten-GigabitEthernet1/0/2] port trunk permit vlan 100 3000
```

**3. Configure the devices between PE 1 and PE 2:**

**# Set the MTU to a minimum of 1504 bytes for each port on the path of QinQ frames. (Details not shown.)**

**# Configure all ports on the forwarding path to allow frames from VLANs 100 and 3000 to pass through without removing the VLAN tag. (Details not shown.)**