

QinQ Operation

XXXX Communication Technology Co., Ltd

Tel: (86)

Fax: (86)

URL:

Email:

All rights reserved. Printed in the People's Republic of China.

No part of this document may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual or otherwise without the prior written consent of XXXX Communication Technology Co., Ltd.

XXXX makes no representations or warranties with respect to this document contents and specifically disclaims any implied warranties of merchantability or fitness for any specific purpose. Further, XXXX reserves the right to revise this document and to make changes from time to time in its content without being obligated to notify any person of such revisions or changes.

XXXX values and appreciates comments you may have concerning our products or this document. Please address comments to:

XXXX Communication Technology Co., Ltd

Tel: (86)

Fax: (86)

URL:

Email:

All other products or services mentioned herein may be registered trademarks, trademarks, or service marks of their respective manufacturers, companies, or organizations.

Contents

- Chapter 1 QinQ Configuration.....4
 - 1.1 Introduction to QinQ4
 - 1.1.1 Understanding QinQ.....4
 - 1.1.2 Implementations of QinQ5
 - 1.1.3 Modification of TPID Value of QinQ Frames.....6
 - 1.2 Configuring QinQ.....7
 - 1.2.1 Default QinQ Configuration7
 - 1.2.2 Configure BASIC QinQ.....7
 - 1.2.3 Configure Selective QinQ.....8
 - 1.3 QinQ Configuration Example:9

Chapter 1 QinQ Configuration

When configuring QinQ, go to these sections for information you are interested in:

Introduction to QinQ

Configuring Basic QinQ

Configuring Selective QinQ

QinQ Configuration Example

1.1 Introduction to QinQ

1.1.1 Understanding QinQ

In the VLAN tag field defined in IEEE 802.1Q, only 12 bits are used for VLAN IDs, so a switch can support a maximum of 4,094 VLANs. In actual applications, however, a large number of VLANs are required to isolate users, especially in metropolitan area networks (MANs), and 4,094 VLANs are far from satisfying such requirements. shows the structure of 802.1Q-tagged and double-tagged Ethernet frames. The QinQ feature enables a device to support up to 4,094 x 4,094 VLANs to satisfy the requirement for the amount of VLANs in the MAN.

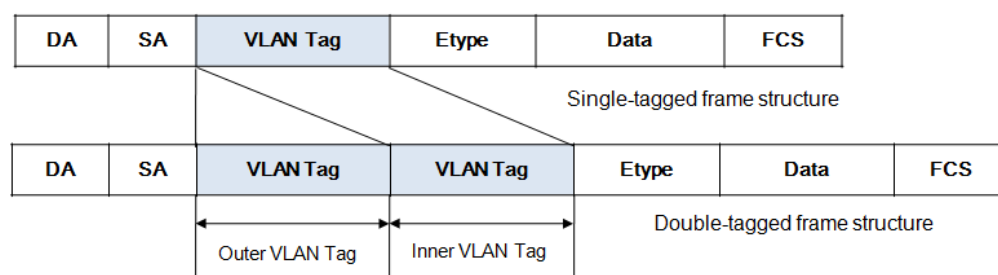


Figure 1-1 QinQ Ethernet frame structure

The port QinQ feature is a flexible, easy-to-implement Layer 2 VPN technique, which enables the access point to encapsulate an outer VLAN tag in Ethernet frames from customer networks (private networks), so that the Ethernet frames will travel across the service provider's backbone network (public network) with

double VLAN tags. The inner VLAN tag is the customer network VLAN tag while the outer one is the VLAN tag assigned by the service provider to the customer. In the public network, frames are forwarded based on the outer VLAN tag only, with the source MAC address learned as a MAC address table entry for the VLAN indicated by the outer tag, while the customer network VLAN tag is transmitted as part of the data in the frames.

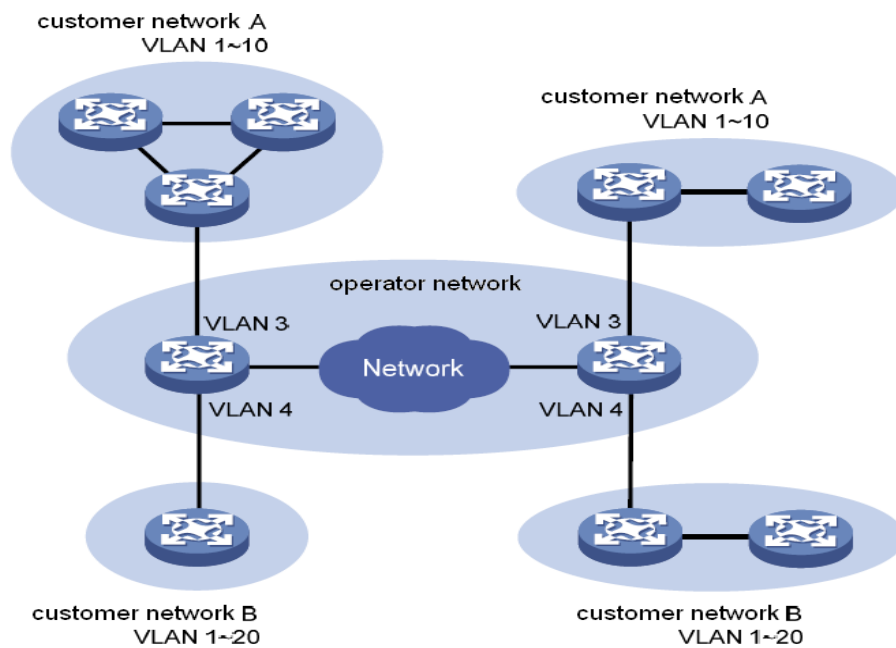


Figure 1-2 QinQ application

1.1.2 Implementations of QinQ

There are two types of QinQ implementations: basic QinQ and selective QinQ.

1) Basic QinQ

Basic QinQ is implemented through VLAN VPN.

With the VLAN VPN feature enabled on a port, when a frame arrives at the port, the switch will tag it with the port's default VLAN tag, regardless of whether the frame is tagged or untagged. If the received frame is already tagged, this frame becomes a double-tagged frame; if it is an untagged frame, it is tagged with the port's default VLAN tag.

2) Selective QinQ

Selective QinQ is a more flexible, VLAN-based implementation of QinQ. Selective QinQ is global. User can enable/disable it on port by using [no] qinq flexible-qinq command. If selective QinQ on port is disabled, the port is on static QinQ mode. If selective QinQ on port is enabled, global dynamic QinQ is for ports. Selective QinQ can:

- For ingress packet, different outer vlan tag can be added according to different inner VLAN ID
- For ingress packet, new VLAN tag can take the place of some specific VLAN Tag
- For ingress packet, some VLAN can be transparent transmit.

For QinQ-enabled port, there are different handlings for different port type:

- Uplink port: The Tag judgment on uplink port is based on the consistency between packet VID and configured global outer-tpid.
- Custom port: The Tag judgment on customer port is based on the consistency between packet VID and inner-tpid. The default inner-tpid is 0x8100

1.1.3 Modification of TPID Value of QinQ Frames

A VLAN tag uses the tag protocol identifier (TPID) field to identify the protocol type of the tag. The value of this field, as defined in IEEE 802.1Q, is 0x8100. The device can identify whether there is corresponded VLAN Tag according to TPID. If configured TPID is the same as the corresponded field, packet is regarded as with VLAN Tag.

The systems of different vendors may set the TPID of the outer VLAN tag of QinQ frames to different values. For compatibility with these systems, the S3750-48 series switches allow you to modify the TPID value so that the QinQ frames, when sent to the public network, carry the TPID value identical to the value of a particular vendor to allow interoperability with the devices of that vendor.

The TPID in an Ethernet frame has the same position with the protocol type field in a frame without a VLAN tag. To avoid problems in packet forwarding and handling in the network, you cannot set the TPID value to any of the values in the table below.

Table 1-1 Reserved protocol type 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 |
| 802.1x | 0x888E |
| GnLink | 0x0765 |
| GSTP | 0X5524 |

1.2 Configuring QinQ

1.2.1 Default QinQ Configuration

Table 1-2 Default QinQ configuration

| Item | Default value |
|----------------|---------------|
| QinQ function | Disable |
| QinQ port type | Uplink port |
| Outer TPID | 0X8100 |
| Inner TPID | 0X8100 |

1.2.2 Configure BASIC QinQ

Perform following commands in privilege mode.

Table 1-3 Enable basic QinQ

| Operation | Command | Description |
|--|--|---|
| Enter global configuration mode | configure terminal | |
| Enable basic QinQ | dtag | required |
| | | |
| Enter interface configuration mode | interface ethernet <i>interface-num</i> | Configurations made in interface configuration mode will take effect on the current port only; configurations made in interface range configuration mode will take effect on all ports in the port group. |
| Enter interface range configuration mode | interface range <i>interface-list</i> | |
| Specify port QinQ mode | dtag mode { customer uplink } | |
| Modify outer TPID | dtag outer-tpid <i>outer-tpid</i> | Optional,default value is 0x8100 |

1.2.3 Configure Selective QinQ

Perform following commands in privilege mode.

Table 1-4 Enable selective QinQ

| Operation | Command | Description |
|--|--|--|
| Enter global configuration mode | configure terminal | |
| Enable basic QinQ | dtag | |
| Enter interface configuration mode | interface ethernet <i>interface-num</i> | Configurations made in interface configuration mode will take effect on the current port only; configurations made in interface range configuration mode will take |
| Enter interface range configuration mode | Interface range <i>interface-list</i> | |

| | | |
|--|---|---|
| | | effect on all ports in the port group. |
| Specify port QinQ mode | dtag mode {customer uplink} | Uplink needs setting to be tag member in service vlan; customer needs setting to be untag member in service vlan. |
| Modify outer TPID | dtag { inner-tpid outer-tpid } <i>tpid-value</i> | Optional,default value is 0x8100 |
| Add different outer VLAN Tag for different inner VID | dtag insert <i>start-vlan-id end-vlan-id service-vlan-id</i> | |

1.3 QinQ Configuration Example:

!Basic QinQ configuration

Network requirements,As shown in Figure 1-3:

- Provider A and Provider B are access switches in the service provider network.
- Customer A and Customer B are access switches in the customer network.
- Provider A and Provider B, belonging to VLAN 100 created in the service provider network, are interconnected through trunk ports.
- In between Provider A and Provider B are network devices with TPID 0x9100.

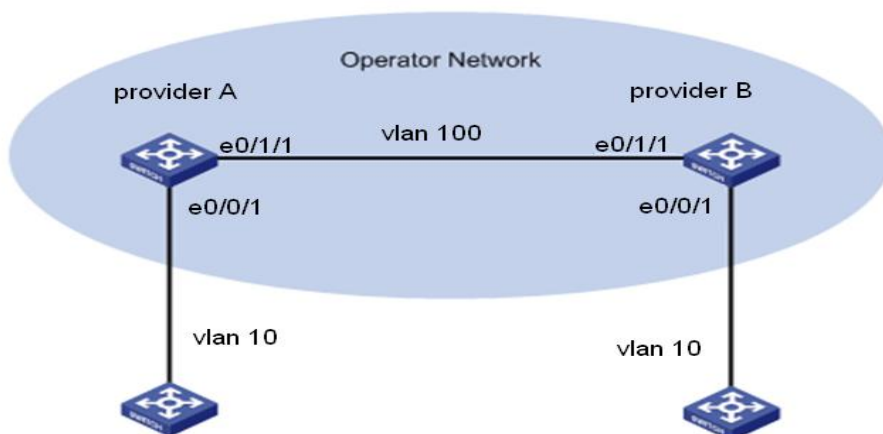


Figure 1-3 Basic QinQ application

It is required that Customer A and Customer B can communicate with each other across the service provider network.

(1) Configuration on Provider A

```
Switch#configure terminal
Switch(config)#dtag
Switch(config)#vlan 100
Switch(config-if-vlan)#switchport ethernet 0/0/1 ethernet 0/1/1
Switch(config-if-vlan)#interface ethernet 0/0/1
Switch(config-if-ethernet-0/0/1)#switchport default vlan 100
Switch(config-if-ethernet-0/0/1)#dtag mode customer
Switch(config-if-ethernet-0/0/1)#interface ethernet 0/1/1
Switch(config-if-ethernet-0/1/1)#switchport mode trunk
Switch(config-if-ethernet-0/1/1)#switchport trunk allowed vlan all
Switch(config-if-ethernet-0/1/1)#dtag outer-tpid 9100
Switch(config-if-ethernet-0/1/1)#exit
```

(2) Configuration on Provider B

```
Switch#configure terminal
Switch(config)#dtag
Switch(config)#vlan 100
Switch(config-if-vlan)#switchport ethernet 0/0/1 ethernet 0/1/1
Switch(config-if-vlan)#interface ethernet 0/0/1
Switch(config-if-ethernet-0/0/1)#switchport default vlan 100
Switch(config-if-ethernet-0/0/1)#qinq mode customer
Switch(config-if-ethernet-0/0/1)#interface ethernet 0/1/1
Switch(config-if-ethernet-0/1/1)#switchport mode trunk
Switch(config-if-ethernet-0/1/1)#switchport trunk allowed vlan all
Switch(config-if-ethernet-0/1/1)#dtag outer-tpid 9100
Switch(config-if-ethernet-0/1/1)#exit
```

!Selective QinQ Configuration

Network requirements,As shown in Figure 1-4:

- Provider A and Provider B are access switches in the service provider network that connect the customer network.

- The customer network is divided into VLAN 10 and VLAN 20.
- In between Provider A and Provider B are network devices with TPID 0x9100

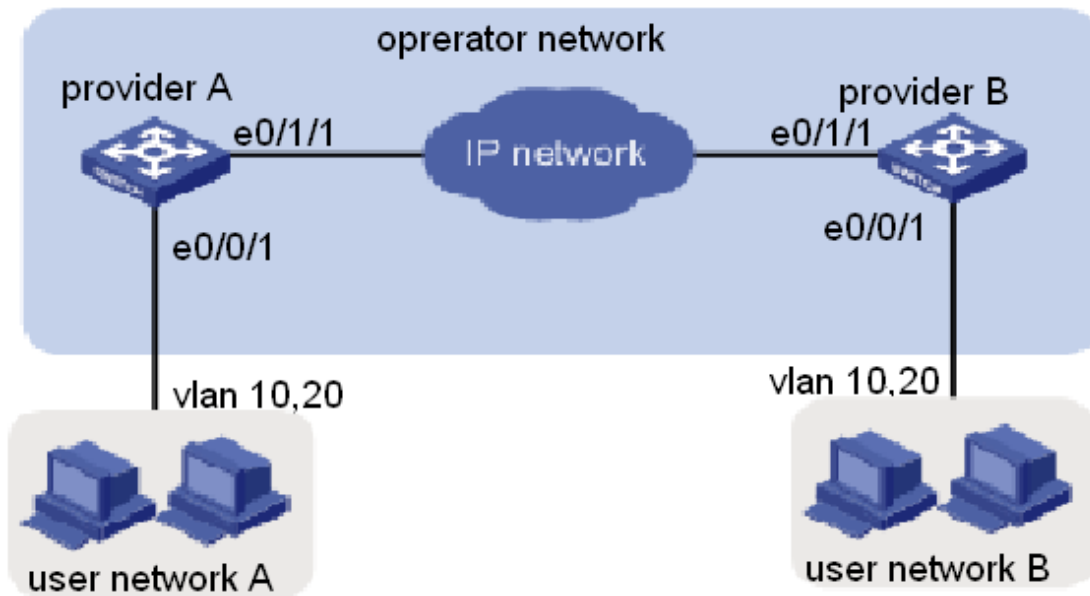


Figure 1-4 selective QinQ application

Configure selective QinQ so that frames from the customer network can pass through the service provider network tagged with SVLAN 100.

(1) Configuration on Provider A

Switch#configure terminal

Switch(config)#dtag

Switch(config)#vlan 100

Switch(config-if-vlan)#switchport ethernet 0/0/1 ethernet 0/1/1

Switch(config-if-vlan)#interface ethernet 0/0/1

Switch(config-if-ethernet-0/0/1)#dtag mode customer

Switch(config-if-ethernet-0/0/1)#dtag insert 10 10 100

Switch(config-if-ethernet-0/0/1)#dtag insert 20 20 100

Switch(config-if-ethernet-0/0/1)#interface ethernet 0/1/1

Switch(config-if-ethernet-0/1/1)#switchport mode trunk

Switch(config-if-ethernet-0/1/1)#switchport trunk allowed vlan all

Switch(config-if-ethernet-0/1/1)#dtag outer-tpid 9100

Switch(config-if-ethernet-0/1/1)#exit

(2) Configuration on Provider B

Switch#configure terminal

Switch#configure terminal

Switch(config)#dtag

Switch(config)#vlan 100

Switch(config-if-vlan)#switchport ethernet 0/0/1 ethernet 0/1/1

Switch(config-if-vlan)#interface ethernet 0/0/1

Switch(config-if-ethernet-0/0/1)#dtag mode customer

Switch(config-if-ethernet-0/0/1)#dtag insert 10 10 100

Switch(config-if-ethernet-0/0/1)#dtag insert 20 20 100

Switch(config-if-ethernet-0/0/1)#interface ethernet 0/1/1

Switch(config-if-ethernet-0/1/1)#switchport mode trunk

Switch(config-if-ethernet-0/1/1)#switchport trunk allowed vlan all

Switch(config-if-ethernet-0/1/1)#dtag outer-tpid 9100

Switch(config-if-ethernet-0/1/1)#exit