HP FlexFabric 5700 Switch Series

ACL and QoS

Configuration Guide

Part number: 5998-6661

Software version: Release 2416

Document version: 6W100-20150130

Legal and notice information

The information contained herein is subject to change without notice.

HEWLETT-PACKARD COMPANY MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Contents

Configuring ACLs

Overview

An access control list (ACL) is a set of rules (or permit or deny statements) for identifying traffic based on criteria such as source IP address, destination IP address, and port number.

ACLs are primarily used for packet filtering. "Configuring packet filtering with ACLs" provides an example. You can use ACLs in QoS, security, routing, and other feature modules for identifying traffic. The packet drop or forwarding decisions depend on the modules that use ACLs.

Applications on the switch

An ACL is implemented in hardware or software, depending on the module that uses it.

- If the module is implemented in hardware (for example, the packet filter or QoS module), the ACL is applied to hardware to process traffic.
- If the module is implemented in software (for example, the routing module or the user interface access control module such as Telnet, or SNMP), the ACL is applied to software to process traffic.

The user interface access control module denies packets that do not match any ACL. Some modules (QoS for example) ignore the permit or deny action in ACL rules and do not base their drop or forwarding decisions on the action set in ACL rules. See the specified module for information about ACL application.

ACL categories

Category	ACL number	IP version	Match criteria
Basic ACLs	2000 to 2999	IPv4	Source IPv4 address.
		IPv6	Source IPv6 address.
Advanced ACLs	3000 to 3999	IPv4	Source IPv4 address, destination IPv4 address, packet priority, protocol number, and other Layer 3 and Layer 4 header fields.
		IPv6	Source IPv6 address, destination IPv6 address, packet priority, protocol number, and other Layer 3 and Layer 4 header fields.
Ethernet frame header ACLs	4000 to 4999	IPv4 and IPv6	Layer 2 header fields, such as source and destination MAC addresses, 802.1p priority, and link layer protocol type.
User-defined ACLs	5000 to 5999	IPv4 and IPv6	User specified matching patterns in protocol headers.

Numbering and naming ACLs

Each ACL category has a unique range of ACL numbers. When creating an ACL, you must assign it a number. In addition, you can assign the ACL a name for ease of identification. After creating an ACL with a name, you cannot rename it or delete its name.

For an IPv4 basic or advanced ACLs, its ACL number and name must be unique in IPv4. For an IPv6 basic or advanced ACL, its ACL number and name must be unique in IPv6.

Match order

The rules in an ACL are sorted in a specific order. When a packet matches a rule, the device stops the match process and performs the action defined in the rule. If an ACL contains overlapping or conflicting rules, the matching result and action to take depend on the rule order.

The following ACL match orders are available:

- **config**—Sorts ACL rules in ascending order of rule ID. A rule with a lower ID is matched before a rule with a higher ID. If you use this method, check the rules and their order carefully.

NOTE:

The match order of user-defined ACLs can only be **config**.

- **auto**—Sorts ACL rules in depth-first order. Depth-first ordering makes sure any subset of a rule is always matched before the rule. Table 1 lists the sequence of tie breakers that depth-first ordering uses to sort rules for each type of ACL.

Table 1 Sort ACL rules in depth-first order

ACL category	Sequence of tie breakers
IPv4 basic ACL	1. More 0s in the source IPv4 address wildcard (more 0s means a narrower IPv4 address range). 2. Rule configured earlier.
IPv4 advanced ACL	1. Specific protocol number. 2. More 0s in the source IPv4 address wildcard mask. 3. More 0s in the destination IPv4 address wildcard. 4. Narrower TCP/UDP service port number range. 5. Rule configured earlier.
IPv6 basic ACL	1. Longer prefix for the source IPv6 address (a longer prefix means a narrower IPv6 address range). 2. Rule configured earlier.
IPv6 advanced ACL	1. Specific protocol number. 2. Longer prefix for the source IPv6 address. 3. Longer prefix for the destination IPv6 address. 4. Narrower TCP/UDP service port number range. 5. Rule configured earlier.
Ethernet frame header ACL	1. More 1s in the source MAC address mask (more 1s means a smaller MAC address). 2. More 1s in the destination MAC address mask. 3. Rule configured earlier.

A wildcard mask, also called an inverse mask, is a 32-bit binary number represented in dotted decimal notation. In contrast to a network mask, the 0 bits in a wildcard mask represent "do care" bits, and the 1 bits represent "don't care" bits. If the "do care" bits in an IP address are identical to the "do care" bits in an IP address criterion, the IP address matches the criterion. All "don't care" bits are ignored. The 0s and 1s in a wildcard mask can be noncontiguous. For example, 0.255.0.255 is a valid wildcard mask.

Rule numbering

ACL rules can be manually numbered or automatically numbered. This section describes how automatic ACL rule numbering works.

Rule numbering step

If you do not assign an ID to the rule you are creating, the system automatically assigns it a rule ID. The rule numbering step sets the increment by which the system automatically numbers rules. For example, the default ACL rule numbering step is 5. If you do not assign IDs to rules you are creating, they are automatically numbered 0, 5, 10, 15, and so on. The wider the numbering step, the more rules you can insert between two rules.

By introducing a gap between rules rather than contiguously numbering rules, you have the flexibility of inserting rules in an ACL. This feature is important for a config-order ACL, where ACL rules are matched in ascending order of rule ID.

Automatic rule numbering and renumbering

The ID automatically assigned to an ACL rule takes the nearest higher multiple of the numbering step to the current highest rule ID, starting with 0.

For example, if the numbering step is 5 (the default), and there are five ACL rules numbered 0, 5, 9, 10, and 12, the newly defined rule is numbered 15. If the ACL does not contain any rule, the first rule is numbered 0.

Whenever the step changes, the rules are renumbered, starting from 0. For example, if there are five rules numbered 5, 10, 13, 15, and 20, changing the step from 5 to 2 causes the rules to be renumbered 0, 2, 4, 6, and 8.

Fragments filtering with ACLs

Traditional packet filtering matches only first fragments of packets, and allows all subsequent non-first fragments to pass through. Attackers can fabricate non-first fragments to attack networks.

To avoid the risks, the HP ACL implementation does the following:

- Filters all fragments by default, including non-first fragments.
- Allows for matching criteria modification for efficiency. For example, you can configure the ACL to filter only non-first fragments.

Configuration task list

Tasks at a glance
(Required.) Perform at least one of the following tasks: • Configuring a basic ACL   ○ Configuring an IPv4 basic ACL   ○ Configuring an IPv6 basic ACL • Configuring an advanced ACL   ○ Configuring an IPv4 advanced ACL   ○ Configuring an IPv6 advanced ACL • Configuring an Ethernet frame header ACL • Configuring a user-defined ACL
(Optional.) Copying an ACL
(Optional.) Configuring packet filtering with ACLs

Configuring a basic ACL

This section describes procedures for configuring IPv4 and IPv6 basic ACLs.

Configuring an IPv4 basic ACL

IPv4 basic ACLs match packets based only on source IP addresses.

To configure an IPv4 basic ACL:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Create an IPv4 basic ACL and enter its view.	**acl number** *acl-number* [**name** *acl-name*] [**match-order** { **auto** | **config** }]	By default, no ACL exists. IPv4 basic ACLs are numbered in the range of 2000 to 2999. You can use the **acl name** *acl-name* command to enter the view of a named ACL.
3. (Optional.) Configure a description for the IPv4 basic ACL.	**description** *text*	By default, an IPv4 basic ACL has no ACL description.
4. (Optional.) Set the rule numbering step.	**step** *step-value*	The default setting is 5.
5. Create or edit a rule.	**rule** [*rule-id*] { **deny** | **permit** } [**counting** | **fragment** | **logging** | **source** { *source-address source-wildcard* | **any** } | **time-range** *time-range-name*] *	By default, an IPv4 basic ACL does not contain any rule. The **logging** keyword takes effect only when the module (for example, packet filtering) that uses the ACL supports logging.

Step	Command	Remarks
6. (Optional.) Add or edit a rule comment.	**rule** *rule-id* **comment** *text*	By default, no rule comments are configured.

Configuring an IPv6 basic ACL

IPv6 basic ACLs match packets based only on source IP addresses.

To configure an IPv6 basic ACL:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Create an IPv6 basic ACL view and enter its view.	**acl ipv6 number** *acl-number* [**name** *acl-name*] [**match-order** { **auto** | **config** }]	By default, no ACL exists. IPv6 basic ACLs are numbered in the range of 2000 to 2999. You can use the **acl ipv6 name** *acl-name* command to enter the view of a named ACL.
3. (Optional.) Configure a description for the IPv6 basic ACL.	**description** *text*	By default, an IPv6 basic ACL has no ACL description.
4. (Optional.) Set the rule numbering step.	**step** *step-value*	The default setting is 5.
5. Create or edit a rule.	**rule** [*rule-id*] { **deny** | **permit** } [**counting** | **fragment** | **logging** | **routing** [**type** *routing-type*] | **source** { *source-address source-prefix* | *source-address/source-prefix* | **any** } | **time-range** *time-range-name*] *	By default, an IPv6 basic ACL does not contain any rule. The **logging** keyword takes effect only when the module (for example, packet filtering) that uses the ACL supports logging. If an ACL is for QoS traffic classification or packet filtering: • Do not specify the **fragment** keywords. • Do not specify the **routing** keyword if the ACL is for outbound QoS traffic classification or outbound packet filtering.
6. (Optional.) Add or edit a rule comment.	**rule** *rule-id* **comment** *text*	By default, no rule comments are configured.

Configuring an advanced ACL

This section describes procedures for configuring IPv4 and IPv6 advanced ACLs.

Configuring an IPv4 advanced ACL

IPv4 advanced ACLs match packets based on the following criteria:

- Source IP addresses.
- Destination IP addresses.
- Packet priorities.
- Protocol numbers.
- Other protocol header information, such as TCP/UDP source and destination port numbers, TCP flags, ICMP message types, and ICMP message codes.

Compared to IPv4 basic ACLs, IPv4 advanced ACLs allow more flexible and accurate filtering.

To configure an IPv4 advanced ACL:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Create an IPv4 advanced ACL and enter its view.	**acl number** *acl-number* [**name** *acl-name*] [**match-order** { **auto** | **config** }]	By default, no ACL exists. IPv4 advanced ACLs are numbered in the range of 3000 to 3999. You can use the **acl name** *acl-name* command to enter the view of a named ACL.
3. (Optional.) Configure a description for the IPv4 advanced ACL.	**description** *text*	By default, an IPv4 advanced ACL has no ACL description.
4. (Optional.) Set the rule numbering step.	**step** *step-value*	The default setting is 5.
5. Create or edit a rule.	**rule** [*rule-id*] { **deny** | **permit** } *protocol* [{ { **ack** *ack-value* | **fin** *fin-value* | **psh** *psh-value* | **rst** *rst-value* | **syn** *syn-value* | **urg** *urg-value* } * | **established** } | **counting** | **destination** { *dest-address dest-wildcard* | **any** } | **destination-port** *operator port1* [*port2*] | { **dscp** *dscp* | { **precedence** *precedence* | **tos** *tos* } * } | **fragment** | **icmp-type** { *icmp-type* [*icmp-code*] | *icmp-message* } | **logging** | **source** { *source-address source-wildcard* | **any** } | **source-port** *operator port1* [*port2*] | **time-range** *time-range-name*] *	By default, an IPv4 advanced ACL does not contain any rule. The **logging** keyword takes effect only when the module (for example, packet filtering) that uses the ACL supports logging. If an ACL is for QoS traffic classification or packet filtering, do not specify **neq** for the *operator* argument.
6. (Optional.) Add or edit a rule comment.	**rule** *rule-id* **comment** *text*	By default, no rule comments are configured.

Configuring an IPv6 advanced ACL

IPv6 advanced ACLs match packets based on the following criteria:

- Source IPv6 addresses.
- Destination IPv6 addresses.
- Packet priorities.
- Protocol numbers.
- Other protocol header fields such as the TCP/UDP source port number, TCP/UDP destination port number, ICMPv6 message type, and ICMPv6 message code.

Compared to IPv6 basic ACLs, IPv6 advanced ACLs allow more flexible and accurate filtering.

To configure an IPv6 advanced ACL:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Create an IPv6 advanced ACL and enter its view.	**acl ipv6 number** *acl-number* [**name** *acl-name*] [**match-order** { **auto** | **config** }]	By default, no ACL exists. IPv6 advanced ACLs are numbered in the range of 3000 to 3999. You can use the **acl ipv6 name** *acl-name* command to enter the view of a named ACL.
3. (Optional.) Configure a description for the IPv6 advanced ACL.	**description** *text*	By default, an IPv6 advanced ACL has no ACL description.
4. (Optional.) Set the rule numbering step.	**step** *step-value*	The default setting is 5.

Step	Command	Remarks
5. Create or edit a rule.	**rule** [*rule-id*] { **deny** | **permit** } *protocol* [{ { **ack** *ack-value* | **fin** *fin-value* | **psh** *psh-value* | **rst** *rst-value* | **syn** *syn-value* | **urg** *urg-value* } * | **established** } | **counting** | **destination** { *dest-address dest-prefix* | *dest-address/dest-prefix* | **any** } | **destination-port** *operator port1* [*port2*] | **dscp** *dscp* | **flow-label** *flow-label-value* | **fragment** | **icmp6-type** { *icmp6-type icmp6-code* | *icmp6-message* } | **logging** | **routing** [**type** *routing-type*] | **hop-by-hop** [**type** *hop-type*] | **source** { *source-address source-prefix* | *source-address/source-prefix* | **any** } | **source-port** *operator port1* [*port2*] | **time-range** *time-range-name*] *	By default, IPv6 advanced ACL does not contain any rule. The **logging** keyword takes effect only when the module (for example, packet filtering) that uses the ACL supports logging. If an ACL is for QoS traffic classification or packet filtering: • Do not specify the **fragment** keyword. • Do not specify **neq** for the *operator* argument. • Do not specify the **routing**, **hop-by-hop**, or **flow-label** keyword if the ACL is for outbound QoS traffic classification or outbound packet filtering. • Do not specify **ipv6-ah** for the *protocol* argument, nor set its value to 0, 43, 44, 51, or 60, if the ACL is for outbound QoS traffic classification or outbound packet filtering.
6. (Optional.) Add or edit a rule comment.	**rule** *rule-id* **comment** *text*	By default, no rule comments are configured.

NOTE:

If an ACL is to match information in the IPv6 packet payload, it can only match packets with one extension header. It cannot match packets with two or more extension headers or with the Encapsulating Security Payload Header.

Configuring an Ethernet frame header ACL

Ethernet frame header ACLs, also called "Layer 2 ACLs," match packets based on Layer 2 protocol header fields, such as:

- Source MAC address.
- Destination MAC address.
- 802.1p priority (VLAN priority).
- Link layer protocol type.

To configure an Ethernet frame header ACL:

Step	Command	Remarks
1. Enter system view.	system-view	N/A

Step	Command	Remarks
2. Create an Ethernet frame header ACL and enter its view.	**acl number** *acl-number* [**name** *acl-name*] [**match-order** { **auto** | **config** }]	By default, no ACL exists. Ethernet frame header ACLs are numbered in the range of 4000 to 4999. You can use the **acl name** *acl-name* command to enter the view of a named ACL.
3. (Optional.) Configure a description for the Ethernet frame header ACL.	**description** *text*	By default, an Ethernet frame header ACL has no ACL description.
4. (Optional.) Set the rule numbering step.	**step** *step-value*	The default setting is 5.
5. Create or edit a rule.	**rule** [*rule-id*] { **deny** | **permit** } [**cos** *vlan-pri* | **counting** | **dest-mac** *dest-address dest-mask* | { **lsap** *lsap-type lsap-type-mask* | **type** *protocol-type protocol-type-mask* } | **source-mac** *source-address source-mask* | **time-range** *time-range-name*] *	By default, an Ethernet frame header ACL does not contain any rule. If an Ethernet frame header ACL is used for packet filtering or QoS traffic classification and the **lsap** keyword is used, the *lsap-type* argument value must be AAAA, and the *lsap-type-mask* argument value must be FFFF. Otherwise, the ACL does not take effect.
6. (Optional.) Add or edit a rule comment.	**rule** *rule-id* **comment** *text*	By default, no rule comments are configured.

Configuring a user-defined ACL

User-defined ACLs allow you to customize rules based on information in protocol headers. You can define a user-defined ACL to match packets. A specific number of bytes after an offset (relative to the specified header) are compared against a match pattern after being ANDed with a match pattern mask.

To configure a user-defined ACL:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Create a user-defined ACL and enter its view.	**acl number** *acl-number* [**name** *acl-name*]	By default, no ACL exists. User-defined ACLs are numbered in the range of 5000 to 5999. You can use the **acl name** *acl-name* command to enter the view of a named ACL.
3. (Optional.) Configure a description for the user-defined ACL.	**description** *text*	By default, a user-defined ACL has no ACL description.

Step	Command	Remarks
4. Create or edit a rule.	**rule** [*rule-id*] { **deny** | **permit** } [{ **l2** *rule-string rule-mask offset* }&<1-8>] [**counting** | **time-range** *time-range-name*] *	By default, a user-defined ACL does not contain any rule. A user-defined ACL cannot be used for outbound QoS traffic classification or outbound packet filtering.
5. (Optional.) Add or edit a rule comment.	**rule** *rule-id* **comment** *text*	By default, no rule comments are configured.

NOTE:

If a user-defined ACL is to match packets with VLAN tags, the offset must include the length of the VLAN tags. Each VLAN tag is 4 bytes long.

Copying an ACL

You can create an ACL by copying an existing ACL (source ACL). The new ACL (destination ACL) has the same properties and content as the source ACL, but not the same ACL number and name.

To successfully copy an ACL, make sure:

- The destination ACL number is from the same category as the source ACL number.
- The source ACL already exists, but the destination ACL does not.

To copy an ACL:

Step	Command
1. Enter system view.	**system-view**
2. Copy an existing ACL to create a new ACL.	**acl** [**ipv6**] **copy** { *source-acl-number* | **name** *source-acl-name* } **to** { *dest-acl-number* | **name** *dest-acl-name* }

Configuring packet filtering with ACLs

This section describes procedures for applying an ACL to filter incoming or outgoing IPv4 or IPv6 packets on the specified interface.

NOTE:

The ACL-based packet filter function is available on Ethernet interfaces, VLAN interfaces, S-channel interfaces, S-channel aggregate interfaces, VSI interfaces, and VSI aggregate interfaces. For more information about S-channel interfaces, S-channel aggregate interfaces, VSI interfaces, and VSI aggregate interfaces, see *EVB Configuration Guide*.

Applying an ACL to an interface for packet filtering

Step		Command	Remarks
1.	Enter system view.	**system-view**	N/A
2.	Enter interface view.	**interface** *interface-type interface-number*	N/A
3.	Apply an ACL to the interface to filter packets.	**packet-filter** [**ipv6**] { *acl-number* | **name** *acl-name* } { **inbound** | **outbound** } [**hardware-count**]	By default, an interface does not filter packets. You can apply only one ACL to the same direction of an interface.

Configuring the applicable scope of packet filtering on a VLAN interface

You can configure the packet filtering on a VLAN interface to filter the following packets:

- Packets forwarded at Layer 3 by the VLAN interface.
- All packets, including packets forwarded at Layer 3 by the VLAN interface and packets forwarded at Layer 2 by the physical ports associated with the VLAN interface.

To configure the applicable scope of packet filtering on a VLAN interface:

Step		Command	Remarks
1.	Enter system view.	**system-view**	N/A
2.	Create a VLAN interface and enter its view.	**interface vlan-interface** *vlan-interface-id*	If the VLAN interface already exists, you directly enter its view. By default, no VLAN interface exists.
3.	Specify the applicable scope of packet filtering on the VLAN interface.	**packet-filter filter** [**route** | **all**]	By default, the packet filtering filters packets forwarded at Layer 3.

Setting the interval for generating and outputting packet filtering logs

After you set the interval, the device periodically generates and outputs the packet filtering logs to the information center, including the number of matching packets and the matched ACL rules. For more information about information center, see *Network Management and Monitoring Configuration Guide*.

To set the interval for generating and outputting packet filtering logs:

Step		Command	Remarks
1.	Enter system view.	**system-view**	N/A
2.	Set the interval for generating and outputting packet filtering logs.	**acl** [**ipv6**] **logging interval** *interval*	The default setting is 0 minutes, which mean that no packet filtering logs are generated.

Setting the packet filtering default action

Step		Command	Remarks
1.	Enter system view.	**system-view**	N/A
2.	Set the packet filtering default action to deny.	**packet-filter default deny**	By default, the packet filter permits packets that do not match any ACL rule to pass.

Displaying and maintaining ACLs

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display ACL configuration and match statistics.	**display acl** [**ipv6**] { *acl-number* | **all** | **name** *acl-name* }
Display whether an ACL has been successfully applied to an interface for packet filtering.	**display packet-filter** { **interface** [*interface-type interface-number*] [**inbound** | **outbound**] | **interface vlan-interface** *vlan-interface-number* [**inbound** | **outbound**] [**slot** *slot-number*] }
Display match statistics for packet filtering ACLs.	**display packet-filter statistics interface** *interface-type interface-number* { **inbound** | **outbound** } [[**ipv6**] { *acl-number* | **name** *acl-name* }] [**brief**]
Display the accumulated statistics for packet filtering ACLs.	**display packet-filter statistics sum** { **inbound** | **outbound** } [**ipv6**] { *acl-number* | **name** *acl-name* } [**brief**]
Display detailed ACL packet filtering information.	**display packet-filter verbose interface** *interface-type interface-number* { **inbound** | **outbound** } [[**ipv6**] { *acl-number* | **name** *acl-name* }] [**slot** *slot-number*]
Display QoS and ACL resource usage.	**display qos-acl resource** [**slot** *slot-number*]
Clear ACL statistics.	**reset acl** [**ipv6**] **counter** { *acl-number* | **all** | **name** *acl-name* }
Clear match statistics (including the accumulated statistics) for packet filtering ACLs.	**reset packet-filter statistics interface** [*interface-type interface-number*] { **inbound** | **outbound** } [[**ipv6**] { *acl-number* | **name** *acl-name* }]

ACL configuration example

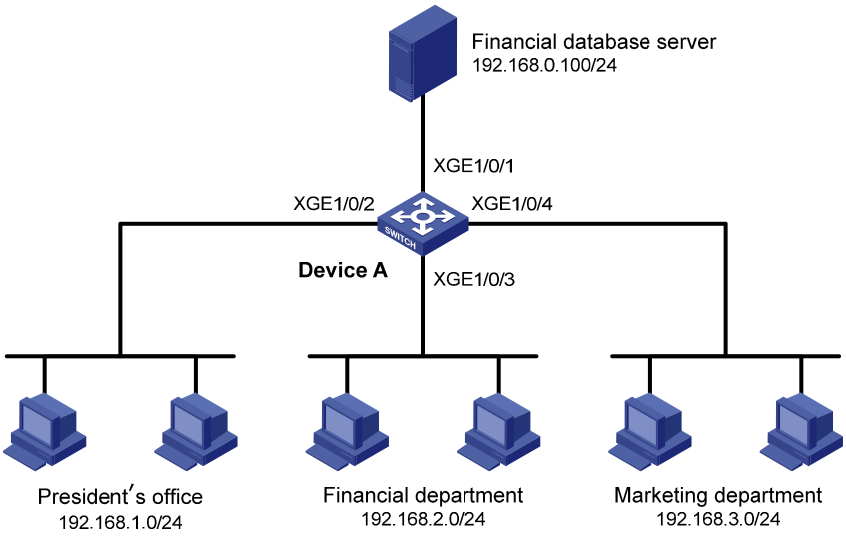
Network requirements

A company interconnects its departments through Device A. Configure an ACL to:

- Permit access from the President's office at any time to the financial database server.

- Permit access from the Financial department to the database server only during working hours (from 8:00 to 18:00) on working days.
- Deny access from any other department to the database server.

Figure 1 Network diagram



Configuration procedure

# Create a periodic time range from 8:00 to 18:00 on working days.

```
<DeviceA> system-view
[DeviceA] time-range work 08:0 to 18:00 working-day
```

# Create an IPv4 advanced ACL numbered 3000 and configure three rules in the ACL. One rule permits access from the President's office to the financial database server, one rule permits access from the Financial department to the database server during working hours, and one rule denies access from any other department to the database server.

```
[DeviceA] acl number 3000
[DeviceA-acl-adv-3000] rule permit ip source 192.168.1.0 0.0.0.255 destination
192.168.0.100 0
[DeviceA-acl-adv-3000] rule permit ip source 192.168.2.0 0.0.0.255 destination
192.168.0.100 0 time-range work
[DeviceA-acl-adv-3000] rule deny ip source any destination 192.168.0.100 0
[DeviceA-acl-adv-3000] quit
```

# Apply IPv4 advanced ACL 3000 to filter outgoing packets on interface Ten-GigabitEthernet 1/0/1.

```
[DeviceA] interface ten-gigabitethernet 1/0/1
[DeviceA-Ten-GigabitEthernet1/0/1] packet-filter 3000 outbound
[DeviceA-Ten-GigabitEthernet1/0/1] quit
```

Verifying the configuration

# Ping the database server from a PC in the Financial department during the working hours. (All PCs in this example use Windows XP).

```
C:\> ping 192.168.0.100
```

```
Pinging 192.168.0.100 with 32 bytes of data:

Reply from 192.168.0.100: bytes=32 time=1ms TTL=255
Reply from 192.168.0.100: bytes=32 time<1ms TTL=255
Reply from 192.168.0.100: bytes=32 time<1ms TTL=255
Reply from 192.168.0.100: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.0.100:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

The output shows that the database server can be pinged.

# Ping the database server from a PC in the Marketing department during the working hours.

```
C:\> ping 192.168.0.100

Pinging 192.168.0.100 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.0.100:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

The output shows the database server cannot be pinged.

# Display configuration and match statistics for IPv4 advanced ACL 3000 on Device A during the working hours.

```
[DeviceA] display acl 3000
Advanced ACL  3000, named -none-, 3 rules,
ACL's step is 5
 rule 0 permit ip source 192.168.1.0 0.0.0.255 destination 192.168.0.100 0
 rule 5 permit ip source 192.168.2.0 0.0.0.255 destination 192.168.0.100 0 time-range work (Active)
 rule 10 deny ip destination 192.168.0.100 0
```

The output shows that rule 5 is active.

QoS overview

In data communications, Quality of Service (QoS) provides differentiated service guarantees for diversified traffic in terms of bandwidth, delay, jitter, and drop rate, all of which can affect QoS.

Network resources are limited. When configuring a QoS scheme, you must consider the characteristics of different applications. For example, when bandwidth is fixed, more bandwidth used by one user leaves less bandwidth for others. QoS prioritizes traffic to balance the interests of users and manages network resources.

The following section describes typical QoS service models and widely used QoS techniques.

QoS service models

This section describes several typical QoS service models.

Best-effort service model

The best-effort model is a single-service model. As the simplest service model, the best-effort model is not as reliable as other models and does not guarantee delay-free delivery.

The best-effort service model is the default model for the Internet and applies to most network applications. It uses the First In First Out (FIFO) queuing mechanism.

IntServ model

The integrated service (IntServ) model is a multiple-service model that can accommodate diverse QoS requirements. This service model provides the most granularly differentiated QoS by identifying and guaranteeing definite QoS for each data flow.

In the IntServ model, an application must request service from the network before it sends data. IntServ signals the service request with the RSVP. All nodes receiving the request reserve resources as requested and maintain state information for the application flow.

The IntServ model demands high storage and processing capabilities because it requires all nodes along the transmission path to maintain resource state information for each flow. This model is suitable for small-sized or edge networks, but not large-sized networks, for example, the core layer of the Internet, where billions of flows are present.

DiffServ model

The differentiated service (DiffServ) model is a multiple-service model that can meet diverse QoS requirements. It is easy to implement and extend. DiffServ does not signal the network to reserve resources before sending data, as IntServ does.

QoS techniques overview

The QoS techniques include the following features:

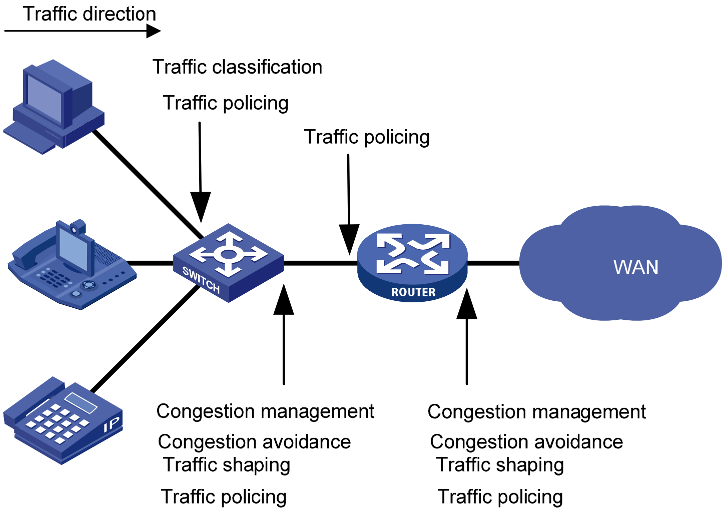
- Traffic classification.
- Traffic policing.
- Traffic shaping.
- Rate limit.
- Congestion management.
- Congestion avoidance.

The following section briefly introduces these QoS techniques.

All QoS techniques in this document are based on the DiffServ model.

Deploying QoS in a network

Figure 2 Position of the QoS techniques in a network



As shown in Figure 2, traffic classification, traffic shaping, traffic policing, congestion management, and congestion avoidance mainly implement the following features:

- **Traffic classification**—Uses match criteria to assign packets with the same characteristics to a traffic class. Based on traffic classes, you can provide differentiated services.
- **Traffic policing**—Polices flows and imposes penalties to prevent aggressive use of network resources. You can apply traffic policing to both incoming and outgoing traffic of a port.
- **Traffic shaping**—Adapts the output rate of traffic to the network resources available on the downstream device to eliminate packet drops. Traffic shaping usually applies to the outgoing traffic of a port.
- **Congestion management**—Provides a resource scheduling policy to determine the packet forwarding sequence when congestion occurs. Congestion management usually applies to the outgoing traffic of a port.

- **Congestion avoidance**—Monitors the network resource usage. It is usually applied to the outgoing traffic of a port. When congestion worsens, congestion avoidance reduces the queue length by dropping packets.

Configuring a QoS policy

You can configure QoS by using the MQC approach or non-MQC approach. Some features support both approaches, but some support only one.

Non-MQC approach

In the non-MQC approach, you configure QoS service parameters without using a QoS policy. For example, you can use the rate limit feature to set a rate limit on an interface without using a QoS policy.

MQC approach

In the modular QoS configuration (MQC) approach, you configure QoS service parameters by using QoS policies. A QoS policy defines the shaping, policing, or other QoS actions to take on different classes of traffic. It is a set of class-behavior associations.

A traffic class is a set of match criteria for identifying traffic, and it uses the AND or OR operator.

- If the operator is AND, a packet must match all the criteria to match the traffic class.
- If the operator is OR, a packet matches the traffic class if it matches any of the criteria in the traffic class.

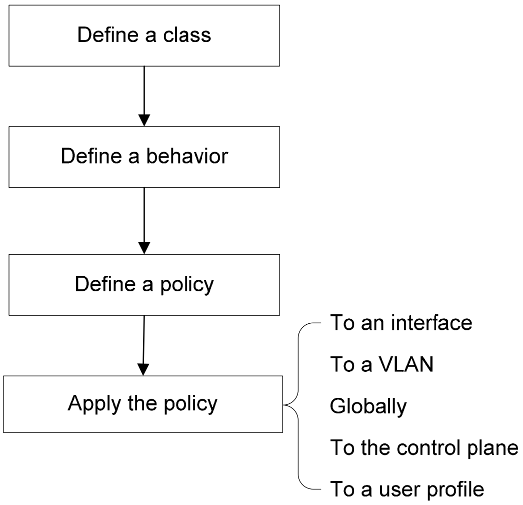
A traffic behavior defines a set of QoS actions to take on packets, such as priority marking and redirect.

By associating a traffic behavior with a traffic class in a QoS policy, you apply QoS actions in the traffic behavior to the traffic class.

Configuration procedure diagram

Figure 3 shows how to configure a QoS policy.

Figure 3 QoS policy configuration procedure



Defining a traffic class

Configuration guidelines

When you configure a traffic class, follow these restrictions and guidelines:

- If the traffic class includes the **customer-vlan-id** match criterion, a QoS policy that contains the traffic class can be applied only to interfaces.
- If the traffic class includes both the **control-plane protocol** or **control-plane protocol-group** criterion and other criteria, the QoS policy that contains the traffic class cannot be applied correctly.
- If the traffic class includes the **control-plane protocol** or **control-plane protocol-group** match criterion, the QoS policy that contains the traffic class can be applied only to a control plane.
- To configure multiple values for a match criterion, perform the following tasks:
  - Set the logical operator to OR.
  - Configure multiple **if-match** commands for the match criterion.

  For the **customer-vlan-id** and **service-vlan-id** match criteria, you can configure multiple values in one **if-match** command when the logical operator is OR or AND.
- If the configured logical operator is AND for the traffic class, the actual logical operator for the rules in an ACL match criterion is OR.

Configuration procedure

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Create a traffic class and enter traffic class view.	**traffic classifier** *classifier-name* [**operator** { **and** | **or** }]	By default, no traffic class is configured.
3. Configure match criteria.	**if-match** *match-criteria*	By default, no match criterion is configured. For more information, see the **if-match** command in *ACL and QoS Command Reference*.

Table 2 Available match criteria

Option	Description
acl [**ipv6**] { *acl-number* | **name** *acl-name* }	Matches an ACL. The *acl-number* argument has the following value ranges: • 2000 to 3999 for IPv4 ACLs. • 2000 to 3999 for IPv6 ACLs. • 4000 to 4999 for Ethernet frame header ACLs. • 5000 to 5999 for user-defined ACLs. The *acl-name* argument is a case-insensitive string of 1 to 63 characters, which must start with an English letter. To avoid confusion, make sure the argument is not **all**.
any	Matches all packets.

Option	Description
control-plane protocol *protocol-name*&<1-8>	Matches control plane protocols. The *protocol-name*&<1-8> argument specifies a space-separated list of up to eight system-defined control plane protocols.
control-plane protocol-group *protocol-group-name*	Matches a control plane protocol group. The *protocol-group-name* argument can be **critical**, **important**, **management**, **monitor**, **normal**, or **redirect**.
customer-dot1p *dot1p-value*&<1-8>	Matches 802.1p priority values in inner VLAN tags of double-tagged packets. The *dot1p-value*&<1-8> argument specifies a space-separated list of up to eight 802.1p priority values. The value range for the *dot1p-value* argument is 0 to 7.
customer-vlan-id *vlan-id-list*	Matches VLAN IDs in inner VLAN tags of double-tagged packets. The *vlan-id-list* argument specifies a space-separated list of up to 10 VLAN items. Each item specifies a VLAN or a range of VLANs in the form of *vlan-id1* **to** *vlan-id2*. The value for *vlan-id2* must be equal to or greater than the value for *vlan-id1*. The value range for the *vlan-id* argument is 1 to 4094.
destination-mac *mac-address*	Matches a destination MAC address.
dscp *dscp-value*&<1-8>	Matches DSCP values. The *dscp-value*&<1-8> argument specifies a space-separated list of up to eight DSCP values. The value range for the *dscp-value* argument is 0 to 63 or keywords shown in Table 10.
ip-precedence *ip-precedence-value*&<1-8>	Matches IP precedence values. The *ip-precedence-value*&<1-8> argument specifies a space-separated list of up to eight IP precedence values. The value range for the *ip-precedence-value* argument is 0 to 7.
protocol *protocol-name*	Matches a protocol. The *protocol-name* argument can be ARP, IP, or IPv6.
qos-local-id *local-id-value*	Matches a local QoS ID in the range of 1 to 4095. The switch supports local QoS IDs in the range of 1 to 3999.
service-dot1p *dot1p-value*&<1-8>	Matches 802.1p priority values in outer VLAN tags. The *dot1p-value*&<1-8> argument specifies a space-separated list of up to eight 802.1p priority values. The value range for the *dot1p-value* argument is 0 to 7.
service-vlan-id *vlan-id-list*	Matches VLAN IDs in outer VLAN tags. The *vlan-id-list* argument specifies a space-separated list of up to 10 VLAN items. Each item specifies a VLAN or a range of VLANs in the form of *vlan-id1* **to** *vlan-id2*. The value for *vlan-id2* must be equal to or greater than the value for *vlan-id1*. The value range for the *vlan-id* argument is 1 to 4094.
source-mac *mac-address*	Matches a source MAC address.

Defining a traffic behavior

A traffic behavior is a set of QoS actions (such as traffic filtering, shaping, policing, and priority marking) to perform on a traffic class.

To define a traffic behavior:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Create a traffic behavior and enter traffic behavior view.	**traffic behavior** *behavior-name*	By default, no traffic behavior is configured.
3. Configure actions in the traffic behavior.	See the subsequent chapters, depending on the purpose of the traffic behavior: traffic policing, traffic filtering, priority marking, traffic accounting, and so on.	By default, no action is configured for a traffic behavior.

Defining a QoS policy

You associate a traffic behavior with a traffic class in a QoS policy to perform the actions defined in the traffic behavior for the traffic class of packets.

When an ACL is used by a QoS policy for traffic classification, the action (permit or deny) in the ACL is ignored, and the actions in the associated traffic behavior are performed.

To associate a traffic class with a traffic behavior in a QoS policy:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Create a QoS policy and enter QoS policy view.	**qos policy** *policy-name*	By default, no QoS policy is configured.
3. Associate a traffic class with a traffic behavior to create a class-behavior association in the QoS policy.	**classifier** *classifier-name* **behavior** *behavior-name* [**mode dcbx** | **insert-before** *before-classifier-name*] *	By default, a traffic class is not associated with a traffic behavior. Repeat this step to create more class-behavior associations. If a class-behavior association has the **mode dcbx** keyword, it applies only to DCBX. For more information about DCBX, see *Layer 2—LAN Switching Configuration Guide*.

Applying the QoS policy

You can apply a QoS policy to the following destinations:

- **Interface**—The QoS policy takes effect on the traffic sent or received on the interface.
- **VLAN**—The QoS policy takes effect on the traffic sent or received on all ports in the VLAN.
- **Globally**—The QoS policy takes effect on the traffic sent or received on all ports.
- **Control plane**—The QoS policy takes effect on the traffic received on the control plane.
- **User profile**—The QoS policy takes effect on the traffic sent or received by the online users of the user profile.

You can modify traffic classes, traffic behaviors, and class-behavior associations in a QoS policy even after it is applied. If a traffic class uses an ACL for traffic classification, you can delete or modify the ACL (such as add rules to, delete rules from, and modify rules of the ACL).

QoS policies applied to an interface, a VLAN, and globally are in descending order of priority. The switch first matches the criteria in the QoS policy applied to an interface. If there is a match, the switch executes the QoS policy applied to the interface and ignores the QoS policies applied to the VLAN and globally.

Applying the QoS policy to an interface

You can apply QoS policies to the following interfaces:

- Ethernet interfaces.
- S-channel interfaces.
- S-channel aggregate interfaces.
- VSI interfaces.
- VSI aggregate interfaces.

For information about the preceding interfaces except Ethernet interfaces, see *EVB Configuration Guide*.

A QoS policy can be applied to multiple interfaces, but only one QoS policy can be applied in one direction (inbound or outbound) of an interface.

The QoS policy applied to the outgoing traffic on an interface does not regulate local packets, which are critical protocol packets sent by the local system for operation maintenance. The most common local packets include link maintenance, routing, LDP, RSVP, and SSH packets.

To apply the QoS policy to an interface:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Enter Ethernet interface view.	**interface** *interface-type interface-number*	N/A
3. Apply the QoS policy to the interface.	**qos apply policy** *policy-name* { **inbound** | **outbound** }	By default, no QoS policy is applied to an interface.

NOTE:

If both packet filtering with the **permit** statement and QoS policies are configured on an interface, the **car** and **filter** actions in the QoS policies do not take effect. For information about packet filtering, see "Configuring ACLs."

Applying the QoS policy to a VLAN

You can apply a QoS policy to a VLAN to regulate traffic of the VLAN.

QoS policies cannot be applied to dynamic VLANs.

To apply the QoS policy to a VLAN:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Apply the QoS policy to VLANs.	**qos vlan-policy** *policy-name* **vlan** *vlan-id-list* { **inbound** | **outbound** }	By default, no QoS policy is applied to a VLAN.

Applying the QoS policy globally

You can apply a QoS policy globally to the inbound or outbound direction of all ports.

To apply the QoS policy globally:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Apply the QoS policy globally.	**qos apply policy** *policy-name* **global** { **inbound** | **outbound** }	By default, no QoS policy is applied globally.

Applying the QoS policy to a control plane

A switch provides the data plane and the control plane.

- **Data plane**—The units (such as various dedicated forwarding chips) at the data plane are responsible for receiving, transmitting, and switching (forwarding) packets. They deliver super processing speeds and throughput.
- **Control plane**—The units (such as CPUs) at the control plane are processing units running most routing and switching protocols. They are responsible for protocol packet resolution and calculation. Compared with data plane units, the control plane units allow for great packet processing flexibility but have lower throughput.

When the data plane receives packets that it cannot recognize or process, it transmits them to the control plane. If the transmission rate exceeds the processing capability of the control plane, the control plane will be busy handling undesired packets and fail to handle legitimate packets correctly or timely. As a result, protocol performance is affected.

To address this problem, apply a QoS policy to the control plane to take QoS actions, such as traffic filtering or rate limiting, on inbound traffic. This ensures that the control plane can correctly receive, transmit, and process packets.

By default, the switch is configured with predefined control plane QoS policies, which take effect on the control planes by default. A predefined control plane QoS policy uses the protocol type or protocol group type to identify the type of packets sent to the control plane. You can use protocol types or protocol group types in **if-match** commands in traffic class view for traffic classification. Then you can reconfigure traffic behaviors for these traffic classes as required. You can use the **display qos policy control-plane pre-defined** command to display predefined control plane QoS policies.

Configuration guidelines

If a QoS policy applied to the control plane uses **if-match control-plane protocol-group** or **if-match control-plane protocol** for traffic classification in a class, the action in the associated traffic behavior can only be **car** or the combination of **car** and **accounting packet**, and only the **cir** keyword in the **car** action can be applied normally.

Configuration procedure

To apply the QoS policy to a control plane:

Step		Command	Remarks
1.	Enter system view.	**system-view**	N/A
2.	Enter control plane view.	**control-plane slot** *slot-number*	N/A
3.	Apply the QoS policy to the control plane.	**qos apply policy** *policy-name* **inbound**	By default, no QoS policy is applied to a control plane.

Applying the QoS policy to a user profile

You can apply a QoS policy to multiple user profiles. In one direction of each user profile, only one policy can be applied. To modify a QoS policy already applied to a user profile, first remove the applied QoS policy.

When you apply a QoS policy to a user profile, follow these restrictions and guidelines:

- The QoS policy supports only the **car** and **accounting** actions in its behaviors.
- The QoS policy cannot be empty, because a user profile configured with an empty QoS policy cannot be activated.
- The switch supports two authentication methods (802.1X and MAC) for online users.

To apply a QoS policy to a user profile:

Step		Command	Remarks
1.	Enter system view.	**system-view**	N/A
2.	Enter user profile view.	**user-profile** *profile-name*	The configuration made in user profile view takes effect only after it is successfully issued to the driver.
3.	Apply the QoS policy.	**qos apply policy** *policy-name* { **inbound** | **outbound** }	By default, no QoS policy is applied to a user profile. Use the **inbound** keyword to apply the QoS policy to the incoming traffic of the device (traffic sent by the online users). Use the **outbound** keyword to apply the QoS policy to the outgoing traffic of the device (traffic received by the online users).

Displaying and maintaining QoS policies

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display traffic class configuration.	**display traffic classifier user-defined** [*classifier-name*] [**slot** *slot-number*]

Display traffic behavior configuration.	**display traffic behavior user-defined** [*behavior-name*] [**slot** *slot-number*]
Display QoS and ACL resource usage.	**display qos-acl resource** [**slot** *slot-number*]
Display QoS policy configuration.	**display qos policy user-defined** [*policy-name* [**classifier** *classifier-name*]] [**slot** *slot-number*]
Display QoS policy configuration on the specified or all interfaces.	**display qos policy interface** [*interface-type interface-number*] [**inbound** | **outbound**]
Display information about QoS policies applied to VLANs.	**display qos vlan-policy** { **name** *policy-name* | **vlan** *vlan-id* } [**slot** *slot-number*] [**inbound** | **outbound**]
Display information about QoS policies applied globally.	**display qos policy global** [**slot** *slot-number*] [**inbound** | **outbound**]
Display information about QoS policies applied to a control plane.	**display qos policy control-plane slot** *slot-number*
Display information about the predefined QoS policy applied to the control plane.	**display qos policy control-plane pre-defined** [**slot** *slot-number*]
Clear the statistics of the QoS policy applied in a certain direction of a VLAN.	**reset qos vlan-policy** [**vlan** *vlan-id*] [**inbound** | **outbound**]
Clear the statistics for a QoS policy applied globally.	**reset qos policy global** [**inbound** | **outbound**]
Clear the statistics for the QoS policy applied to a control plane.	**reset qos policy control-plane slot** *slot-number*

Configuring priority mapping

Overview

When a packet arrives, a device assigns a set of QoS priority parameters to the packet based on either a priority field carried in the packet or the port priority of the incoming port. This process is called priority mapping. During this process, the device can modify the priority of the packet according to the priority mapping rules. The set of QoS priority parameters decides the scheduling priority and forwarding priority of the packet.

Priority mapping is implemented with priority maps and involves the following priorities:

- 802.1p priority.
- DSCP.
- IP precedence.
- Local precedence.
- Drop priority.

Introduction to priorities

Priorities include the following types: priorities carried in packets, and priorities locally assigned for scheduling only.

Packet-carried priorities include 802.1p priority, DSCP precedence, and IP precedence. These priorities have global significance and affect the forwarding priority of packets across the network. For more information about these priorities, see "Appendixes."

Locally assigned priorities only have local significance. They are assigned by the switch only for scheduling. These priorities include the local precedence and drop priority, as follows:

- **Local precedence**—Used for queuing. A local precedence value corresponds to an output queue. A packet with higher local precedence is assigned to a higher priority output queue to be preferentially scheduled.
- **Drop priority**—Used for making packet drop decisions. Packets with the highest drop priority are dropped preferentially.

Priority maps

The switch provides various types of priority maps. By looking through a priority map, the switch decides which priority value to assign to a packet for subsequent packet processing. The switch provides the following priority mapping tables:

- **dot1p-dp**—802.1p-to-drop priority mapping table.
- **dot1p-lp**—802.1p-to-local priority mapping table.
- **dscp-dot1p**—DSCP-to-802.1p priority mapping table, which is applicable only to IP packets.
- **dscp-dp**—DSCP-to-drop priority mapping table, which is applicable only to IP packets.
- **dscp-dscp**—DSCP-to-DSCP priority mapping table, which is applicable only to IP packets.

The default priority maps (as shown in "Appendix A Default priority maps") are available for priority mapping. They are adequate in most cases. If a default priority map cannot meet your requirements, you can modify the priority map as required.

Priority trust mode on a port

The priority trust mode on a port determines which priority is used for priority mapping table lookup. Port priority was introduced to use for priority mapping in addition to the priority fields carried in packets. The Switch Series provides the following priority trust modes:

- Using the 802.1p priority carried in packets for priority mapping.

Table 3 Priority mapping results of trusting the 802.1p priority (when the default dot1p-lp priority mapping table is used)

802.1p priority carried in packets	Local precedence	Queue ID
0	2	2
1	0	0
2	1	1
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7

NOTE:

When the 802.1p priority carried in packets is trusted, the port priority is used for priority mapping for packets which do not carry VLAN tags (namely, do not carry 802.1p priorities.) The priority mapping results are the same as not trusting packet priority, as shown in Table 5.

- Using the DSCP carried in packets for priority mapping.

Table 4 Priority mapping results of trusting the DSCP (when the default dscp-dot1p and dot1p-lp priority mapping tables are used)

DSCP value carried in packets	Local precedence	Queue ID
0 to 7	2	2
8 to 15	0	0
16 to 23	1	1
24 to 31	3	3
32 to 39	4	4
40 to 47	5	5
48 to 55	6	6
56 to 63	7	7

- Using the port priority as the 802.1p priority for priority mapping. The port priority is user configurable.

Table 5 Priority mapping results of not trusting packet priority (when the default dot1p-lp priority mapping table is used)

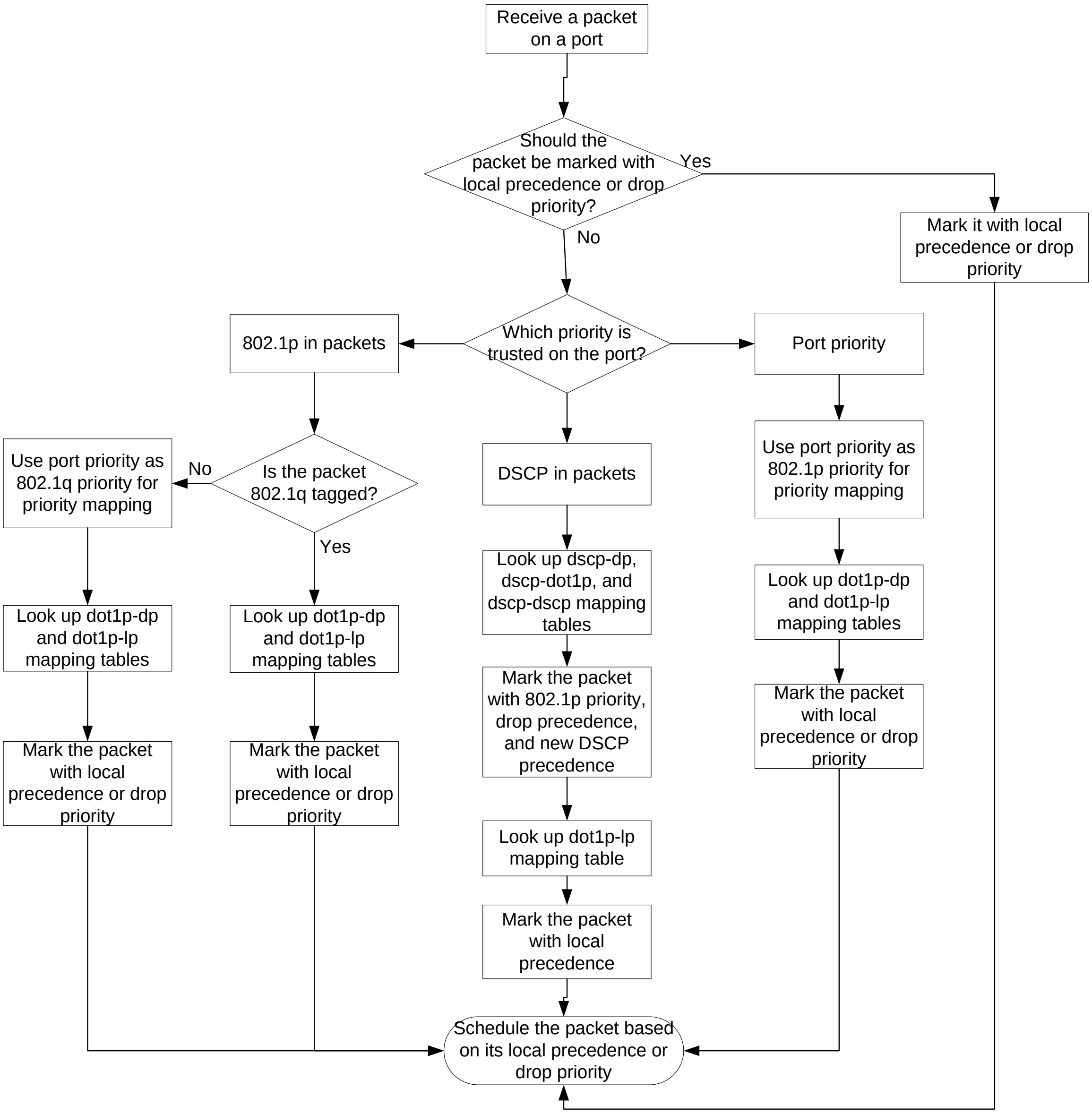
Port priority	Local precedence	Queue ID
0 (default)	2	2
1	0	0
2	1	1
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7

The priority mapping process varies with priority trust mode. For more information, see the subsequent section.

Priority mapping process

On receiving an Ethernet packet on a port, the switch marks the scheduling priorities (local precedence and drop precedence) for the Ethernet packet. This procedure is done according to the priority trust mode of the receiving port and the 802.1Q tagging status of the packet, as shown in Figure 4.

Figure 4 Priority mapping process for an Ethernet packet


Receive a packet on a port
Should the packet be marked with local precedence or drop priority?
Yes
No
Mark it with local precedence or drop priority
Which priority is trusted on the port?
802.1p in packets
DSCP in packets
Port priority
Is the packet 802.1q tagged?
No
Yes
Use port priority as 802.1q priority for priority mapping
Look up dot1p-dp and dot1p-lp mapping tables
Mark the packet with local precedence or drop priority
Look up dot1p-dp and dot1p-lp mapping tables
Mark the packet with local precedence or drop priority
Look up dscp-dp, dscp-dot1p, and dscp-dscp mapping tables
Mark the packet with 802.1p priority, drop precedence, and new DSCP precedence
Look up dot1p-lp mapping table
Mark the packet with local precedence
Use port priority as 802.1p priority for priority mapping
Look up dot1p-dp and dot1p-lp mapping tables
Mark the packet with local precedence or drop priority
Schedule the packet based on its local precedence or drop priority

For information about priority marking, see "Configuring priority marking."

Priority mapping configuration tasks

You can modify priority mappings by modifying priority mapping tables, priority trust mode on a port, and port priority.

To configure priority mapping, perform the following tasks:

Tasks at a glance
(Optional.) Configuring a priority map

Tasks at a glance
(Required.) Perform one of the following tasks: • Configuring an interface to trust packet priority for priority mapping • Changing the port priority of an interface

Configuring a priority map

Step		Command	Remarks
1.	Enter system view.	**system-view**	N/A
2.	Enter priority map view.	**qos map-table** { **dot1p-dp** | **dot1p-lp** | **dscp-dot1p** | **dscp-dp** | **dscp-dscp** }	N/A
3.	Configure mappings for the priority map.	**import** *import-value-list* **export** *export-value*	By default, the default priority maps are used. For more information, see "Appendixes." Newly configured mappings overwrite the old ones.

Configuring an interface to trust packet priority for priority mapping

You can configure the switch to trust a particular priority field carried in packets for priority mapping on interfaces.

When you configure the following trusted packet priority type on an interface, use the following available keywords:

- **dot1p**—Uses the 802.1p priority of received packets for mapping.
- **dscp**—Uses the DSCP precedence of received IP packets for mapping.
- **none**—Uses the port priority as the 802.1p priority for mapping.

To configure the trusted packet priority type on an interface:

Step		Command	Remarks
1.	Enter system view.	**system-view**	N/A
2.	Enter Ethernet interface view.	**interface** *interface-type interface-number*	N/A

3. Configure the trusted packet priority type.	• Configure the interface to trust the DSCP precedence. **qos trust dscp** • Configure the interface to trust the 802.1p priority of received packets. **qos trust dot1p** • Configure the interface not to trust any packet priority. **undo qos trust**	Use one of these commands. By default, an interface does not trust any packet priority.

Changing the port priority of an interface

If an interface does not trust any packet priority, the switch uses its port priority to look for the set of priority parameters for the incoming packets. By changing port priority, you can prioritize traffic received on different interfaces.

To change the port priority of an interface:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Enter Ethernet interface view.	**interface** *interface-type interface-number*	N/A
3. Set the port priority of the interface.	**qos priority** *priority-value*	The default setting is 0.

Displaying and maintaining priority mapping

Execute **display** commands in any view.

Task	Command
Display priority map configuration.	**display qos map-table** { **dot1p-dp** | **dot1p-lp** | **dscp-dot1p** | **dscp-dp** | **dscp-dscp** }
Display the trusted packet priority type on a port.	**display qos trust interface** [*interface-type interface-number*]

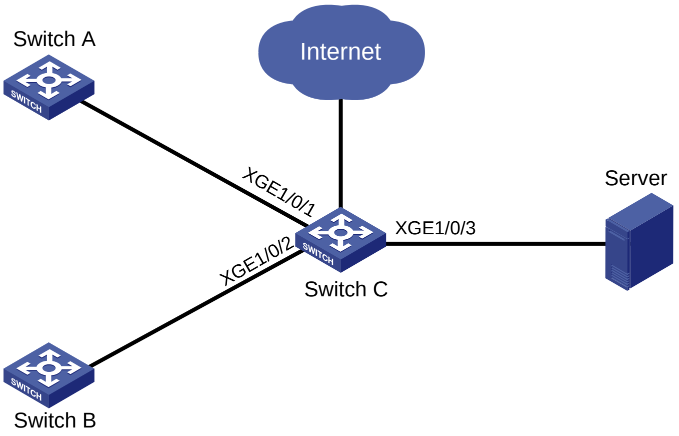
Priority trust mode configuration example

Network requirements

As shown in Figure 5, the packets from Switch A and Switch B to Switch C are not VLAN tagged.

Configure Switch C to preferentially process packets from Switch A to Server when Ten-GigabitEthernet 1/0/3 of Switch C is congested.

Figure 5 Network diagram



Configuration procedure

# Assign port priority to Ten-GigabitEthernet 1/0/1 and Ten-GigabitEthernet 1/0/2. Make sure the following requirements are met:

- The port priority of Ten-GigabitEthernet 1/0/1 is higher than that of Ten-GigabitEthernet 1/0/2.
- No trusted packet priority type is configured on Ten-GigabitEthernet 1/0/1 and Ten-GigabitEthernet 1/0/2.

```
<SwitchC> system-view
[SwitchC] interface ten-gigabitethernet 1/0/1
[SwitchC-Ten-GigabitEthernet1/0/1] qos priority 3
[SwitchC-Ten-GigabitEthernet1/0/1] quit
[SwitchC] interface ten-gigabitethernet 1/0/2
[SwitchC-Ten-GigabitEthernet1/0/2] qos priority 1
[SwitchC-Ten-GigabitEthernet1/0/2] quit
```

Priority mapping table and priority marking configuration example

Network requirements

As shown in Figure 6:

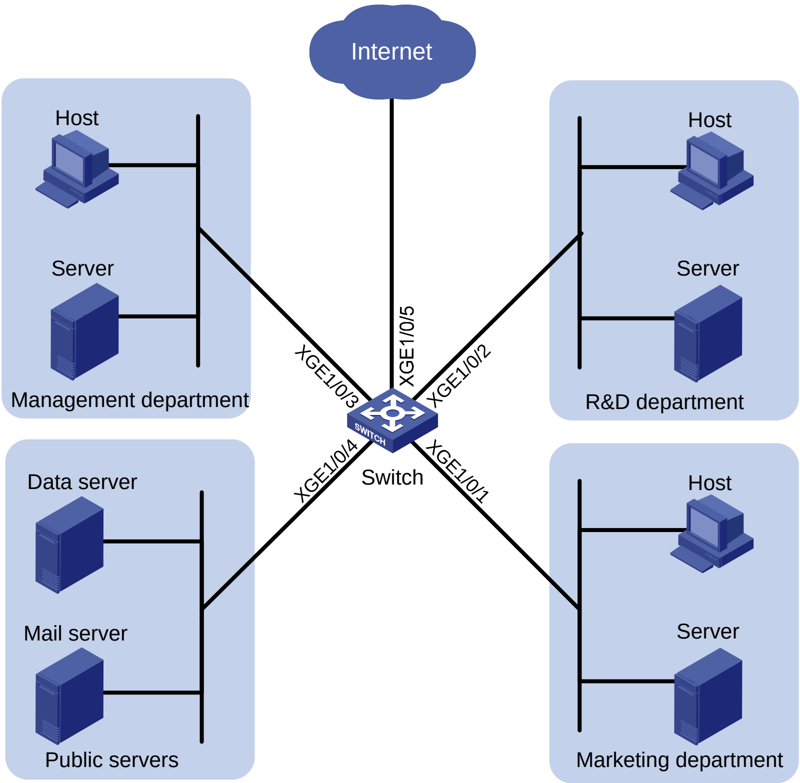
- The Marketing department connects to Ten-GigabitEthernet 1/0/1 of the device, which sets the 802.1p priority of traffic from the Marketing department to 3.
- The R&D department connects to Ten-GigabitEthernet 1/0/2 of the switch, which sets the 802.1p priority of traffic from the R&D department to 4.
- The Management department connects to Ten-GigabitEthernet 1/0/3 of the switch, which sets the 802.1p priority of traffic from the Management department to 5.

Configure port priority, 802.1p-to-local mapping table, and priority marking to implement the plan as described in Table 6.

Table 6 Configuration plan

Traffic destination	Traffic priority order	Queuing plan		
		Traffic source	Output queue	Queue priority
Public servers	R&D department > Management department > Marketing department	R&D department	6	High
		Management department	4	Medium
		Marketing department	2	Low
Internet	Management department > Marketing department > R&D department	R&D department	2	Low
		Management department	6	High
		Marketing department	4	Medium

Figure 6 Network diagram



Configuration procedure

1. Enable trusting port priority:

   # Set the port priority of Ten-GigabitEthernet 1/0/1 to 3.

```
<Switch> system-view
[Switch] interface ten-gigabitethernet 1/0/1
[Switch-Ten-GigabitEthernet1/0/1] qos priority 3
[Switch-Ten-GigabitEthernet1/0/1] quit
```

   # Set the port priority of Ten-GigabitEthernet 1/0/2 to 4.

```
[Switch] interface ten-gigabitethernet 1/0/2
[Switch-Ten-GigabitEthernet1/0/2] qos priority 4
[Switch-Ten-GigabitEthernet1/0/2] quit
```

# Set the port priority of Ten-GigabitEthernet 1/0/3 to 5.

```
[Switch] interface ten-gigabitethernet 1/0/3
[Switch-Ten-GigabitEthernet1/0/3] qos priority 5
[Switch-Ten-GigabitEthernet1/0/3] quit
```

2. Configure the 802.1p-to-local mapping table to map 802.1p priority values 3, 4, and 5 to local precedence values 2, 6, and 4. This guarantees the R&D department, management department, and marketing department decreased priorities to access the public server.

```
[Switch] qos map-table dot1p-lp
[Switch-maptbl-dot1p-lp] import 3 export 2
[Switch-maptbl-dot1p-lp] import 4 export 6
[Switch-maptbl-dot1p-lp] import 5 export 4
[Switch-maptbl-dot1p-lp] quit
```

3. Configure priority marking:

# Mark the HTTP traffic of the management department, marketing department, and R&D department to the Internet with 802.1p priorities 4, 5, and 3, respectively. Use the priority mapping table you have configured to map the 802.1p priorities to local precedence values 6, 4, and 2, respectively, for differentiated traffic treatment.

# Create ACL 3000 to match HTTP traffic.

```
[Switch] acl number 3000
[Switch-acl-adv-3000] rule permit tcp destination-port eq 80
[Switch-acl-adv-3000] quit
```

# Create class **http** and use ACL 3000 in the class.

```
[Switch] traffic classifier http
[Switch-classifier-http] if-match acl 3000
[Switch-classifier-http] quit
```

# Configure a priority marking policy for the management department, and apply the policy to the incoming traffic of Ten-GigabitEthernet 1/0/3.

```
[Switch] traffic behavior admin
[Switch-behavior-admin] remark dot1p 4
[Switch-behavior-admin] quit
[Switch] qos policy admin
[Switch-qospolicy-admin] classifier http behavior admin
[Switch-qospolicy-admin] quit
[Switch] interface ten-gigabitethernet 1/0/3
[Switch-Ten-GigabitEthernet1/0/3] qos apply policy admin inbound
```

# Configure a priority marking policy for the marketing department, and apply the policy to the incoming traffic of Ten-GigabitEthernet 1/0/1.

```
[Switch] traffic behavior market
[Switch-behavior-market] remark dot1p 5
[Switch-behavior-market] quit
[Switch] qos policy market
[Switch-qospolicy-market] classifier http behavior market
[Switch-qospolicy-market] quit
[Switch] interface ten-gigabitethernet 1/0/1
```

```
[Switch-Ten-GigabitEthernet1/0/1] qos apply policy market inbound
```

# Configure a priority marking policy for the R&D department, and apply the policy to the incoming traffic of Ten-GigabitEthernet 1/0/2.

```
[Switch] traffic behavior rd
[Switch-behavior-rd] remark dot1p 3
[Switch-behavior-rd] quit
[Switch] qos policy rd
[Switch-qospolicy-rd] classifier http behavior rd
[Switch-qospolicy-rd] quit
[Switch] interface ten-gigabitethernet 1/0/2
[Switch-Ten-GigabitEthernet1/0/2] qos apply policy rd inbound
```

Configuring traffic policing, GTS, and rate limit

Overview

Traffic policing helps assign network resources (including bandwidth) and increase network performance. For example, you can configure a flow to use only the resources committed to it in a certain time range. This avoids network congestion caused by burst traffic.

Traffic policing, Generic Traffic Shaping (GTS), and rate limit control the traffic rate and resource usage according to traffic specifications. You can use token buckets for evaluating traffic specifications.

Traffic evaluation and token buckets

Token bucket features

A token bucket is analogous to a container that holds a certain number of tokens. Each token represents a certain forwarding capacity. The system puts tokens into the bucket at a constant rate. When the token bucket is full, the extra tokens cause the token bucket to overflow.

Evaluating traffic with the token bucket

A token bucket mechanism evaluates traffic by looking at the number of tokens in the bucket. If the number of tokens in the bucket is enough for forwarding the packets, the following events occur:

- The traffic conforms to the specification (called conforming traffic).
- The corresponding tokens are taken away from the bucket.

Otherwise, the traffic does not conform to the specification (called excess traffic).

A token bucket has the following configurable parameters:

- Mean rate at which tokens are put into the bucket, which is the permitted average rate of traffic. It is usually set to the committed information rate (CIR).
- Burst size or the capacity of the token bucket. It is the maximum traffic size permitted in each burst. It is usually set to the committed burst size (CBS). The set burst size must be greater than the maximum packet size.

Each arriving packet is evaluated.

Complicated evaluation

You can set two token buckets, bucket C and bucket E, to evaluate traffic in a more complicated environment and achieve more policing flexibility. For example, traffic policing can use the following mechanisms:

- **Single rate two color**—Uses one token bucket and the following parameters:
  - **CIR**—Rate at which tokens are put into bucket C. It sets the average packet transmission or forwarding rate allowed by bucket C.
  - **CBS**—Size of bucket C, which specifies the transient burst of traffic that bucket C can forward.

  When a packet arrives, the following rules apply:

  - If bucket C has enough tokens to forward the packet, the packet is colored green.

    - Otherwise, the packet is colored red.
- **Single rate three color**—Uses two token buckets and the following parameters:
    - **CIR**—Rate at which tokens are put into bucket C. It sets the average packet transmission or forwarding rate allowed by bucket C.
    - **CBS**—Size of bucket C, which specifies the transient burst of traffic that bucket C can forward.
    - **EBS**—Size of bucket E minus size of bucket C. The EBS specifies the transient burst of traffic that bucket E can forward. The EBS cannot be 0. The size of E bucket is the sum of the CBS and EBS.

  When a packet arrives, the following rules apply:
    - If bucket C has enough tokens, the packet is colored green.
    - If bucket C does not have enough tokens but bucket E has enough tokens, the packet is colored yellow.
    - If neither bucket C nor bucket E has enough tokens, the packet is colored red.
- **Two rate three color**—Uses two token buckets and the following parameters:
    - **CIR**—Rate at which tokens are put into bucket C. It sets the average packet transmission or forwarding rate allowed by bucket C.
    - **CBS**—Size of bucket C, which specifies the transient burst of traffic that bucket C can forward.
    - **PIR**—Rate at which tokens are put into bucket E, which specifies the average packet transmission or forwarding rate allowed by bucket E.
    - **EBS**—Size of bucket E, which specifies the transient burst of traffic that bucket E can forward.

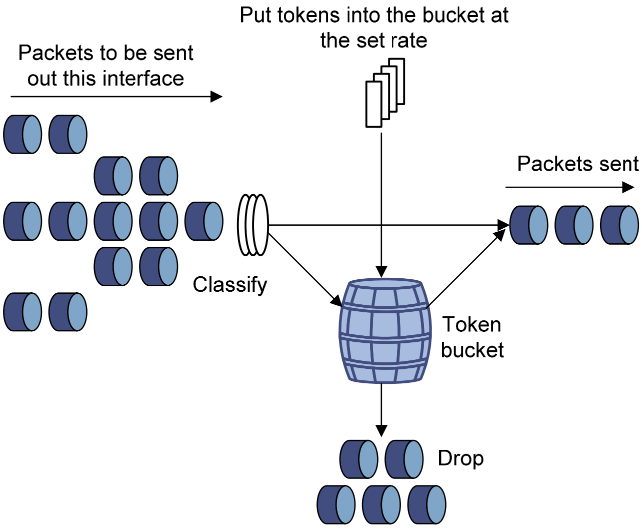
  When a packet arrives, the following rules apply:
    - If bucket C has enough tokens, the packet is colored green.
    - If bucket C does not have enough tokens but bucket E has enough tokens, the packet is colored yellow.
    - If neither bucket C nor bucket E has enough tokens, the packet is colored red.

Traffic policing

Traffic policing supports policing the inbound traffic and the outbound traffic.

A typical application of traffic policing is to supervise the specification of traffic entering a network and limit it within a reasonable range. Another application is to "discipline" the extra traffic to prevent aggressive use of network resources by an application. For example, you can limit bandwidth for HTTP packets to less than 50% of the total. If the traffic of a session exceeds the limit, traffic policing can drop the packets or reset the IP precedence of the packets. Figure 7 shows an example of policing outbound traffic on an interface.

Figure 7 Traffic policing


Put tokens into the bucket at the set rate
Packets to be sent out this interface
Packets sent
Classify
Token bucket
Drop

Traffic policing is widely used in policing traffic entering the ISP networks. It can classify the policed traffic and take predefined policing actions on each packet depending on the evaluation result as follows:

- Forwarding the packet if the evaluation result is "conforming."
- Dropping the packet if the evaluation result is "excess."
- Forwarding the packet with its precedence re-marked if the evaluation result is "conforming." Priorities that can be re-marked include 802.1p priority, DSCP precedence, and local precedence.

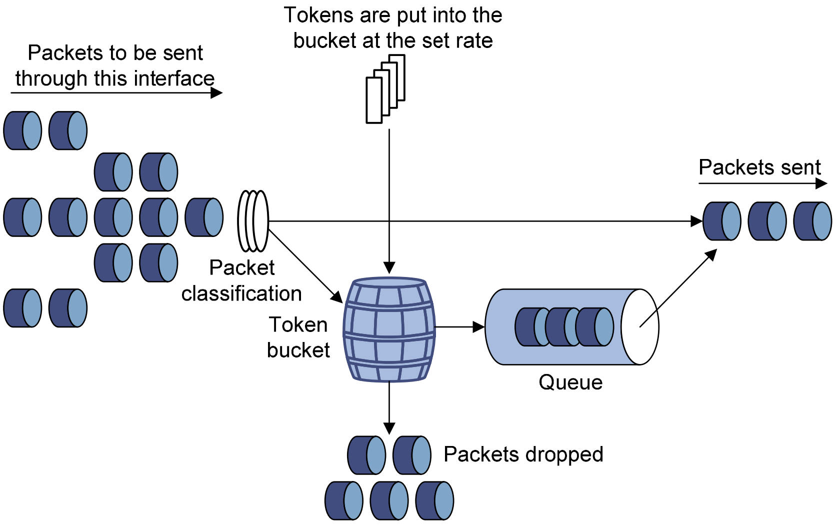
GTS

GTS supports shaping the outbound traffic. GTS limits the outbound traffic rate by buffering exceeding traffic. You can use GTS to adapt the traffic output rate on a device to the input traffic rate of its connected device to avoid packet loss.

The differences between traffic policing and GTS are as follows:

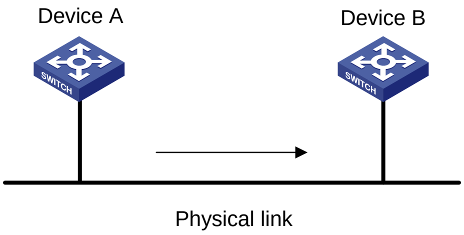
- Packets to be dropped with traffic policing are retained in a buffer or queue with GTS, as shown in Figure 8. When enough tokens are in the token bucket, the buffered packets are sent at an even rate.
- GTS can result in additional delay and traffic policing does not.

Figure 8 GTS


Tokens are put into the bucket at the set rate
Packets to be sent through this interface
Packets sent
Packet classification
Token bucket
Queue
Packets dropped

For example, in Figure 9, Switch B performs traffic policing on packets from Switch A and drops packets exceeding the limit. To avoid packet loss, you can perform GTS on the outgoing interface of Switch A so that packets exceeding the limit are cached in Switch A. Once resources are released, GTS takes out the cached packets and sends them out.

Figure 9 GTS application



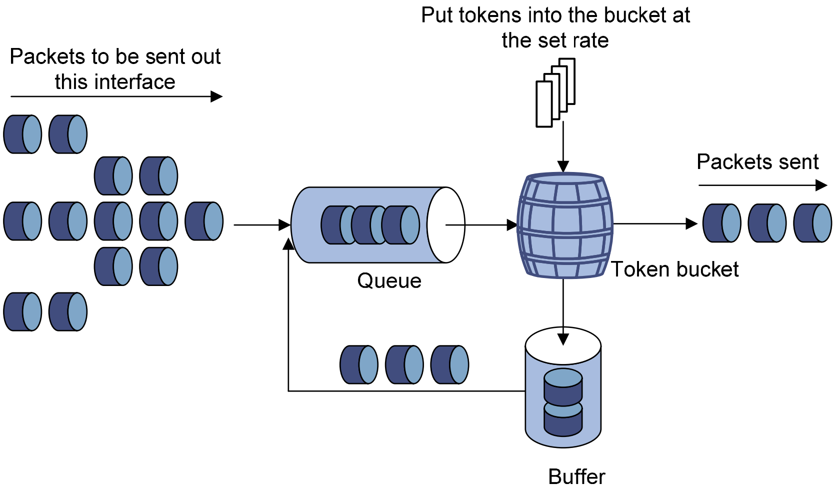
Rate limit

Rate limit controls the rate of inbound and outbound traffic. The outbound traffic is taken for example.

The rate limit of a physical interface specifies the maximum rate for sending or receiving packets (including critical packets).

Rate limit also uses token buckets for traffic control. When rate limit is configured on an interface, a token bucket handles all packets to be sent through the interface for rate limiting. If enough tokens are in the token bucket, packets can be forwarded. Otherwise, packets are put into QoS queues for congestion management. In this way, the traffic passing the physical interface is controlled.

Figure 10 Rate limit implementation


Put tokens into the bucket at the set rate
Packets to be sent out this interface
Packets sent
Queue
Token bucket
Buffer

The token bucket mechanism limits traffic rate when accommodating bursts. It allows bursty traffic to be transmitted if enough tokens are available. If tokens are scarce, packets cannot be transmitted until efficient tokens are generated in the token bucket. It restricts the traffic rate to the rate for generating tokens.

Rate limit controls the total rate of all packets on a physical interface. It is easier to use than traffic policing in controlling the total traffic rate on a physical interface.

Configuring traffic policing

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Create a traffic class and enter traffic class view.	**traffic classifier** *classifier-name* [**operator** { **and** | **or** }]	By default, no traffic class is configured.
3. Configure match criteria.	**if-match** *match-criteria*	By default, no match criterion is configured. For more information about the **if-match** command, see *ACL and QoS Command Reference*.
4. Return to system view.	**quit**	N/A
5. Create a traffic behavior and enter traffic behavior view.	**traffic behavior** *behavior-name*	By default, no traffic behavior is configured.

Step	Command	Remarks
6. Configure a traffic policing action.	**car cir** *committed-information-rate* [**cbs** *committed-burst-size* [**ebs** *excess-burst-size*]] [**green** *action* | **red** *action* | **yellow** *action*] * **car cir** *committed-information-rate* [**cbs** *committed-burst-size*] **pir** *peak-information-rate* [**ebs** *excess-burst-size*] [**green** *action* | **red** *action* | **yellow** *action*] *	Use either of the commands. By default, no traffic policing action is configured.
7. Return to system view.	**quit**	N/A
8. Create a QoS policy and enter QoS policy view.	**qos policy** *policy-name*	By default, no QoS policy is configured.
9. Associate the traffic class with the traffic behavior in the QoS policy.	**classifier** *classifier-name* **behavior** *behavior-name* [**insert-before** *before-classifier-name*]	By default, a traffic class is not associated with a traffic behavior.
10. Return to system view.	**quit**	N/A
11. Apply the QoS policy.	• Applying the QoS policy to an interface • Applying the QoS policy to a VLAN • Applying the QoS policy globally • Applying the QoS policy to a control plane • Applying the QoS policy to a user profile	Choose one of the application destinations as needed. By default, no QoS policy is applied.
12. (Optional.) Display traffic policing configuration.	**display traffic behavior user-defined** [*behavior-name*]	Available in any view.

Configuring GTS

The switch supports configuring queue-based GTS by using the non-MQC approach. In queue-based GTS, you set GTS parameters for packets of a queue.

To configure GTS:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Enter Ethernet interface view.	**interface** *interface-type interface-number*	N/A
3. Configure GTS for a queue.	**qos gts queue** *queue-id* **cir** *committed-information-rate* [**cbs** *committed-burst-size*]	By default, GTS is not configured on an interface.

Configuring the rate limit

The rate limit of a physical interface specifies the maximum rate of incoming packets or outgoing packets.

To configure the rate limit:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Enter Ethernet interface view.	**interface** *interface-type interface-number*	N/A
3. Configure the rate limit for the interface.	**qos lr** { **inbound** | **outbound** } **cir** *committed-information-rate* [**cbs** *committed-burst-size*]	By default, rate limit is not configured on an interface.

Displaying and maintaining traffic policing, GTS, and rate limit

Execute **display** commands in any view.

Task	Command
Display QoS and ACL resource usage.	**display qos-acl resource** [**slot** *slot-number*]
Display traffic behavior configuration.	**display traffic behavior user-defined** [*behavior-name*]
Display GTS configuration on an interface.	**display qos gts interface** [*interface-type interface-number*]
Display rate limit configuration on an interface.	**display qos lr interface** [*interface-type interface-number*]

Traffic policing configuration example

Network requirements

As shown in Figure 11, configure traffic policing on Ten-GigabitEthernet 1/0/1 of Switch A to meet the following requirements:

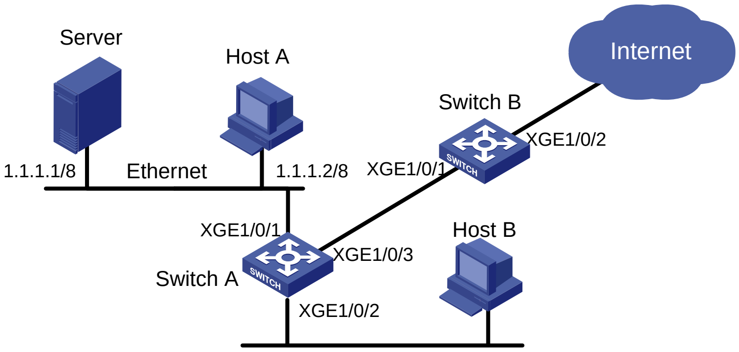
- Limit the rate of incoming traffic from the server to 102400 kbps: Transmit the conforming traffic normally, mark the excess traffic with DSCP value 0, and then transmit the traffic.
- Limit the rate of incoming traffic from Host A to 25600 kbps: Transmit the conforming traffic normally, and drop the excess traffic.

Configure traffic policing on Ten-GigabitEthernet 1/0/1 and Ten-GigabitEthernet 1/0/2 of Switch B to meet the following requirements:

- Limit the total incoming traffic rate of Ten-GigabitEthernet 1/0/1 to 204800 kbps and drop the excess traffic.

- Limit the outgoing HTTP traffic (traffic accessing the Internet) rate of Ten-GigabitEthernet 1/0/2 to 102400 kbps and drop the excess traffic.

Figure 11 Network diagram



Configuration procedures

1. Configure Switch A:

# Configure ACL 2001 and ACL 2002 to match traffic from the server and Host A, respectively.

```
<SwitchA> system-view
[SwitchA] acl number 2001
[SwitchA-acl-basic-2001] rule permit source 1.1.1.1 0
[SwitchA-acl-basic-2001] quit
[SwitchA] acl number 2002
[SwitchA-acl-basic-2002] rule permit source 1.1.1.2 0
[SwitchA-acl-basic-2002] quit
```

# Create a class named **server** and use ACL 2001 as the match criterion. Create a class named **host** and use ACL 2002 as the match criterion.

```
[SwitchA] traffic classifier server
[SwitchA-classifier-server] if-match acl 2001
[SwitchA-classifier-server] quit
[SwitchA] traffic classifier host
[SwitchA-classifier-host] if-match acl 2002
[SwitchA-classifier-host] quit
```

# Create a behavior named **server** and configure the CAR action for the behavior as follows: Set the CIR to 102400 kbps, and mark the excess packets (red packets) with DSCP value 0 and transmit them.

```
[SwitchA] traffic behavior server
[SwitchA-behavior-server] car cir 102400 red remark-dscp-pass 0
[SwitchA-behavior-server] quit
```

# Create a behavior named **host** and configure the CAR action for the behavior as follows: Set the CIR to 25600 kbps.

```
[SwitchA] traffic behavior host
[SwitchA-behavior-host] car cir 25600
[SwitchA-behavior-host] quit
```

# Create a QoS policy named **car** and associate class **server** with behavior **server** and class **host** with behavior **host**.

```
[SwitchA] qos policy car
[SwitchA-qospolicy-car] classifier server behavior server
[SwitchA-qospolicy-car] classifier host behavior host
[SwitchA-qospolicy-car] quit
```

# Apply QoS policy **car** to the incoming traffic of port Ten-GigabitEthernet 1/0/1.

```
[SwitchA] interface Ten-GigabitEthernet 1/0/1
[SwitchA-Ten-GigabitEthernet1/0/1] qos apply policy car inbound
```

2. Configure Switch B:

# Configure advanced ACL 3001 to match HTTP traffic.

```
<SwitchB> system-view
[SwitchB] acl number 3001
[SwitchB-acl-adv-3001] rule permit tcp destination-port eq 80
[SwitchB-acl-adv-3001] quit
```

# Create a class named **http** and use ACL 3001 as the match criterion.

```
[SwitchB] traffic classifier http
[SwitchB-classifier-http] if-match acl 3001
[SwitchB-classifier-http] quit
```

# Create a class named **class** and configure the class to match all packets.

```
[SwitchB] traffic classifier class
[SwitchB-classifier-class] if-match any
[SwitchB-classifier-class] quit
```

# Create a behavior named **car_inbound** and configure the CAR action for the behavior as follows: Set the CIR to 204800 kbps.

```
[SwitchB] traffic behavior car_inbound
[SwitchB-behavior-car_inbound] car cir 204800
[SwitchB-behavior-car_inbound] quit
```

# Create a behavior named **car_outbound** and configure a CAR action for the behavior as follows: Set the CIR to 102400 kbps.

```
[SwitchB] traffic behavior car_outbound
[SwitchB-behavior-car_outbound] car cir 102400
[SwitchB-behavior-car_outbound] quit
```

# Create a QoS policy named **car_inbound** and associate class **class** with traffic behavior **car_inbound** in the QoS policy.

```
[SwitchB] qos policy car_inbound
[SwitchB-qospolicy-car_inbound] classifier class behavior car_inbound
[SwitchB-qospolicy-car_inbound] quit
```

# Create a QoS policy named **car_outbound** and associate class **http** with traffic behavior **car_outbound** in the QoS policy.

```
[SwitchB] qos policy car_outbound
[SwitchB-qospolicy-car_outbound] classifier http behavior car_outbound
[SwitchB-qospolicy-car_outbound] quit
```

# Apply QoS policy **car_inbound** to the incoming traffic of port Ten-GigabitEthernet 1/0/1.

```
[SwitchB] interface Ten-GigabitEthernet 1/0/1
[SwitchB-Ten-GigabitEthernet1/0/1] qos apply policy car_inbound inbound
```

# Apply QoS policy **car_outbound** to the outgoing traffic of port Ten-GigabitEthernet 1/0/2.

```
[SwitchB] interface Ten-GigabitEthernet 1/0/2
```

```
[SwitchB-Ten-GigabitEthernet1/0/2] qos apply policy car_outbound outbound
```

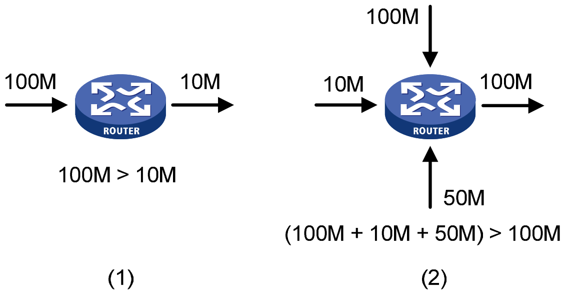
Configuring congestion management

Overview

Congestion occurs on a link or node when traffic size exceeds the processing capability of the link or node. It is typical of a statistical multiplexing network and can be caused by link failures, insufficient resources, and various other causes.

Figure 12 shows two typical congestion scenarios.

Figure 12 Traffic congestion scenarios



Congestion produces the following negative results:

- Increased delay and jitter during packet transmission.
- Decreased network throughput and resource use efficiency.
- Network resource (memory, in particular) exhaustion and even system breakdown.

Congestion is unavoidable in switched networks and multiuser application environments. To improve the service performance of your network, take measures to manage and control it.

The key to congestion management is defining a resource dispatching policy to prioritize packets for forwarding when congestion occurs.

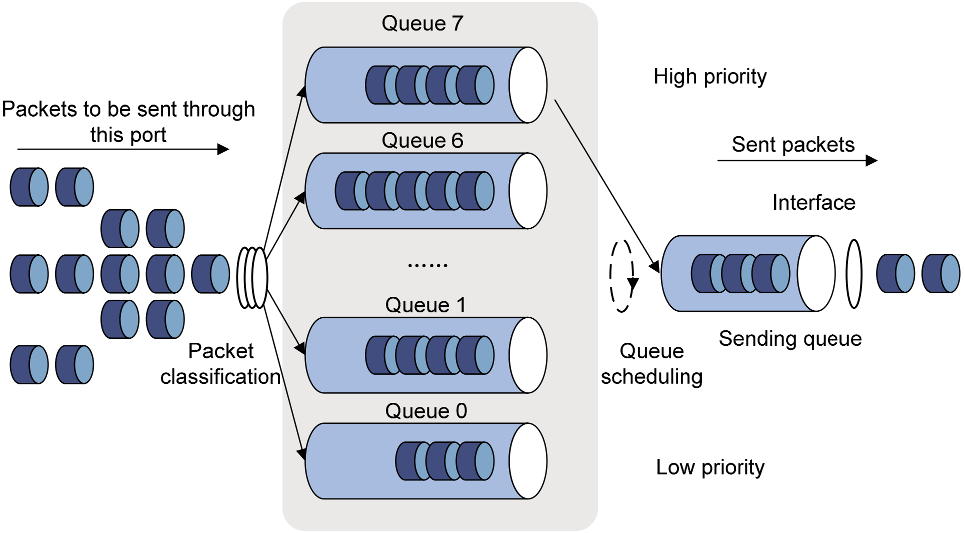
Congestion management uses queuing and scheduling algorithms to classify and sort traffic leaving a port.

The Switch Series supports the following queue-scheduling mechanisms.

SP queuing

SP queuing is designed for mission-critical applications that require preferential service to reduce the response delay when congestion occurs.

Figure 13 SP queuing



In Figure 13, SP queuing classifies eight queues on a port into eight classes, numbered 7 to 0 in descending priority order.

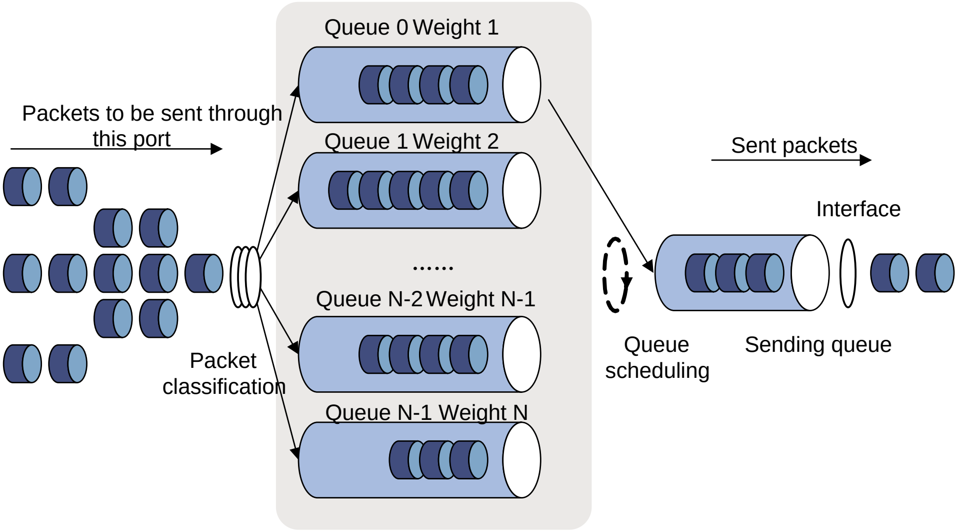
SP queuing schedules the eight queues in descending order of priority. SP queuing sends packets in the queue with the highest priority first. When the queue with the highest priority is empty, it sends packets in the queue with the second highest priority, and so on. You can assign mission-critical packets to a high priority queue to make sure they are always served first. Common service packets can be assigned to low priority queues to be transmitted when high priority queues are empty.

The disadvantage of SP queuing is that packets in the lower priority queues cannot be transmitted if packets exist in the higher priority queues for a long time. In the worst case, lower priority traffic might never get serviced.

WRR queuing

WRR queuing schedules all the queues in turn to ensure that every queue is served for a certain time, as shown in Figure 14.

Figure 14 WRR queuing



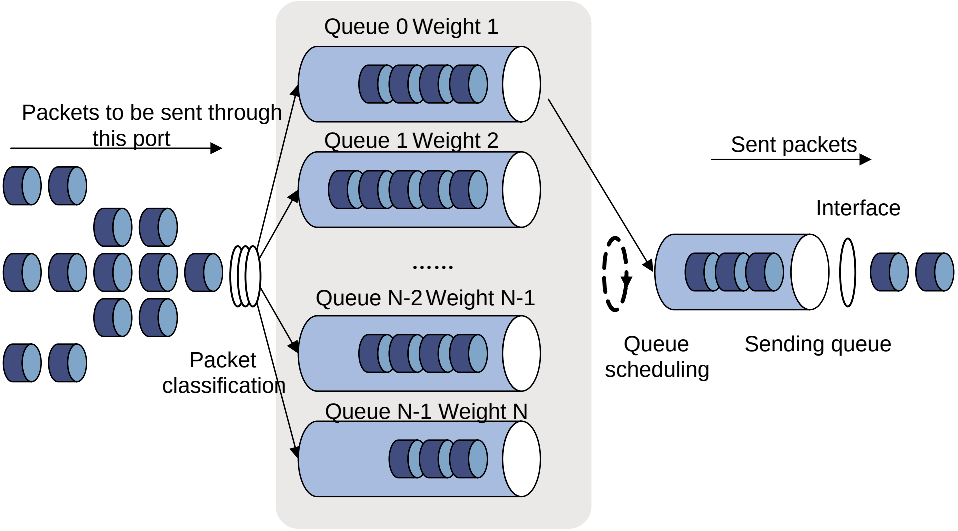
Assume a port provides eight output queues. WRR assigns each queue a weight value (represented by w7, w6, w5, w4, w3, w2, w1, or w0) to decide the proportion of resources assigned to the queue.

The switch implements the weight of a queue by scheduling a certain number of bytes (byte-count WRR) or packets (packet-based WRR) for that queue. Take byte-count WRR as an example: On a 10 Gbps port, you can configure the weight values of WRR queuing to 5, 5, 3, 3, 1, 1, 1, and 1 (corresponding to w7, w6, w5, w4, w3, w2, w1, and w0, respectively). In this way, the queue with the lowest priority can get a minimum of 500 Mbps of bandwidth. WRR solves the problem that SP queuing might fail to serve packets in low-priority queues for a long time.

The switch supports WRR priority queue groups. You can assign the output queues to WRR priority queue group 1 and WRR priority queue group 2. You can set the weight for each queue and WRR schedules queues in each group based on the weights in a round robin manner. WRR schedules the traffic of group 1 and the traffic of group 2 in the ratio of 1:1.

WFQ queuing

Figure 15 WFQ queuing



WFQ is similar to WRR. The difference is that WFQ enables you to set guaranteed bandwidth that a WFQ queue can get during congestion.

The switch supports WFQ priority queue groups. You can assign the output queues to WFQ priority queue group 1 and WFQ priority queue group 2. You can configure the weight for each queue and WFQ schedules queues in each group based on the weights in a round robin manner. WFQ schedules the traffic of group 1 and the traffic of group 2 in the ratio of 1:1.

SP+WRR queuing

You can configure some queues on an interface to use SP queuing and others to use WRR queuing by assigning the queues to the SP group and WRR groups (group 1 and group 2). With this SP+WRR queuing method, the system first schedules the queues in the SP group and then schedules queues in the WRR groups when all queues in the SP group are empty. The queues in the SP group are scheduled based on their priorities. The queues in a WRR group are scheduled based on their weights, and the two WRR groups are scheduled in the ratio of 1:1.

SP+WFQ queuing

You can configure some queues on an interface to use SP queuing and others to use WFQ queuing by assigning the queues to the SP group and WFQ groups (group 1 and group 2). With this SP+WFQ queuing method, the system schedules traffic as follows:

1. The system schedules the traffic conforming to the minimum guaranteed bandwidth in each WFQ group and schedules the traffic of the two WFQ groups in the ratio of 1:1 in a round robin manner.
2. The system uses SP to schedule queues in the SP group.
3. If there is remaining bandwidth, the system schedules the traffic of queues in each WFQ group based on their weights and schedules the traffic of the two WFQ groups in the ratio of 1:1 ratio in a round robin manner.

Congestion management configuration task list

Tasks at a glance	Remarks
(Required.) Configuring queuing • Configuring SP queuing • Configuring WRR queuing • Configuring WFQ queuing • Configuring SP+WRR queuing • Configuring SP+WFQ queuing	Perform one of the tasks.
(Optional.) Configuring queue scheduling profiles	N/A
(Optional.) Setting the queue aging time	N/A

Configuring queuing

Configuring SP queuing

Configuration procedure

To configure SP queuing:

Step		Command	Remarks
1.	Enter system view.	**system-view**	N/A
2.	Enter Ethernet interface view.	**interface** *interface-type interface-number*	N/A
3.	Configure SP queuing.	**qos sp**	The default queuing algorithm on an interface is WRR queuing.

Configuration example

Configure Ten-GigabitEthernet 1/0/1 to use SP queuing:

# Enter system view

```
<Sysname> system-view
```

# Configure Ten-GigabitEthernet1/0/1 to use SP queuing.

```
[Sysname] interface ten-gigabitethernet 1/0/1
[Sysname-Ten-GigabitEthernet1/0/1] qos sp
```

Configuring WRR queuing

Configuration procedure

To configure WRR queuing:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Enter Ethernet interface view.	**interface** *interface-type interface-number*	N/A
3. Enable byte-count or packet-based WRR queuing.	**qos wrr** { **byte-count** | **weight** }	By default, byte-count WRR queuing is used.
4. Assign a queue to a WRR group, and configure scheduling parameters for the queue.	**qos wrr** *queue-id* **group** { **1** | **2** } { **byte-count** | **weight** } *schedule-value*	Select **weight** or **byte-count** according to the type (byte-count or packet-based) of WRR you have enabled. By default, all queues are in group 1, and the weights of queues 0 through 7 are 1, 2, 3, 4, 5, 9, 13, and 15, respectively.

Configuration example

1. Network requirements

   Enable packet-based WRR on port Ten-GigabitEthernet 1/0/1, assign queues 0 through 3 to WRR group 1, with their weights being 1, 2, 4, 6, respectively, and assign queues 4 through 7 to WRR group 2, with their weights being 1, 2, 4, 6, respectively.

2. Configuration procedure

   # Enter system view.

```
<Sysname> system-view
```

   # Configure WRR queuing on Ten-GigabitEthernet 1/0/1.

```
[Sysname] interface Ten-GigabitEthernet 1/0/1
[Sysname-Ten-GigabitEthernet1/0/1] qos wrr weight
[Sysname-Ten-GigabitEthernet1/0/1] qos wrr 0 group 1 weight 1
[Sysname-Ten-GigabitEthernet1/0/1] qos wrr 1 group 1 weight 2
[Sysname-Ten-GigabitEthernet1/0/1] qos wrr 2 group 1 weight 4
[Sysname-Ten-GigabitEthernet1/0/1] qos wrr 3 group 1 weight 6
[Sysname-Ten-GigabitEthernet1/0/1] qos wrr 4 group 2 weight 1
[Sysname-Ten-GigabitEthernet1/0/1] qos wrr 5 group 2 weight 2
[Sysname-Ten-GigabitEthernet1/0/1] qos wrr 6 group 2 weight 4
[Sysname-Ten-GigabitEthernet1/0/1] qos wrr 7 group 2 weight 6
```

Configuring WFQ queuing

Configuration procedure

To configure WFQ queuing:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Enter Ethernet interface view.	**interface** *interface-type interface-number*	N/A
3. Enable byte-count or packet-based WFQ queuing.	**qos wfq** { **byte-count** | **weight** }	The default queuing algorithm on an interface is WRR queuing.

4. Assign a queue to a WFQ group, and configure scheduling parameters for the queue.	**qos wfq** *queue-id* **group** { **1** | **2** } { **byte-count** | **weight** } *schedule-value*	Select **weight** or **byte-count** according to the type (byte-count or packet-based) of WFQ you have enabled. By default, all queues are in WFQ group 1 and have a weight of 1.
5. (Optional.) Set the minimum guaranteed bandwidth for a WFQ queue.	**qos bandwidth queue** *queue-id* **min** *bandwidth-value*	The default setting is 64 kbps for each queue.

Configuration example

1. Network requirements
   - Configure byte-count WFQ queuing on interface Ten-GigabitEthernet 1/0/1.
   - Assign queues 1, 3, 4, 5, and 6 to WFQ group 1, with their weights being 2, 5, 10, 10, and 10, respectively, and assign queues 0, 2, and 7 to WFQ group 2, with their weights being 1, 2, and 4, respectively.
   - Configure the minimum guaranteed bandwidth as 100 Mbps for each queue.
2. Configuration procedure

   # Enter system view.

   ```
   <Sysname> system-view
   ```

   # Configure byte-count WFQ queuing on interface Ten-GigabitEthernet 1/0/1.

   ```
   [Sysname] interface ten-gigabitethernet 1/0/1
   [Sysname-Ten-GigabitEthernet1/0/1] qos wfq byte-count
   [Sysname-Ten-GigabitEthernet1/0/1] qos wfq 1 group 1 byte-count 2
   [Sysname-Ten-GigabitEthernet1/0/1] qos wfq 3 group 1 byte-count 5
   [Sysname-Ten-GigabitEthernet1/0/1] qos wfq 4 group 1 byte-count 10
   [Sysname-Ten-GigabitEthernet1/0/1] qos wfq 5 group 1 byte-count 10
   [Sysname-Ten-GigabitEthernet1/0/1] qos wfq 6 group 1 byte-count 10
   [Sysname-Ten-GigabitEthernet1/0/1] qos wfq 0 group 2 byte-count 1
   [Sysname-Ten-GigabitEthernet1/0/1] qos wfq 2 group 2 byte-count 2
   [Sysname-Ten-GigabitEthernet1/0/1] qos wfq 7 group 2 byte-count 4
   [Sysname-Ten-GigabitEthernet1/0/1] qos bandwidth queue 0 min 100000
   [Sysname-Ten-GigabitEthernet1/0/1] qos bandwidth queue 1 min 100000
   [Sysname-Ten-GigabitEthernet1/0/1] qos bandwidth queue 2 min 100000
   [Sysname-Ten-GigabitEthernet1/0/1] qos bandwidth queue 3 min 100000
   [Sysname-Ten-GigabitEthernet1/0/1] qos bandwidth queue 4 min 100000
   [Sysname-Ten-GigabitEthernet1/0/1] qos bandwidth queue 5 min 100000
   [Sysname-Ten-GigabitEthernet1/0/1] qos bandwidth queue 6 min 100000
   [Sysname-Ten-GigabitEthernet1/0/1] qos bandwidth queue 7 min 100000
   ```

Configuring SP+WRR queuing

Configuration procedure

To configure SP+WRR queuing:

Step		Command	Remarks
1.	Enter system view.	**system-view**	N/A
2.	Enter Ethernet interface view.	**interface** *interface-type interface-number*	N/A
3.	Enable byte-count or packet-based WRR queuing.	**qos wrr** { **byte-count** | **weight** }	By default, all ports use WRR queuing.
4.	Assign a queue to the SP group.	**qos wrr** *queue-id* **group sp**	By default, all the queues of a WRR-enabled port are in WRR group 1.
5.	Assign a queue to a WRR group, and configure the scheduling weight for the queue.	**qos wrr** *queue-id* **group** { **1** | **2** } { **weight** | **byte-count** } *schedule-value*	Select **weight** or **byte-count** according to the type (byte-count or packet-based) of WRR you have enabled. By default, all queues are in WRR group 1, and the weights of queues 0 through 7 are 1, 2, 3, 4, 5, 9, 13, and 15, respectively.

Configuration example

1. Network requirements
   - Configure SP+WRR queuing on Ten-GigabitEthernet 1/0/1, and use byte-count WRR.
   - Assign queues 4 through 7 on Ten-GigabitEthernet 1/0/1 to the SP group.
   - Assign queues 0 and 1 on Ten-GigabitEthernet 1/0/1 to WRR group 1, with the weights being 1 and 2, respectively. Assign queues 2 and 3 to WRR group 2, with the weights being 1 and 3, respectively.
2. Configuration procedure

   # Enter system view.

   ```
   <Sysname> system-view
   ```

   # Configure SP+WRR queuing on Ten-GigabitEthernet1/0/1.

   ```
   [Sysname] interface Ten-GigabitEthernet 1/0/1
   [Sysname-Ten-GigabitEthernet1/0/1] qos wrr byte-count
   [Sysname-Ten-GigabitEthernet1/0/1] qos wrr 4 group sp
   [Sysname-Ten-GigabitEthernet1/0/1] qos wrr 5 group sp
   [Sysname-Ten-GigabitEthernet1/0/1] qos wrr 6 group sp
   [Sysname-Ten-GigabitEthernet1/0/1] qos wrr 7 group sp
   [Sysname-Ten-GigabitEthernet1/0/1] qos wrr 0 group 1 byte-count 1
   [Sysname-Ten-GigabitEthernet1/0/1] qos wrr 1 group 1 byte-count 2
   [Sysname-Ten-GigabitEthernet1/0/1] qos wrr 2 group 2 byte-count 1
   [Sysname-Ten-GigabitEthernet1/0/1] qos wrr 3 group 2 byte-count 3
   ```

Configuring SP+WFQ queuing

Configuration procedure

To configure SP+WFQ queuing:

Step		Command	Remarks
1.	Enter system view.	**system-view**	N/A

Step	Command	Remarks
2. Enter Ethernet interface view.	**interface** *interface-type interface-number*	N/A
3. Enable byte-count or packet-based WFQ queuing.	**qos wfq** [**byte-count** | **weight**]	The default queuing algorithm on an interface is WRR.
4. Assign a queue to the SP group.	**qos wfq** *queue-id* **group sp**	By default, all the queues of a WFQ-enabled port are in WFQ group 1.
5. Assign a queue to the WFQ queue scheduling group, and configure a scheduling weight for the queue.	**qos wfq** *queue-id* **group** { **1** | **2** } { **weight** | **byte-count** } *schedule-value*	Select **weight** or **byte-count** according to the type (byte-count or packet-based) of WFQ you have enabled. If you have enabled WFQ on the port, all the queues are in WFQ group 1 by default and the default scheduling weight is 1 for each queue.
6. (Optional.) Configure the minimum guaranteed bandwidth for a queue.	**qos bandwidth queue** *queue-id* **min** *bandwidth-value*	The default setting is 64 kbps for each queue in a WFQ group.

Configuration example

1. Network requirements
   - Configure SP+WFQ queuing on interface Ten-GigabitEthernet 1/0/1, and use packet-based WFQ.
   - Assign queues 4 through 7 to the SP group.
   - Assign queues 0 and 1 to WFQ group 1, with the weights being 1 and 2, respectively. Assign queues 2 and 3 to WFQ group 2, with the weights being 1 and 3, respectively.
   - Configure the minimum guaranteed bandwidth for each of the four queues as 128 Mbps.
2. Configuration procedure

   # Enter system view.

```
<Sysname> system-view
```

   # Configure SP+WFQ queuing on Ten-GigabitEthernet 1/0/1.

```
[Sysname] interface ten-gigabitEthernet 1/0/1
[Sysname-Ten-GigabitEthernet1/0/1] qos wfq weight
[Sysname-Ten-GigabitEthernet1/0/1] qos wfq 4 group sp
[Sysname-Ten-GigabitEthernet1/0/1] qos wfq 5 group sp
[Sysname-Ten-GigabitEthernet1/0/1] qos wfq 6 group sp
[Sysname-Ten-GigabitEthernet1/0/1] qos wfq 7 group sp
[Sysname-Ten-GigabitEthernet1/0/1] qos wfq 0 group 1 weight 1
[Sysname-Ten-GigabitEthernet1/0/1] qos bandwidth queue 4 min 128000
[Sysname-Ten-GigabitEthernet1/0/1] qos wfq 1 group 1 weight 2
[Sysname-Ten-GigabitEthernet1/0/1] qos bandwidth queue 5 min 128000
[Sysname-Ten-GigabitEthernet1/0/1] qos wfq 2 group 2 weight 1
[Sysname-Ten-GigabitEthernet1/0/1] qos bandwidth queue 6 min 128000
[Sysname-Ten-GigabitEthernet1/0/1] qos wfq 3 group 2 weight 3
[Sysname-Ten-GigabitEthernet1/0/1] qos bandwidth queue 7 min 128000
```

Displaying and maintaining queuing

Execute **display** commands in any view.

Task	Command
Display SP queuing configuration.	**display qos queue sp interface** [*interface-type interface-number*]
Display WRR queuing configuration.	**display qos queue wrr interface** [*interface-type interface-number*]
Display WFQ queuing configuration.	**display qos queue wfq interface** [*interface-type interface-number*]
Display queue-based outbound traffic statistics.	**display qos queue-statistics interface** [*interface-type interface-number*] **outbound**

Configuring queue scheduling profiles

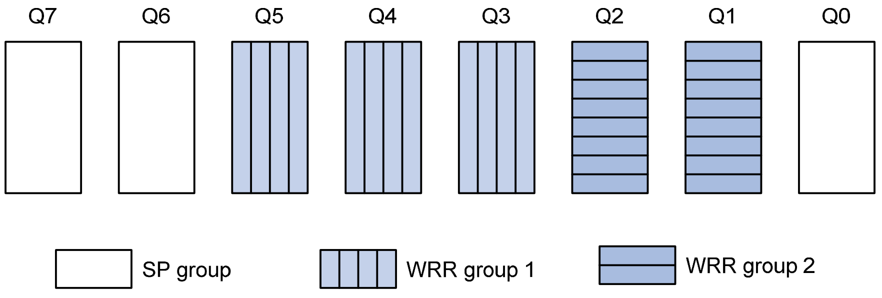
In a queue scheduling profile, you can configure scheduling parameters for each queue. By applying the queue scheduling profile to an interface, you can implement congestion management on the interface.

Queue scheduling profiles support three queue scheduling methods: SP, WRR, and WFQ. In a queue scheduling profile, you can configure SP+WRR or SP+WFQ. When SP+WRR or SP+WFQ is configured, the scheduling priority is as follows:

- The SP group has higher priority than WRR groups and WFQ groups.
- Queues in the SP group are scheduled in descending order of queue IDs.
- WRR or WFQ groups are scheduled in the 1:1 ratio.
- In a WRR or WFQ group, queues are scheduled based on their weights.

When SP and WRR groups are configured in a queue scheduling profile, Figure 16 shows the scheduling order.

Figure 16 Queue scheduling profile configured with both SP and WRR



- Queue 7 has the highest priority. Its packets are sent preferentially.
- Queue 6 has the second highest priority. Packets in queue 6 are sent when queue 7 is empty.
- Queue 0 has the third highest priority, and it is scheduled when queue 7 and queue 6 are empty.
- Queues 3 through 5 in WRR group 1 are scheduled according to their weights when queue 7, queue 6, and queue 0 are empty.

- Queues 1 and 2 in WRR group 2 are scheduled according to their weights when all other queues are empty.

Configuring a queue scheduling profile

You can modify the scheduling parameters in a queue scheduling profile already applied to an interface. The modification takes effect immediately.

To configure a queue scheduling profile:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Create a queue scheduling profile and enter queue scheduling profile view.	**qos qmprofile** *profile-name*	By default, no queue scheduling profile exists.
3. Configure queue scheduling parameters.	• Configure a queue to use SP: **queue** *queue-id* **sp** • Configure a queue to use WRR: **queue** *queue-id* **wrr group** *group-id* { **byte-count** | **weight** } *schedule-value* • Configure a queue to use WFQ: **queue** *queue-id* **wfq group** *group-id* { **byte-count** | **weight** } *schedule-value* **bandwidth queue** *queue-id* **min** *bandwidth-value*	By default, all queues use SP. You can configure all queues to use one queuing method or different queuing methods (WRR+WFQ is not allowed).
4. Return to system view.	**quit**	N/A
5. Enter Ethernet interface view.	**interface** *interface-type interface-number*	N/A
6. Apply the queue scheduling profile to the interface.	**qos apply qmprofile** *profile-name*	Only one queue scheduling profile can be applied an interface.

Displaying and maintaining queue scheduling profiles

Execute **display** commands in any view.

Task	Command
Display the configuration of queue scheduling profiles.	**display qos qmprofile configuration** [*profile-name*] [**slot** *slot-number*]
Display the queue scheduling profiles applied to interfaces.	**display qos qmprofile interface** [*interface-type interface-number*]

Queue scheduling profile configuration example

Network requirements

Configure a queue scheduling profile on interface Ten-GigabitEthernet 1/0/1 to meet the following requirements:

- Queue 7 has the highest priority, and its packets are sent preferentially.
- Queue 4, queue 5, and queue 6 in WRR group 1 are scheduled according to their weights, which are 1, 5, and 10, respectively. When queue 7 is empty, WRR group 1 is scheduled.
- Queues 0 through 3 in WRR group 2 are scheduled according to their weights, which are 1, 1, 10, and 15, respectively. When queues 4 through 7 are all empty, WRR group 2 is scheduled.

Configuration procedure

# Enter system view.

```
<Sysname> system-view
```

# Create a queue scheduling profile named **qm1**.

```
[Sysname] qos qmprofile qm1
[Sysname-qmprofile-qm1]
```

# Configure queue 7 to use SP queuing.

```
[Sysname-qmprofile-qm1] queue 7 sp
```

# Assign queue 4, queue 5, and queue 6 to WRR group 1, with the weights of 1, 5, and 10, respectively.

```
[Sysname-qmprofile-qm1] queue 4 wrr group 1 weight 1
[Sysname-qmprofile-qm1] queue 5 wrr group 1 weight 5
[Sysname-qmprofile-qm1] queue 6 wrr group 1 weight 10
```

# Assign queues 0 through 3 to WRR group 2, with their weights as 1, 1, 10, and 15, respectively.

```
[Sysname-qmprofile-qm1] queue 0 wrr group 2 weight 1
[Sysname-qmprofile-qm1] queue 1 wrr group 2 weight 1
[Sysname-qmprofile-qm1] queue 2 wrr group 2 weight 10
[Sysname-qmprofile-qm1] queue 3 wrr group 2 weight 15
[Sysname-qmprofile-qm1] quit
```

# Apply the queue scheduling profile **qm1** to interface Ten-GigabitEthernet 1/0/1.

```
[Sysname] interface Ten-GigabitEthernet 1/0/1
[Sysname-Ten-GigabitEthernet1/0/1] qos apply qmprofile qm1
```

Setting the queue aging time

When the queue aging time expires, packets already in queues are dropped.

To set the queue aging time:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Set the queue aging time.	**qos queue aging-time** *time-value*	By default, the queue aging time is 0 milliseconds (the aging feature is disabled).

Configuring congestion avoidance

Overview

Avoiding congestion before it occurs is a proactive approach to improving network performance. As a flow control mechanism, congestion avoidance:

- Actively monitors network resources (such as queues and memory buffers).
- Drops packets when congestion is expected to occur or deteriorate.

When dropping packets from a source end, congestion avoidance cooperates with the flow control mechanism at the source end to regulate the network traffic size. The combination of the local packet drop policy and the source-end flow control mechanism helps maximize throughput and network use efficiency and minimize packet loss and delay.

Tail drop

Congestion management techniques drop all packets that are arriving at a full queue. This tail drop mechanism results in global TCP synchronization. If packets from multiple TCP connections are dropped, these TCP connections go into the state of congestion avoidance and slow start to reduce traffic, but traffic peak occurs later. Consequently, the network traffic jitters all the time.

RED and WRED

You can use Random Early Detection (RED) or Weighted Random Early Detection (WRED) to avoid global TCP synchronization.

Both RED and WRED avoid global TCP synchronization by randomly dropping packets. When the sending rates of some TCP sessions slow down after their packets are dropped, other TCP sessions remain at high sending rates. Link bandwidth is efficiently used, because TCP sessions at high sending rates always exist.

The RED or WRED algorithm sets an upper limit and lower limit for each queue, and processes the packets in a queue as follows:

- When the queue size is shorter than the lower limit, no packet is dropped.
- When the queue size reaches the upper limit, all subsequent packets are dropped.
- When the queue size is between the lower limit and the upper limit, the received packets are dropped based on the user-configured drop probability.

If the current queue size is compared with the upper limit and lower limit to determine the drop policy, burst traffic is not fairly treated. To solve this problem, WRED compares the average queue size with the upper limit and lower limit to determine the drop probability.

The average queue size reflects the queue size change trend but is not sensitive to burst queue size changes, and burst traffic can be fairly treated.

ECN

By dropping packets, WRED alleviates the influence of congestion on the network. However, the network resources for transmitting packets from the sender to the device which drops the packets are wasted. When congestion occurs, it is a better idea to inform the sender of the congestion status and have the sender proactively slow down the packet sending rate or decrease the window size of packets. This better utilizes the network resources.

RFC 2482 defined an end-to-end congestion notification mechanism named Explicit Congestion Notification (ECN). ECN uses the DS field in the IP header to mark the congestion status along the packet transmission path. A ECN-capable terminal can determine whether congestion occurs on the transmission path according to the packet contents, and then adjusts the packet sending speed to avoid deteriorating congestion. ECN defines the last two bits (ECN field) in the DS field of the IP header as follows:

- Bit 6 indicates whether the sending terminal device supports ECN, and is called the "ECN-Capable Transport (ECT)" bit.
- Bit 7 indicates whether the packet has experienced congestion along the transmission path, and is called the "Congestion Experienced (CE)" bit.

For more information about the DS field, see "Appendixes."

In actual applications, the packets with ECT set to 1 and CE set to 0 and the packets with ECT set to 0 and CE set to 1 are considered as packets that an ECN-capable endpoint transmits.

After you enable ECN on a device, congestion management processes packets as follows:

- When the average queue size is below the lower limit, no packet is dropped, and the ECN fields of packets are not identified or marked.
- When the average queue size exceeds the lower limit and is below the upper limit, before the device drops a packet which should be dropped according to the drop probability, the device examines the ECN field of the packet.
    - If the ECN field shows that the packet is sent out of ECN-capable terminal, the device sets both the ECT bit and the CE bit to 1 and forwards the packet.
    - If the ECN field shows that the packet has experienced congestion along the transmission path (both the ECT bit and the CE bit are 1), the device forwards the packet without modifying the ECN field.
    - If both the ECT bit and the CE bit are 0s, the device drops the packet.
- When the average queue size exceeds the upper limit, the device drops the packet, regardless of whether the packet is sent out from an ECN-capable terminal.

ECN is enabled on a per-queue basis. You can configure the switch to identify and mark the ECN fields of packets for a specific queue.

Configuring and applying a WRED table

The switch supports the queue-based WRED table. You can configure separate drop parameters for different queues. When congestion occurs, packets of a queue are randomly dropped based on drop parameters of the queue.

Determine the following parameters before configuring WRED:

- **Upper limit and lower limit**—When the average queue size is smaller than the lower limit, packets are not dropped. When the average queue size is between the lower limit and the upper limit, the packets are dropped based on the user-configured drop probability. When the average queue size exceeds the upper limit, subsequent packets are dropped.
- **Drop precedence**—A parameter used for packet drop. The value 0 corresponds to green packets, the value 1 corresponds to yellow packets, and the value 2 corresponds to red packets. Red packets are dropped preferentially.
- **Exponent for average queue size calculation**—The greater the exponent, the less sensitive the average queue size is to real-time queue size changes. The formula for calculating the average queue size is average queue size = (previous average queue size x (1 – 2^{-n})) + (current queue size x 2^{-n}), where n is the exponent.
- **Drop probability in percentage**—The larger the value is, the greater the drop probability is.

Configuration procedure

A WRED table can be applied to multiple interfaces. For a WRED table already applied to an interface, you can modify the values of the WRED table, but you cannot remove the WRED table.

To configure and apply a WRED table:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Create a WRED table and enter its view.	**qos wred queue table** *table-name*	N/A
3. (Optional.) Set the WRED exponent for average queue size calculation.	**queue** *queue-id* **weighting-constant** *exponent*	The default setting is 9.
4. (Optional.) Configure drop parameters.	**queue** *queue-id* [**drop-level** *drop-level*] **low-limit** *low-limit* **high-limit** *high-limit* [**discard-probability** *discard-prob*]	By default, the low limit is 100, the high limit is 1000, and the drop probability is 10%.
5. (Optional.) Enable ECN for a queue.	**queue** *queue-id* **ecn**	By default, ECN is not enabled on any queue.
6. Enter Ethernet interface view.	**interface** *interface-type interface-number*	N/A
7. Apply the WRED table to the interface.	**qos wred apply** [*table-name*]	By default, no WRED table is applied to an interface, and tail drop is used on an interface.

Configuration example

Network requirements

Apply a WRED table to interface Ten-GigabitEthernet 1/0/2, so that the packets are dropped as follows when congestion occurs:

- For the interface to preferentially forward higher-priority traffic, set a lower drop probability for a queue with a greater queue ID. Set different drop parameters for queue 0, queue 3, and queue 7.
- Drop packets according to their colors.

- In queue 0, set the drop probability to 25%, 50%, and 75% for green, yellow, and red packets, respectively.
- In queue 3, set the drop probability to 5%, 10%, and 25% for green, yellow, and red packets, respectively.
- In queue 7, set the drop probability to 1%, 5%, and 10% for green, yellow, and red packets, respectively.

- Enable ECN for queue 7.

Configuration procedure

# Configure a queue-based WRED table, and set different drop parameters for packets with different drop levels in different queues.

```
<Sysname> system-view
[Sysname] qos wred queue table queue-table1
[Sysname-wred-table-queue-table1] queue 0 drop-level 0 low-limit 128 high-limit 512
discard-probability 25
[Sysname-wred-table-queue-table1] queue 0 drop-level 1 low-limit 128 high-limit 512
discard-probability 50
[Sysname-wred-table-queue-table1] queue 0 drop-level 2 low-limit 128 high-limit 512
discard-probability 75
[Sysname-wred-table-queue-table1] queue 3 drop-level 0 low-limit 256 high-limit 640
discard-probability 5
[Sysname-wred-table-queue-table1] queue 3 drop-level 1 low-limit 256 high-limit 640
discard-probability 10
[Sysname-wred-table-queue-table1] queue 3 drop-level 2 low-limit 256 high-limit 640
discard-probability 25
[Sysname-wred-table-queue-table1] queue 7 drop-level 0 low-limit 512 high-limit 1024
discard-probability 1
[Sysname-wred-table-queue-table1] queue 7 drop-level 1 low-limit 512 high-limit 1024
discard-probability 5
[Sysname-wred-table-queue-table1] queue 7 drop-level 2 low-limit 512 high-limit 1024
discard-probability 10
[Sysname-wred-table-queue-table1] queue 7 ecn
[Sysname-wred-table-queue-table1] quit
```

# Apply the queue-based WRED table to interface Ten-GigabitEthernet 1/0/2.

```
[Sysname] interface Ten-GigabitEthernet 1/0/2
[Sysname-Ten-GigabitEthernet1/0/2] qos wred apply queue-table1
```

Displaying and maintaining WRED

Execute **display** commands in any view.

Task	Command
Display WRED configuration and statistics for interfaces.	**display qos wred interface** [*interface-type interface-number*]
Display the configuration of WRED tables.	**display qos wred table** [**name** *table-name*] [**slot** *slot-number*]

Configuring traffic filtering

You can filter in or filter out traffic of a class by associating the class with a traffic filtering action. For example, you can filter packets sourced from an IP address according to network status.

Configuration procedure

To configure traffic filtering:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Create a traffic class and enter traffic class view.	**traffic classifier** *classifier-name* [**operator** { **and** | **or** }]	By default, no traffic class is configured.
3. Configure match criteria.	**if-match** *match-criteria*	By default, no match criterion is configured.
4. Return to system view.	**quit**	N/A
5. Create a traffic behavior and enter traffic behavior view.	**traffic behavior** *behavior-name*	By default, no traffic behavior is configured.
6. Configure the traffic filtering action.	**filter** { **deny** | **permit** }	By default, no traffic filtering action is configured. If a traffic behavior has the **filter deny** action, all the other actions except for class-based accounting in the traffic behavior do not take effect.
7. Return to system view.	**quit**	N/A
8. Create a QoS policy and enter QoS policy view.	**qos policy** *policy-name*	By default, no QoS policy is configured.
9. Associate the traffic class with the traffic behavior in the QoS policy.	**classifier** *classifier-name* **behavior** *behavior-name* [**insert-before** *before-classifier-name*]	By default, a traffic class is not associated with a traffic behavior.
10. Return to system view.	**quit**	N/A
11. Apply the QoS policy.	• Applying the QoS policy to an interface • Applying the QoS policy to a VLAN • Applying the QoS policy globally • Applying the QoS policy to a control plane	Choose one of the application destinations as needed. By default, no QoS policy is applied.

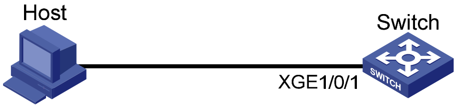
Step	Command	Remarks
12. (Optional.) Display the traffic filtering configuration.	**display traffic behavior user-defined** [*behavior-name*]	Available in any view.

Configuration example

Network requirements

As shown in Figure 17, configure traffic filtering on Ten-GigabitEthernet 1/0/1 to deny the incoming packets with port 21 as the source port.

Figure 17 Network diagram



Configuration procedure

# Create advanced ACL 3000, and configure a rule to match packets whose source port number is 21.

```
<Switch> system-view
[Switch] acl number 3000
[Switch-acl-adv-3000] rule 0 permit tcp source-port eq 21
[Switch-acl-adv-3000] quit
```

# Create a traffic class named **classifier_1**, and use ACL 3000 as the match criterion in the traffic class.

```
[Switch] traffic classifier classifier_1
[Switch-classifier-classifier_1] if-match acl 3000
[Switch-classifier-classifier_1] quit
```

# Create a traffic behavior named **behavior_1**, and configure the traffic filtering action to drop packets.

```
[Switch] traffic behavior behavior_1
[Switch-behavior-behavior_1] filter deny
[Switch-behavior-behavior_1] quit
```

# Create a QoS policy named **policy**, and associate traffic class **classifier_1** with traffic behavior **behavior_1** in the QoS policy.

```
[Switch] qos policy policy
[Switch-qospolicy-policy] classifier classifier_1 behavior behavior_1
[Switch-qospolicy-policy] quit
```

# Apply the QoS policy named **policy** to the incoming traffic of Ten-GigabitEthernet 1/0/1.

```
[Switch] interface ten-gigabitethernet 1/0/1
[Switch-Ten-GigabitEthernet1/0/1] qos apply policy policy inbound
```

Configuring priority marking

Overview

Priority marking sets the priority fields or flag bits of packets to modify the priority of packets. For example, you can use priority marking to set IP precedence or DSCP for a traffic class of IP packets to control the forwarding of these packets.

To configure priority marking to set the priority fields or flag bits for a class of packets, perform the following tasks:

1. Configure a traffic behavior with a priority marking action.
2. Associate the traffic class with the traffic behavior.

Priority marking can be used together with priority mapping. For more information, see "Configuring priority mapping."

Color-based priority marking

Packet coloring methods

The color of a packet indicates the device's evaluation for the packet transmission priority. The device can color a packet by using either of the following methods:

- Traffic policing
- Mapping drop precedence

Traffic policing

Traffic policing is a common traffic control technology. Traffic policing uses the token bucket mechanism to evaluate the incoming or outgoing packets and colors the packets according to the evaluation result. By configuring different traffic control polices for packets in different colors, you can provide differentiated services for different traffic flows and ensure that the network resources are well utilized.

The device supports evaluating traffic by using two token buckets (bucket C and bucket E), and it colors a packet according to the number of tokens in the token buckets.

The device supports coloring packets by using either of the following traffic policing functions: common CAR and aggregate CAR. For more information about coloring packets by using token buckets and about common CAR and aggregate CAR, see "Configuring traffic policing, GTS, and rate limit" and "Configuring aggregate CAR."

Mapping drop precedence

Without traffic policing configured, a switch looks up the 802.1p priority of a packet in the 802.1p-to-drop priority mapping table, allocates the drop precedence value to the packet, and colors the packet according to the drop precedence value. Drop precedence value 0 denotes green packets, 1 denotes yellow packets, and 2 denotes red packets. For more information about priority mapping tables, see "Configuring priority mapping."

Configuring color-based priority marking

Configuring priority marking based on colors obtained through traffic policing

After traffic policing evaluates and colors packets, the switch can mark traffic with various priority values (including DSCP values, 802.1p priority values, and local precedence values) by color. Configure priority marking by using either of the following methods:

- Configuring the priority marking actions by color in the traffic policing action.
- Configuring the priority marking actions by color in the behavior where the traffic policing action is configured.

You can use both methods to mark multiple priority values for packets in the same color. However, do not use the two methods to mark different values of the same priority type for packets. Otherwise, the QoS policy configured with the behavior cannot be applied normally.

In a traffic behavior, an aggregate CAR action cannot be configured together with a priority marking action. Otherwise, the QoS policy configured with the behavior cannot be applied normally.

The switch implements both common CAR and aggregate CAR by using a QoS policy. For more information about configuring classes and behaviors in a QoS policy, see "Configuring traffic policing, GTS, and rate limit" and "Configuring aggregate CAR."

Configuring priority marking based on colors obtained through mapping drop precedence

When packets are colored based on drop precedence values, you can create priority marking actions for packets in different colors in a traffic behavior and mark DSCP values, 802.1p priority values, and local precedence values for packets.

Configuration procedure

To configure priority marking:

Step		Command	Remarks
1.	Enter system view.	**system-view**	N/A
2.	Create a traffic class and enter traffic class view.	**traffic classifier** *classifier-name* [**operator** { **and** | **or** }]	By default, no traffic class is configured.
3.	Configure match criteria.	**if-match** *match-criteria*	By default, no match criterion is configured. For more information about the **if-match** command, see *ACL and QoS Command Reference*.
4.	Return to system view.	**quit**	N/A
5.	Create a traffic behavior and enter traffic behavior view.	**traffic behavior** *behavior-name*	By default, no traffic behavior is configured.

Step	Command	Remarks
6. Configure a priority marking action.	- Set the DSCP value for packets: **remark** [**green** | **red** | **yellow**] **dscp** *dscp-value* - Set the 802.1p priority for packets or configure the inner-to-outer tag priority copying function: **remark** [**green** | **red** | **yellow**] **dot1p** *dot1p-value* **remark dot1p customer-dot1p-trust** - Set the drop priority for packets: **remark drop-precedence** *drop-precedence-value* - Set the IP precedence for packets: **remark ip-precedence** *ip-precedence-value* - Set the local precedence for packets: **remark**[**green** | **red** | **yellow**] **local-precedence** *local-precedence-value* - Set the local QoS ID for packets: **remark qos-local-id** *local-id-value* - Set the CVLAN for packets: **remark customer-vlan-id** *vlan-id* - Set the SVLAN for packets: **remark service-vlan-id** *vlan-id*	By default, no priority marking action is configured. The switch supports local QoS IDs in the range of 1 to 3999. The **remark local-precedence**, **remark qos-local-id**, and **remark drop-precedence** commands apply only to the incoming traffic. The customer VLAN (CVLAN) is the private network VLAN of the customer, and the service provider VLAN (SVLAN) is the public network VLAN assigned by the service provider to the customer. For more information about the CVLAN and SVLAN, see *Layer 2—LAN Switching Configuration Guide*.
7. Return to system view.	**quit**	N/A
8. Create a QoS policy and enter QoS policy view.	**qos policy** *policy-name*	By default, no QoS policy is configured.
9. Associate the traffic class with the traffic behavior in the QoS policy.	**classifier** *classifier-name* **behavior** *behavior-name* [**insert-before** *before-classifier-name*]	By default, a traffic class is not associated with a traffic behavior.
10. Return to system view.	**quit**	N/A
11. Apply the QoS policy.	- Applying the QoS policy to an interface - Applying the QoS policy to a VLAN - Applying the QoS policy globally - Applying the QoS policy to a control plane	Choose one of the application destinations as needed. By default, no QoS policy is applied.
12. (Optional.) Display the priority marking configuration.	**display traffic behavior user-defined** [*behavior-name*]	Available in any view.

Priority marking configuration examples

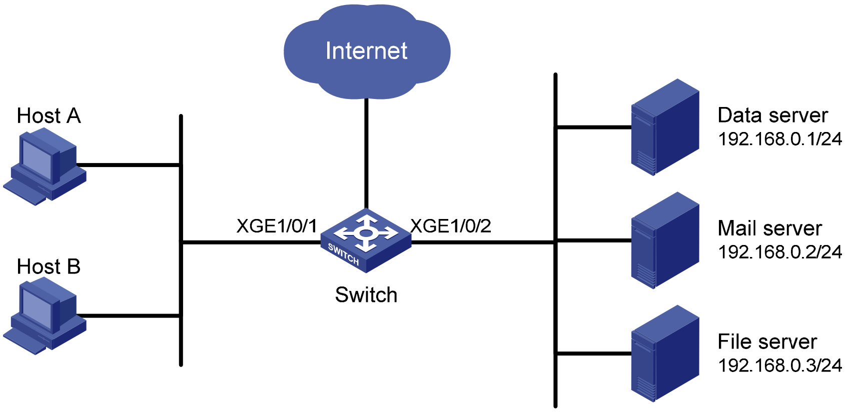
Local precedence marking configuration example

Network requirements

As shown in Figure 18, configure priority marking on the switch to meet the following requirements:

Traffic source	Destination	Processing priority
Host A, B	Data server	High
Host A, B	Mail server	Medium
Host A, B	File server	Low

Figure 18 Network diagram



Configuration procedure

# Create advanced ACL 3000, and configure a rule to match packets with destination IP address 192.168.0.1.

```
<Switch> system-view
[Switch] acl number 3000
[Switch-acl-adv-3000] rule permit ip destination 192.168.0.1 0
[Switch-acl-adv-3000] quit
```

# Create advanced ACL 3001, and configure a rule to match packets with destination IP address 192.168.0.2.

```
[Switch] acl number 3001
[Switch-acl-adv-3001] rule permit ip destination 192.168.0.2 0
[Switch-acl-adv-3001] quit
```

# Create advanced ACL 3002, and configure a rule to match packets with destination IP address 192.168.0.3.

```
[Switch] acl number 3002
[Switch-acl-adv-3002] rule permit ip destination 192.168.0.3 0
[Switch-acl-adv-3002] quit
```

# Create a traffic class named **classifier_dbserver**, and use ACL 3000 as the match criterion in the traffic class.

```
[Switch] traffic classifier classifier_dbserver
[Switch-classifier-classifier_dbserver] if-match acl 3000
[Switch-classifier-classifier_dbserver] quit
```

# Create a traffic class named **classifier_mserver**, and use ACL 3001 as the match criterion in the traffic class.

```
[Switch] traffic classifier classifier_mserver
[Switch-classifier-classifier_mserver] if-match acl 3001
```

```
[Switch-classifier-classifier_mserver] quit
```

# Create a traffic class named **classifier_fserver**, and use ACL 3002 as the match criterion in the traffic class.

```
[Switch] traffic classifier classifier_fserver
[Switch-classifier-classifier_fserver] if-match acl 3002
[Switch-classifier-classifier_fserver] quit
```

# Create a traffic behavior named **behavior_dbserver**, and configure the action of setting the local precedence value to 4.

```
[Switch] traffic behavior behavior_dbserver
[Switch-behavior-behavior_dbserver] remark local-precedence 4
[Switch-behavior-behavior_dbserver] quit
```

# Create a traffic behavior named **behavior_mserver**, and configure the action of setting the local precedence value to 3.

```
[Switch] traffic behavior behavior_mserver
[Switch-behavior-behavior_mserver] remark local-precedence 3
[Switch-behavior-behavior_mserver] quit
```

# Create a traffic behavior named **behavior_fserver**, and configure the action of setting the local precedence value to 2.

```
[Switch] traffic behavior behavior_fserver
[Switch-behavior-behavior_fserver] remark local-precedence 2
[Switch-behavior-behavior_fserver] quit
```

# Create a QoS policy named **policy_server**, and associate traffic classes with traffic behaviors in the QoS policy.

```
[Switch] qos policy policy_server
[Switch-qospolicy-policy_server] classifier classifier_dbserver behavior
behavior_dbserver
[Switch-qospolicy-policy_server] classifier classifier_mserver behavior
behavior_mserver
[Switch-qospolicy-policy_server] classifier classifier_fserver behavior
behavior_fserver
[Switch-qospolicy-policy_server] quit
```

# Apply the QoS policy named **policy_server** to the incoming traffic of Ten-GigabitEthernet 1/0/1.

```
[Switch] interface Ten-GigabitEthernet 1/0/1
[Switch-Ten-GigabitEthernet1/0/1] qos apply policy policy_server inbound
[Switch-Ten-GigabitEthernet1/0/1] quit
```

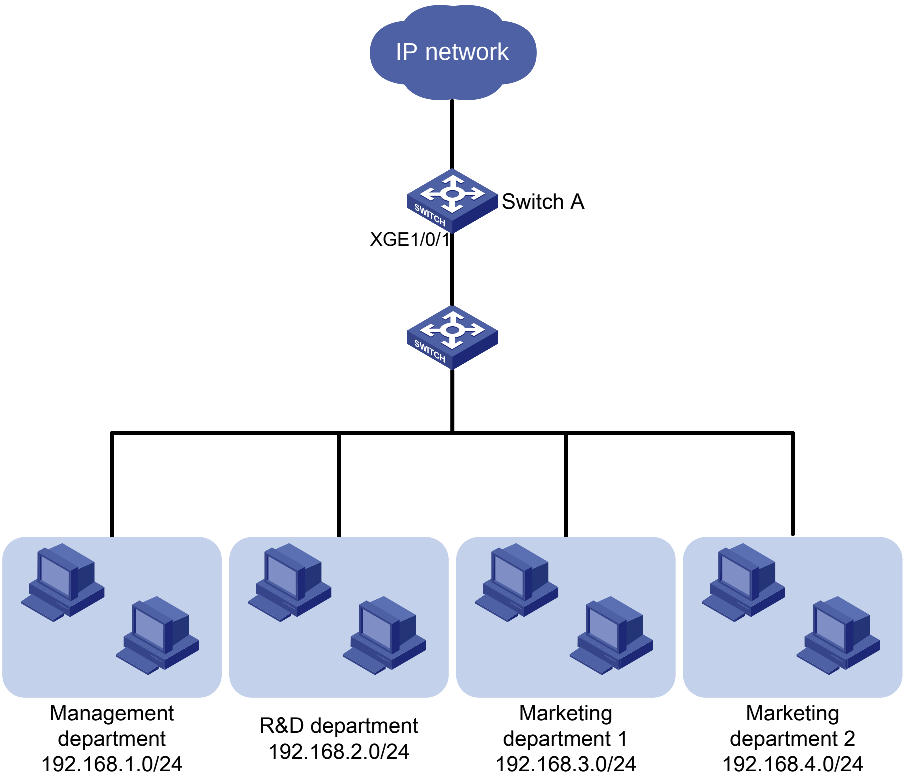
Local QoS ID marking configuration example

Local QoS ID marking allows you to mark the same local QoS ID for packets of multiple classes and configure a new class to match the local QoS ID to group these packets into the new class. With this feature, you can perform QoS actions for the old classes respectively and perform other QoS actions for the new class. In this way, you can perform layers of QoS actions for the specific packets.

Network requirements

As shown in Figure 19, configure local QoS ID marking and traffic policing to limit the outgoing traffic of the Management department and the R&D department to 102400 kbps, respectively, and limit the outgoing traffic of the Marketing department (containing two sub-departments) to 204800 kbps.

Figure 19 Network diagram


IP network
Switch A
SWITCH
XGE1/0/1
SWITCH
Management department 192.168.1.0/24
R&D department 192.168.2.0/24
Marketing department 1 192.168.3.0/24
Marketing department 2 192.168.4.0/24

Configuration considerations

- Configure two classes to match the traffic from the Management department and the R&D department, respectively, and then configure traffic policing behaviors for the two classes.
- Mark the same local QoS ID for the traffic from the two sub-departments of the Marketing department, configure a class to match packets with the local QoS ID, and then configure a traffic policing behavior for the class to limit the outgoing traffic of the two sub-departments.

Configuration procedure

1. Limit the upstream traffic of the Management department and R&D department:

# Configure IPv4 basic ACL 2001 to match the outgoing traffic of the Management department.

```
<SwitchA> system-view
[SwitchA] acl number 2001
[SwitchA-acl-basic-2001] rule permit source 192.168.1.0 0.0.0.255
[SwitchA-acl-basic-2001] quit
```

# Configure IPv4 basic ACL 2002 to match the outgoing traffic of the R&D department.

```
[SwitchA] acl number 2002
[SwitchA-acl-basic-2002] rule permit source 192.168.2.0 0.0.0.255
[SwitchA-acl-basic-2002] quit
```

# Create class **admin**, and use ACL 2001 as the match criterion.

```
[SwitchA] traffic classifier admin
[SwitchA-classifier-admin] if-match acl 2001
[SwitchA-classifier-admin] quit
```

# Create class **rd**, and use ACL 2002 as the match criterion.

```
[SwitchA] traffic classifier rd
```

```
[SwitchA-classifier-rd] if-match acl 2002
[SwitchA-classifier-rd] quit
```

# Create traffic behavior **car_admin_rd**, and configure traffic policing to limit the traffic rate to 102400 kbps.

```
[SwitchA] traffic behavior car_admin_rd
[SwitchA-behavior-car_admin_rd] car cir 102400
[SwitchA-behavior-car_admin_rd] quit
```

# Create QoS policy **car**, and associate classes **admin** and **rd** with behavior **car_admin_rd**.

```
[SwitchA] qos policy car
[SwitchA-qospolicy-car] classifier admin behavior car_admin_rd
[SwitchA-qospolicy-car] classifier rd behavior car_admin_rd
[SwitchA-qospolicy-car] quit
```

2. Limit the upstream traffic of the marketing department:

# Configure IPv4 basic ACL 2003 to match the outgoing traffic of the sub-department 1 of the marketing department.

```
[SwitchA] acl number 2003
[SwitchA-acl-basic-2003] rule permit source 192.168.3.0 0.0.0.255
[SwitchA-acl-basic-2003] quit
```

# Configure IPv4 basic ACL 2004 to match the outgoing traffic of the sub-department 2 of the Marketing department.

```
[SwitchA] acl number 2004
[SwitchA-acl-basic-2004] rule permit source 192.168.4.0 0.0.0.255
[SwitchA-acl-basic-2004] quit
```

# Configure class **marketing** to match the outgoing traffic of the two sub-departments of the marketing department.

```
[SwitchA] traffic classifier marketing operator or
[SwitchA-classifier-marketing] if-match acl 2003
[SwitchA-classifier-marketing] if-match acl 2004
[SwitchA-classifier-marketing] quit
```

# Configure behavior **remark_local_id** to mark traffic with local QoS ID 100.

```
[SwitchA] traffic behavior remark_local_id
[SwitchA-behavior-remark_local_id] remark qos-local-id 100
[SwitchA-behavior-remark_local_id] quit
```

# Configure class **marketing_car** to match the outgoing traffic of the two sub-departments of the Marketing department.

```
[SwitchA] traffic classifier marketing_car
[SwitchA-classifier-marketing_car] if-match qos-local-id 100
[SwitchA-classifier-marketing_car] quit
```

# Create behavior **marketing_car**, and configure traffic policing to limit the traffic rate to 204800 kbps.

```
[SwitchA] traffic behavior marketing_car
[SwitchA-behavior-marketing_car] car cir 204800
[SwitchA-behavior-marketing_car] quit
```

# In QoS policy **car**, associate class **marketing** with behavior **remark_local_id** to mark the outgoing traffic of the Marketing department with local QoS ID 100.

```
[SwitchA] qos policy car
[SwitchA-qospolicy-car] classifier marketing behavior remark_local_id
```

# In QoS policy **car**, associate class **marketing_car** with behavior **marketing_car** to limit the traffic rate of traffic with local QoS ID 100.

```
[SwitchA-qospolicy-car] classifier marketing_car behavior marketing_car
[SwitchA-qospolicy-car] quit
```

# Apply QoS policy **car** to the incoming traffic of Ten-GigabitEthernet1/0/1.

```
[SwitchA] interface ten-gigabitethernet 1/0/1
[SwitchA-Ten-GigabitEthernet1/0/1] qos apply policy car inbound
```

Configuring nesting

Nesting adds a VLAN tag to the matching packets to allow the VLAN-tagged packets to pass through the corresponding VLAN. For example, you can add an outer VLAN tag to packets from a customer network to a service provider network. This allows the packets to pass through the service provider network by carrying a VLAN tag assigned by the service provider.

Configuration procedure

To configure nesting:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Create a traffic class and enter traffic class view.	**traffic classifier** *classifier-name* [**operator** { **and** | **or** }]	By default, no traffic class exists.
3. Configure match criteria.	**if-match** *match-criteria*	By default, no match criterion is configured for a traffic class. For more information about the match criteria, see the **if-match** command in *ACL and QoS Command Reference*.
4. Return to system view.	**quit**	N/A
5. Create a traffic behavior and enter traffic behavior view.	**traffic behavior** *behavior-name*	By default, no traffic behavior exists.
6. Configure a VLAN tag adding action.	**nest top-most vlan** *vlan-id*	By default, no VLAN tag adding action is configured for a traffic behavior.
7. Return to system view.	**quit**	N/A
8. Create a QoS policy and enter QoS policy view.	**qos policy** *policy-name*	By default, no QoS policy exists.
9. Associate the traffic class with the traffic behavior in the QoS policy.	**classifier** *classifier-name* **behavior** *behavior-name* [**insert-before** *before-classifier-name*]	By default, no class-behavior association is configured for a QoS policy.
10. Return to system view.	**quit**	N/A

Step	Command	Remarks
11. Apply the QoS policy.	• Applying the QoS policy to an interface • Applying the QoS policy to a VLAN • Applying the QoS policy globally	Choose one of the application destinations as needed. By default, a QoS policy is not applied.

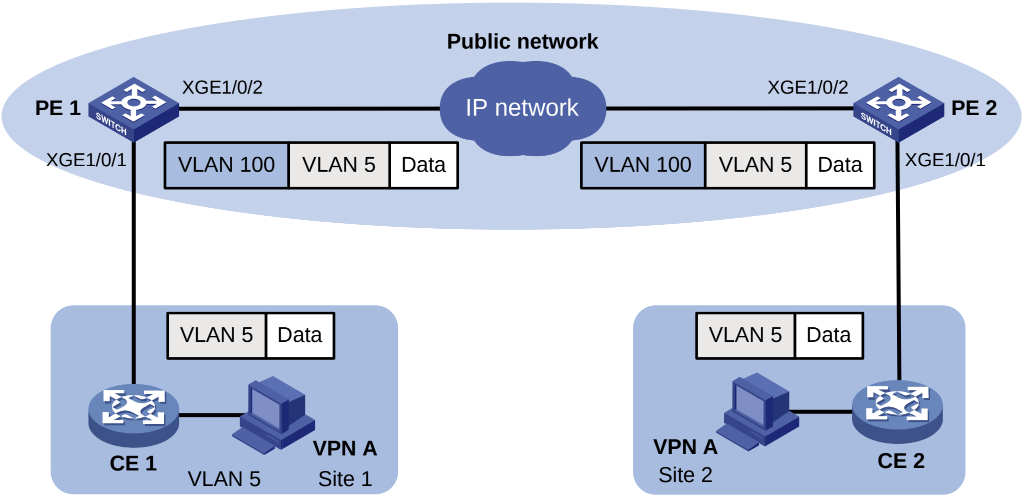
Configuration example

Network requirements

As shown in Figure 20, Site 1 and Site 2 in VPN A are two branches of a company, and they use VLAN 5 to transmit traffic. Because Site 1 and Site 2 are located in different areas, the two sites use the VPN access service of a service provider. The service provider assigns VLAN 100 to the two sites.

Configure nesting, so that the two branches can communicate through the service provider network.

Figure 20 Network diagram



Configuration procedure

Configuring PE 1

# Create a class named **test** to match packets with VLAN ID 5.

```
<PE1> system-view
[PE1] traffic classifier test
[PE1-classifier-test] if-match service-vlan-id 5
[PE1-classifier-test] quit
```

# Configure an action to add outer VLAN tag 100 in the traffic behavior named **test**.

```
[PE1] traffic behavior test
[PE1-behavior-test] nest top-most vlan 100
[PE1-behavior-test] quit
```

# Create a QoS policy named **test**, and associate class **test** with behavior **test** in the QoS policy.

```
[PE1] qos policy test
[PE1-qospolicy-test] classifier test behavior test
[PE1-qospolicy-test] quit
```

# Configure the downlink port Ten-GigabitEthernet 1/0/1 as a hybrid port, and assign the port to VLAN 100 as an untagged member.

```
[PE1] interface Ten-GigabitEthernet 1/0/1
[PE1-Ten-GigabitEthernet1/0/1] port link-type hybrid
[PE1-Ten-GigabitEthernet1/0/1] port hybrid vlan 100 untagged
```

# Apply QoS policy **test** to the incoming traffic of the downlink port Ten-GigabitEthernet 1/0/1.

```
[PE1-Ten-GigabitEthernet1/0/1] qos apply policy test inbound
[PE1-Ten-GigabitEthernet1/0/1] quit
```

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

```
[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
[PE1-Ten-GigabitEthernet1/0/2] quit
```

Configuring PE 2

Configure PE 2 in the same way PE 1 is configured.

Configuring traffic redirecting

Traffic redirecting redirects packets matching the specified match criteria to a location for processing.

The following redirect actions are supported:

- **Redirecting traffic to the CPU**—Redirects packets that require processing by the CPU to the CPU.
- **Redirecting traffic to an interface**—Redirects packets that require processing by an interface to the interface. This action applies only to Layer 2 packets, and the target interface must be a Layer 2 interface.

Configuration procedure

To configure traffic redirecting:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Create a traffic class and enter traffic class view.	**traffic classifier** *classifier-name* [**operator** { **and** | **or** }]	By default, no traffic class exists.
3. Configure match criteria.	**if-match** *match-criteria*	By default, no match criterion is configured for a traffic class. For more information about the match criteria, see the **if-match** command in *ACL and QoS Command Reference*.
4. Return to system view.	**quit**	N/A
5. Create a traffic behavior and enter traffic behavior view.	**traffic behavior** *behavior-name*	By default, no traffic behavior exists.
6. Configure a traffic redirecting action.	**redirect** { **cpu** | **interface** *interface-type interface-number* }	By default, no traffic redirecting action is configured for a traffic behavior. The actions of redirecting traffic to the CPU and redirecting traffic to an interface are mutually exclusive with each other in the same traffic behavior. The last redirecting action configured takes effect.
7. Return to system view.	**quit**	N/A

Step	Command	Remarks
8. Create a QoS policy and enter QoS policy view.	**qos policy** *policy-name*	By default, no QoS policy exists.
9. Associate the traffic class with the traffic behavior in the QoS policy.	**classifier** *classifier-name* **behavior** *behavior-name* [**insert-before** *before-classifier-name*]	By default, no class-behavior association is configured for a QoS policy.
10. Return to system view.	**quit**	N/A
11. Apply the QoS policy.	• Applying the QoS policy to an interface • Applying the QoS policy to a VLAN • Applying the QoS policy globally • Applying the QoS policy to a control plane	Choose one of the application destinations as needed. By default, a QoS policy is not applied.
12. (Optional.) Display traffic redirecting configuration.	**display traffic behavior user-defined** [*behavior-name*]	Available in any view.

Configuration example

Network requirements

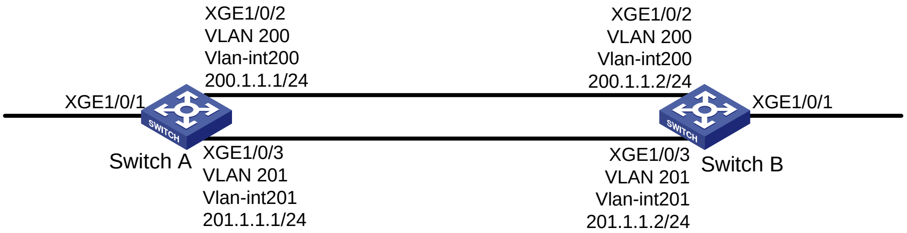
As shown in Figure 21:

- Switch A is connected to Switch B through two links. Switch A and Switch B are each connected to other devices.
- Ten-GigabitEthernet 1/0/2 of Switch A and Ten-GigabitEthernet 1/0/2 of Switch B belong to VLAN 200.
- Ten-GigabitEthernet 1/0/3 of Switch A and Ten-GigabitEthernet 1/0/3 of Switch B belong to VLAN 201.
- On Switch A, the IP address of VLAN-interface 200 is 200.1.1.1/24, and that of VLAN-interface 201 is 201.1.1.1/24.
- On Switch B, the IP address of VLAN-interface 200 is 200.1.1.2/24, and that of VLAN-interface 201 is 201.1.1.2/24.

Configure the actions of redirecting traffic to an interface to meet the following requirements:

- Packets with source IP address 2.1.1.1 received on Ten-GigabitEthernet 1/0/1 of Switch A are forwarded to Ten-GigabitEthernet 1/0/2.
- Packets with source IP address 2.1.1.2 received on Ten-GigabitEthernet 1/0/1 of Switch A are forwarded to Ten-GigabitEthernet 1/0/3.
- Other packets received on Ten-GigabitEthernet 1/0/1 of Switch A are forwarded according to the routing table.

Figure 21 Network diagram



Configuration procedure

# Create basic ACL 2000, and configure a rule to match packets with source IP address 2.1.1.1.

```
<SwitchA> system-view
[SwitchA] acl number 2000
[SwitchA-acl-basic-2000] rule permit source 2.1.1.1 0
[SwitchA-acl-basic-2000] quit
```

# Create basic ACL 2001, and configure a rule to match packets with source IP address 2.1.1.2.

```
[SwitchA] acl number 2001
[SwitchA-acl-basic-2001] rule permit source 2.1.1.2 0
[SwitchA-acl-basic-2001] quit
```

# Create a traffic class named **classifier_1**, and use ACL 2000 as the match criterion in the traffic class.

```
[SwitchA] traffic classifier classifier_1
[SwitchA-classifier-classifier_1] if-match acl 2000
[SwitchA-classifier-classifier_1] quit
```

# Create a traffic class named **classifier_2**, and use ACL 2001 as the match criterion in the traffic class.

```
[SwitchA] traffic classifier classifier_2
[SwitchA-classifier-classifier_2] if-match acl 2001
[SwitchA-classifier-classifier_2] quit
```

# Create a traffic behavior named **behavior_1**, and configure the action of redirecting traffic to Ten-GigabitEthernet 1/0/2.

```
[SwitchA] traffic behavior behavior_1
[SwitchA-behavior-behavior_1] redirect interface ten-gigabitethernet 1/0/2
[SwitchA-behavior-behavior_1] quit
```

# Create a traffic behavior named **behavior_2**, and configure the action of redirecting traffic to Ten-GigabitEthernet 1/0/3.

```
[SwitchA] traffic behavior behavior_2
[SwitchA-behavior-behavior_2] redirect interface ten-gigabitethernet 1/0/3
[SwitchA-behavior-behavior_2] quit
```

# Create a QoS policy named **policy**, associate traffic class **classifier_1** with traffic behavior **behavior_1**, and associate traffic class **classifier_2** with traffic behavior **behavior_2** in the QoS policy.

```
[SwitchA] qos policy policy
[SwitchA-qospolicy-policy] classifier classifier_1 behavior behavior_1
[SwitchA-qospolicy-policy] classifier classifier_2 behavior behavior_2
[SwitchA-qospolicy-policy] quit
```

# Apply the QoS policy named **policy** to the incoming traffic of Ten-GigabitEthernet 1/0/1.

```
[SwitchA] interface ten-gigabitethernet 1/0/1
[SwitchA-Ten-GigabitEthernet1/0/1] qos apply policy policy inbound
```

Configuring aggregate CAR

An aggregate CAR action is created globally and can be directly applied to interfaces or used in the traffic behaviors associated with different traffic classes to police multiple traffic flows as a whole. The total rate of the traffic flows must conform to the traffic policing specifications set in the aggregate CAR action.

Configuration procedure

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Configure an aggregate CAR action.	**qos car** *car-name* **aggregative cir** *committed-information-rate* [**cbs** *committed-burst-size* [**ebs** *excess-burst-size*]] [**green** *action* | **red** *action* | **yellow** *action*] * **qos car** *car-name* **aggregative cir** *committed-information-rate* [**cbs** *committed-burst-size*] **pir** *peak-information-rate* [**ebs** *excess-burst-size*] [**green** *action* | **red** *action* | **yellow** *action*] *	Use either of the commands. By default, no aggregate CAR action is configured.
3. Enter traffic behavior view.	**traffic behavior** *behavior-name*	N/A
4. Use the aggregate CAR in the traffic behavior.	**car name** *agg-car-name*	N/A

Displaying and maintaining aggregate CAR

Execute **display** commands in any view and **reset** commands in user view.

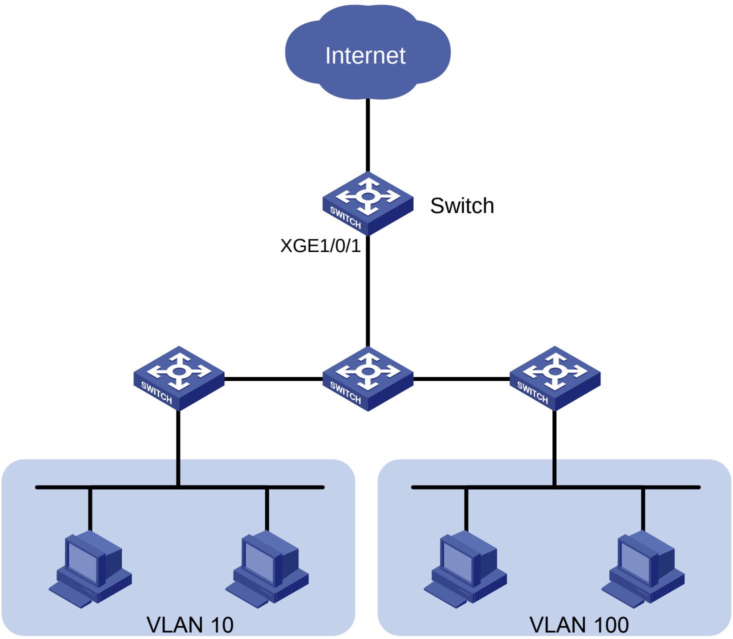
Task	Command
Display statistics for aggregate CAR actions.	**display qos car name** [*car-name*]
Clear statistics for aggregate CAR actions.	**reset qos car name** [*car-name*]

Aggregate CAR configuration example

Network requirements

As shown in Figure 22, configure an aggregate CAR to rate-limit the traffic of VLAN 10 and VLAN 100 received on Ten-GigabitEthernet 1/0/1 by using these parameters: CIR is 2560 kbps, CBS is 20480 bytes, and the action for red packets is **discard**.

Figure 22 Network diagram



Configuration procedure

# Configure an aggregate CAR according to the rate limit requirements.

```
<Switch> system-view
[Switch] qos car aggcar-1 aggregative cir 2560 cbs 20480 red discard
```

# Create class 1 to match traffic of VLAN 10. Create behavior 1 and use the aggregate CAR in the behavior.

```
[Switch] traffic classifier 1
[Switch-classifier-1] if-match customer-vlan-id 10
[Switch-classifier-1] quit
[Switch] traffic behavior 1
[Switch-behavior-1] car name aggcar-1
[Switch-behavior-1] quit
```

# Create class 2 to match traffic of VLAN 100. Create behavior 2 and use the aggregate CAR in the behavior.

```
[Switch] traffic classifier 2
[Switch-classifier-2] if-match customer-vlan-id 100
[Switch-classifier-2] quit
[Switch] traffic behavior 2
[Switch-behavior-2] car name aggcar-1
[Switch-behavior-2] quit
```

# Create QoS policy **car**, associate class 1 with behavior 1, and associate class 2 with behavior 2.

```
[Switch] qos policy car
[Switch-qospolicy-car] classifier 1 behavior 1
[Switch-qospolicy-car] classifier 2 behavior 2
[Switch-qospolicy-car] quit
```

Apply the QoS policy to the incoming traffic of Ten-GigabitEthernet 1/0/1.

```
[Switch] interface ten-gigabitethernet 1/0/1
[Switch-Ten-GigabitEthernet1/0/1]qos apply policy car inbound
```

Configuring class-based accounting

Class-based accounting collects statistics (in packets or bytes) on a per-traffic class basis. For example, you can define the action to collect statistics for traffic sourced from a certain IP address. By analyzing the statistics, you can determine whether anomalies have occurred and what action to take.

Configuration procedure

To configure class-based accounting:

Step		Command	Remarks
1.	Enter system view.	**system-view**	N/A
2.	Create a traffic class and enter traffic class view.	**traffic classifier** *classifier-name* [**operator** { **and** | **or** }]	By default, no traffic class is configured.
3.	Configure match criteria.	**if-match** *match-criteria*	By default, no match criterion is configured. For more information about the **if-match** command, see *ACL and QoS Command Reference*.
4.	Return to system view.	**quit**	N/A
5.	Create a traffic behavior and enter traffic behavior view.	**traffic behavior** *behavior-name*	By default, no traffic behavior is configured.
6.	Configure the accounting action.	**accounting** { **byte** | **packet** } *	By default, no traffic accounting action is configured.
7.	Return to system view.	**quit**	N/A
8.	Create a QoS policy and enter QoS policy view.	**qos policy** *policy-name*	By default, no QoS policy is configured.
9.	Associate the traffic class with the traffic behavior in the QoS policy.	**classifier** *classifier-name* **behavior** *behavior-name* [**insert-before** *before-classifier-name*]	By default, a traffic class is not associated with a traffic behavior.
10.	Return to system view.	**quit**	N/A
11.	Apply the QoS policy.	• Applying the QoS policy to an interface • Applying the QoS policy to a VLAN • Applying the QoS policy globally • Applying the QoS policy to a control plane • Applying the QoS policy to a user profile	Choose one of the application destinations as needed. By default, no QoS policy is applied.

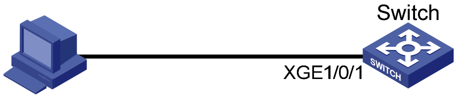
Step	Command	Remarks
12. Display traffic accounting configuration.	- **display qos policy control-plane slot** *slot-number* - **display qos policy global** [**slot** *slot-number*] [**inbound** | **outbound**] - **display qos policy interface** [*interface-type interface-number*] [**inbound** | **outbound**] - **display qos vlan-policy** { **name** *policy-name* | **vlan** [*vlan-id*] } [**slot** *slot-number*] [**inbound** | **outbound**]	Available in any view.

Configuration example

Network requirements

As shown in Figure 23, configure class-based accounting on Ten-GigabitEthernet 1/0/1 to collect statistics for the incoming packets with 1.1.1.1/24 as the source IP address.

Figure 23 Network diagram



Configuration procedure

# Create basic ACL 2000, and configure a rule to match packets with source IP address 1.1.1.1.

```
<Switch> system-view
[Switch] acl number 2000
[Switch-acl-basic-2000] rule permit source 1.1.1.1 0
[Switch-acl-basic-2000] quit
```

# Create a traffic class named **classifier_1**, and use ACL 2000 as the match criterion in the traffic class.

```
[Switch] traffic classifier classifier_1
[Switch-classifier-classifier_1] if-match acl 2000
[Switch-classifier-classifier_1] quit
```

# Create a traffic behavior named **behavior_1**, and configure the class-based accounting action.

```
[Switch] traffic behavior behavior_1
[Switch-behavior-behavior_1] accounting packet
[Switch-behavior-behavior_1] quit
```

# Create a QoS policy named **policy**, and associate traffic class **classifier_1** with traffic behavior **behavior_1** in the QoS policy.

```
[Switch] qos policy policy
[Switch-qospolicy-policy] classifier classifier_1 behavior behavior_1
[Switch-qospolicy-policy] quit
```

# Apply the QoS policy named **policy** to the incoming traffic of Ten-GigabitEthernet 1/0/1.

```
[Switch] interface Ten-GigabitEthernet 1/0/1
[Switch-Ten-GigabitEthernet1/0/1] qos apply policy policy inbound
[Switch-Ten-GigabitEthernet1/0/1] quit
```

# Display traffic statistics to verify the configuration.

```
[Switch] display qos policy interface Ten-GigabitEthernet 1/0/1

Interface: Ten-GigabitEthernet1/0/1

  Direction: Inbound

  Policy: policy
   Classifier: classifier_1
     Operator: AND
     Rule(s) :
      If-match acl 2000
     Behavior: behavior_1
      Accounting enable:
        28529 (Packets)
```

Appendixes

Appendix A Default priority maps

For the default **dscp-dscp** priority maps, an input value yields a target value equal to it.

Table 7 Default dot1p-lp and dot1p-dp priority maps

Input priority value	dot1p-lp map	dot1p-dp map
dot1p	**lp**	**dp**
0	2	0
1	0	0
2	1	0
3	3	0
4	4	0
5	5	0
6	6	0
7	7	0

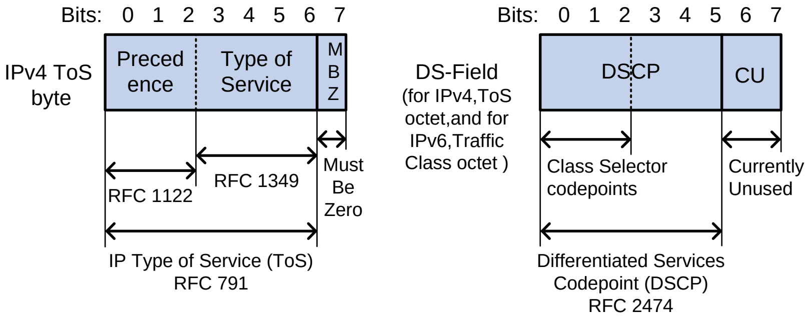
Table 8 Default dscp-dp and dscp-dot1p priority maps

Input priority value	dscp-dp map	dscp-dot1p map
dscp	**dp**	**dot1p**
0 to 7	0	0
8 to 15	0	1
16 to 23	0	2
24 to 31	0	3
32 to 39	0	4
40 to 47	0	5
48 to 55	0	6
56 to 63	0	7

Appendix B Introduction to packet precedences

IP precedence and DSCP values

Figure 24 ToS and DS fields


Bits: 0 1 2 3 4 5 6 7
IPv4 ToS byte
Preced ence
Type of Service
M B Z
RFC 1122
RFC 1349
Must Be Zero
IP Type of Service (ToS) RFC 791
Bits: 0 1 2 3 4 5 6 7
DS-Field (for IPv4,ToS octet,and for IPv6,Traffic Class octet)
DSCP
CU
Class Selector codepoints
Currently Unused
Differentiated Services Codepoint (DSCP) RFC 2474

As shown in Figure 24, the ToS field in the IP header contains eight bits. The first three bits (0 to 2) represent IP precedence from 0 to 7. According to RFC 2474, the ToS field is redefined as the differentiated services (DS) field, where a DSCP value is represented by the first six bits (0 to 5) and is in the range 0 to 63. The remaining two bits (6 and 7) are reserved.

Table 9 IP precedence

IP precedence (decimal)	IP precedence (binary)	Description
0	000	Routine
1	001	priority
2	010	immediate
3	011	flash
4	100	flash-override
5	101	critical
6	110	internet
7	111	network

Table 10 DSCP values

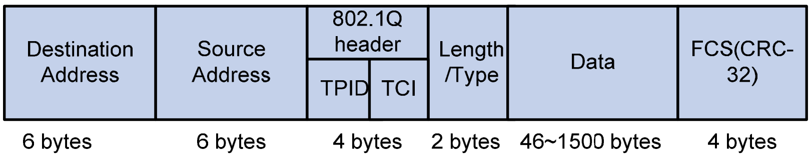
DSCP value (decimal)	DSCP value (binary)	Description
46	101110	ef
10	001010	af11
12	001100	af12
14	001110	af13
18	010010	af21
20	010100	af22
22	010110	af23
26	011010	af31

DSCP value (decimal)	DSCP value (binary)	Description
28	011100	af32
30	011110	af33
34	100010	af41
36	100100	af42
38	100110	af43
8	001000	cs1
16	010000	cs2
24	011000	cs3
32	100000	cs4
40	101000	cs5
48	110000	cs6
56	111000	cs7
0	000000	be (default)

802.1p priority

802.1p priority lies in the Layer 2 header. It applies to occasions where Layer 3 header analysis is not needed and QoS must be assured at Layer 2.

Figure 25 An Ethernet frame with an 802.1Q tag header



As shown in Figure 25, the 4-byte 802.1Q tag header consists of the 2-byte tag protocol identifier (TPID) and the 2-byte tag control information (TCI). The value of the TPID is 0x8100. Figure 26 shows the format of the 802.1Q tag header. The Priority field in the 802.1Q tag header is called the "802.1p priority", because its use is defined in IEEE 802.1p. Table 11 shows the values for 802.1p priority.

Figure 26 802.1Q tag header

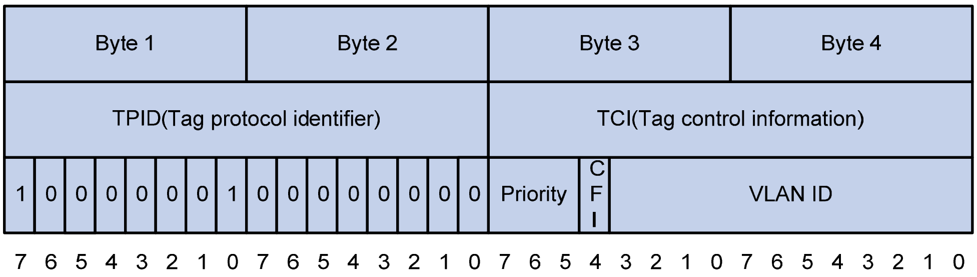


Table 11 Description on 802.1p priority

802.1p priority (decimal)	802.1p priority (binary)	Description
0	000	best-effort
1	001	background
2	010	spare
3	011	excellent-effort
4	100	controlled-load
5	101	video
6	110	voice
7	111	network-management

Configuring time ranges

You can implement a service based on the time of the day by applying a time range to it. A time-based service takes effect only in time periods specified by the time range. For example, you can implement time-based ACL rules by applying a time range to them. If a time range does not exist, the service based on the time range does not take effect.

The following basic types of time ranges are available:

- **Periodic time range**—Recurs periodically on a day or days of the week.
- **Absolute time range**—Represents only a period of time and does not recur.

A time range is uniquely identified by the time range name. You can create a maximum of 1024 time ranges, each with a maximum of 32 periodic statements and 12 absolute statements. The active period of a time range is calculated as follows:

1. Combining all periodic statements.
2. Combining all absolute statements.
3. Taking the intersection of the two statement sets as the active period of the time range.

Configuration procedure

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Create or edit a time range.	**time-range** *time-range-name* { *start-time* **to** *end-time days* [**from** *time1 date1*] [**to** *time2 date2*] | **from** *time1 date1* [**to** *time2 date2*] | **to** *time2 date2* }	No time range exists.

Displaying and maintaining time ranges

Execute the **display** command in any view.

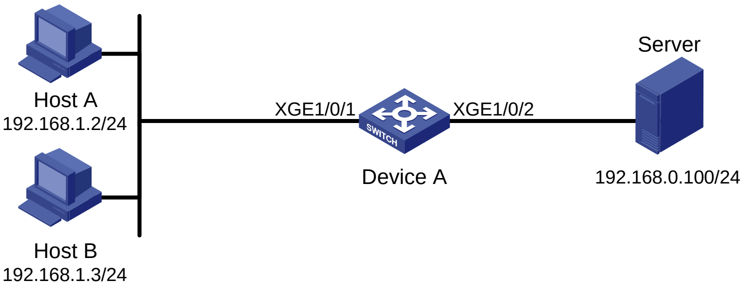
Task	Command
Display time range configuration and status.	**display time-range** { *time-range-name* | **all** }

Time range configuration example

Network requirements

As shown in Figure 27, configure an ACL on Device A to allow Host A to access the server only during 8:00 and 18:00 on working days from June 2011 to the end of the year.

Figure 27 Network diagram



Configuration procedure

# Create a periodic time range during 8:00 and 18:00 on working days from June 2011 to the end of the year.

```
<DeviceA> system-view
[DeviceA] time-range work 8:0 to 18:0 working-day from 0:0 6/1/2011 to 24:0 12/31/2011
```

# Create an IPv4 basic ACL numbered 2001, and configure a rule in the ACL to permit packets only from 192.168.1.2/32 during the time range **work**.

```
[DeviceA] acl number 2001
[DeviceA-acl-basic-2001] rule permit source 192.168.1.2 0 time-range work
[DeviceA-acl-basic-2001] rule deny source any time-range work
[DeviceA-acl-basic-2001] quit
```

# Apply IPv4 basic ACL 2001 to filter outgoing packets on interface Ten-GigabitEthernet 1/0/2.

```
[DeviceA] interface Ten-GigabitEthernet 1/0/2
[DeviceA-Ten-GigabitEthernet1/0/2] packet-filter 2001 outbound
[DeviceA-Ten-GigabitEthernet1/0/2] quit
```

Verifying the configuration

# Display time range configuration and status on Device A.

```
[DeviceA] display time-range all
Current time is 13:58:35 6/20/2011 Monday

Time-range : work ( Active )
 08:00 to 18:00 working-day
 from 00:00 6/1/2011 to 00:00 1/1/2012
```

The output shows that the time range **work** is active.

Configuring data buffers

An interface stores outgoing packets in the egress buffer when congestion occurs.

An egress buffer uses the following types of resources:

- **Cell resources**—Store packets. The buffer uses cell resources based on packet sizes. Suppose a cell resource provides 208 bytes. The buffer allocates one cell resource to a 128-byte packet and two cell resources to a 300-byte packet.
- **Packet resources**—Store packet pointers. A packet pointer indicates where the packet is located in cell resources. The buffer uses one packet resource for each incoming or outgoing packet.

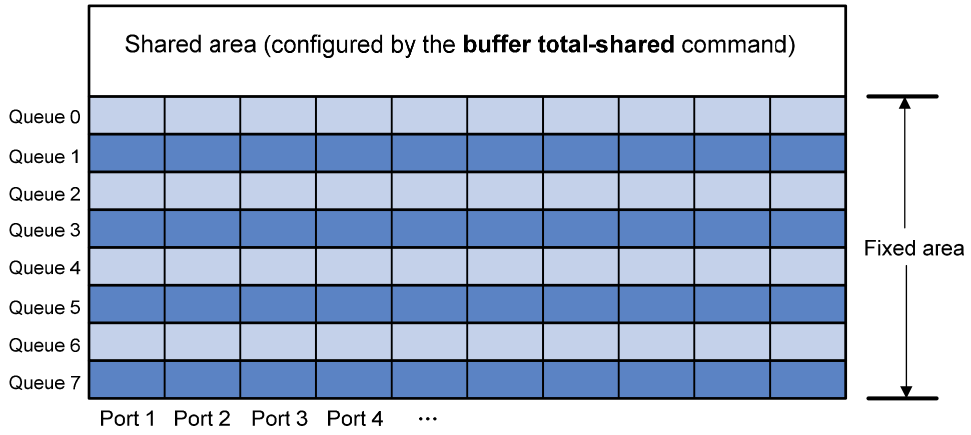
Each type of resources has a fixed area and a shared area.

- **Fixed area**—Partitioned into queues, each of which is equally divided by all the interfaces on a device, as shown in Figure 28. When congestion occurs, the following rules apply:
  a. An interface first uses the relevant queues of the fixed area to store packets.
  b. When a queue is full, the interface uses the space for the queue in the shared area.
  c. When the queue in the shared area is also full, the interface discards subsequent packets.

  The system allocates the fixed area among queues as specified by the user. Even if a queue is not full, other queues cannot preempt its space. Similarly, the share of a queue for an interface cannot be preempted by other interfaces even if it is not full.
- **Shared area**—Partitioned into queues, each of which is not equally divided by the interfaces, as shown in Figure 28. The system determines the actual shared-area ratio for each queue according to user configuration and the number of packets actually sent. If a queue is not full, other queues can preempt its space.

  The system puts packets received on all interfaces into a queue in the order they arrive. When the queue is full, subsequent packets are dropped.

Figure 28 Fixed area and shared area



Configuration task list

You can configure data buffers either automatically by enabling the Burst function or manually.

If you have configured data buffers in one way, delete the configuration before using the other way. Otherwise, the new configuration does not take effect.

To configure the data buffer, perform the following tasks:

Tasks at a glance
Perform one of the following tasks: • Enabling the Burst function • Configuring data buffers manually ○ Configuring the total shared-area ratio ○ Setting the maximum shared-area ratio for a queue ○ Setting the fixed-area ratio for a queue ○ Applying data buffer configuration

Enabling the Burst function

The Burst function enables the device to automatically allocate cell and packet resources. It is well suited to the following scenarios:

- Broadcast or multicast traffic is intensive, resulting in bursts of traffic.
- Traffic enters and goes out in one of the following ways:
    - Enters from a high-speed interface and goes out of a low-speed interface.
    - Enters from multiple same-rate interfaces at the same time and goes out of an interface with the same rate.

To enable the Burst function:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Enable the Burst function.	**burst-mode enable**	By default, the Burst function is disabled.

Configuring data buffers manually

△ CAUTION:

- To avoid impact on the forwarding function of the system, do not manually change data buffer settings. If large buffer spaces are needed, use the Burst function.
- Manually configuring data buffers might cause generic flow control and PFC to operate incorrectly. For more information about generic flow control and PFC, see *Layer 2—LAN Switching Configuration Guide*.

The switch only supports configuring cell resources.

Configuring the total shared-area ratio

Each type of resources of a buffer, packet or cell, has a fixed size. After you set the total shared-area ratio for a type of resources, the rest is automatically assigned to the fixed area.

To configure the total shared-area ratio:

Step		Command	Remarks
1.	Enter system view.	**system-view**	N/A
2.	Configure the total shared-area ratio.	**buffer egress** [**slot** *slot-number*] **cell total-shared ratio** *ratio-value*	By default, the total shared-area ratio is 84% of the buffer.

Setting the maximum shared-area ratio for a queue

By default, all queues have an equal share of the shared area. This task allows you to change the maximum shared-area ratio for a queue. The unconfigured queues use the default setting.

The actual maximum shared-area ratio for each queue is determined by the chip based on your configuration and the number of packets to be sent.

To set the maximum shared-area ratio for a queue:

Step		Command	Remarks
1.	Enter system view.	**system-view**	N/A
2.	Set the maximum shared-area ratio for a queue.	**buffer egress** [**slot** *slot-number*] **cell queue** *queue-id* **shared ratio** *ratio-value*	The default setting is 33% for each queue.

For the maximum shared-area ratio for a queue, the percentage values 0 to 100 are divided into 10 ranges. Table 12 shows the effective values that correspond to the configured values of *ratio-value*.

Table 12 Mapping between values of *ratio-value* and effective values

Value of *ratio-value*	Effective value
0 to 1	1
2 to 3	3
4 to 7	6
8 to 16	11
17 to 29	20
30 to 42	33
43 to 60	50
61 to 76	67
77 to 86	80
87 to 100	89

Setting the fixed-area ratio for a queue

By default, all queues have an equal share of the fixed area. This task allows you to change the fixed-area ratio for a queue. The unconfigured queues equally share the remaining part.

The fixed-area space for a queue cannot be used by other queues. It is also called the minimum guaranteed buffer.

When you set the fixed-area ratio for a queue, follow these restrictions and guidelines:

- The sum of ratios configured for all queues cannot be greater than or equal to 100%. Queues 5, 6, and 7 must have available fixed-area space.
- After you configure the fixed-area ratios for some queues, the other queues each are assigned an equal share of the remaining part of the fixed area. The **display buffer queue** command displays the preceding whole number for each assignment result. Therefore, the sum of the ratios for all queues might be less than 100%.

To set the fixed-area ratio for a queue:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Set the fixed-area ratio for a queue.	**buffer egress** [**slot** *slot-number*] **cell queue** *queue-id* **guaranteed ratio** *ratio-value*	The default setting is 12.5% for each queue, but the default value in the **display buffer queue** command output is 13%.

Applying data buffer configuration

Perform this task to apply the data buffer configuration.

You cannot directly modify the applied configuration. To modify the configuration, you must cancel the application, reconfigure data buffers, and reapply the configuration.

To apply data buffer configuration:

Step	Command
1. Enter system view.	**system-view**
2. Apply data buffer configuration.	**buffer apply**

Displaying and maintaining data buffers

Execute **display** commands in any view.

Task	Command
Display data buffer configuration.	**display buffer** [**slot** *slot-number*] [**queue** [*queue-id*]]
Display data buffer usage.	**display buffer usage** [**slot** *slot-number*]

Configuring QCN

Quantized Congestion Notification (QCN) is an end-to-end congestion notification mechanism that can reduce packet loss and delay in Layer 2 networks by actively sending reverse notifications. As part of data center standards, QCN is primarily used in data center networks.

Basic concepts

- **Reaction point (RP)**—A source end host that supports QCN.
- **Congestion point (CP)**—A congestion detection device that is enabled with QCN.
- **Congestion notification message (CNM)**—A message transmitted by a CP to an RP when a queue on the CP is congested.
- **Congestion controlled flow (CCF)**—A flow of frames with the same priority value. A CP assigns frames of the same CCF to one queue before forwarding them.
- **Congestion notification tag (CN tag)**—Identifies a CCF. Devices in a CND must be able to process packets with a CN tag.
- **Congestion notification priority (CNP)**—An 802.1p priority that is enabled with QCN. The value of that 802.1p priority is called a Congestion Notification Priority Value (CNPV).
- **Congestion notification domain (CND)**—A set of RPs and CPs with QCN enabled for a CNPV.
- **Congestion point identifier (CPID)**—An 8-byte unique identifier for a CP in the network.
- **Quantized feedback (QntzFb)**—A 6-bit quantized feedback value indicating the extent of congestion.

QCN message format

Data flow format

An RP can add CN tags to outgoing Ethernet frames to distinguish between CCFs. A CN tag defines a CCF.

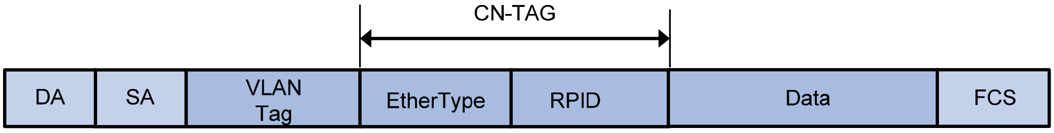
As shown in Figure 29, the CN tag contains the following fields:

- **EtherType**—Indicates the Ethernet type of the data packet, 2 bytes in length and assigned a value of 0x22E9.
- **RPID**—Locally assigned and 2 bytes in length. When receiving a CNM, the RP uses this field to identify the CCF that causes congestion and then rate limits that CCF.

When only one CCF exists, the RP may not add a CN tag to packets. In this case, the triggered CNM carries a CN tag with the RPID as 0.

A CN tag is confined within its CND. When a packet leaves a CND, the CN tag is stripped off.

Figure 29 Data flow format



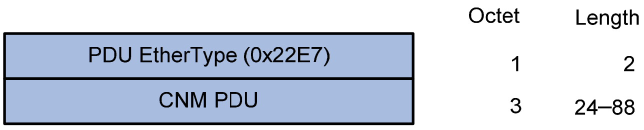
CNM format

When a CP detects the congestion state by sampling frames, it sends CNMs to the RPs.

The CP constructs a CNM as follows:

- Uses the source MAC address of the sampled frame as the destination MAC address.
- Uses the destination MAC address of the sampled frame as the source MAC addresses.
- Copies the VLAN tag and CN tag of the sampled frame.
- Places the data as shown in Figure 30.
    - **PDU EtherType**—2 bytes in length. It indicates the Ethernet type of the PDU and has a value of 0x22E7.
    - **CNM PDU**—24 to 88 bytes of payload of the PDU.

Figure 30 CNM PDU format



As shown in Figure 31, a payload contains the following fields:

Field	Length	Description
Version	4 bits	Its value is fixed at 0.
ReserverV	6 bits	Its value is fixed at 0.
Quantized Feedback	6 bits	Quantized value indicating the extent of congestion.
CPID	8 bytes	Identifies the CP where congestion occurs.
cnmQoffset	2 bytes	Indicates the difference between instantaneous queue size at the sampling point and desired queue length.
cnmQdelta	2 bytes	Indicates the difference between instantaneous queue sizes at the current sampling point and at the previous sampling point.
Encapsulated priority	2 bytes	Priority of the sampled frame that triggered the CNM.
Encapsulated destination MAC address	6 bytes	Destination MAC address of the sampled frame that triggered the CNM.
Encapsulated MSDU length	2 bytes	Number of bytes in the Encapsulated MSDU field of the sampled frame that triggered the CNM.
Encapsulated MSDU	0 to 64 bytes	Initial bytes of the Encapsulated MSDU field of the sampled frame that triggered the CNM.

Figure 31 CNM PDU format

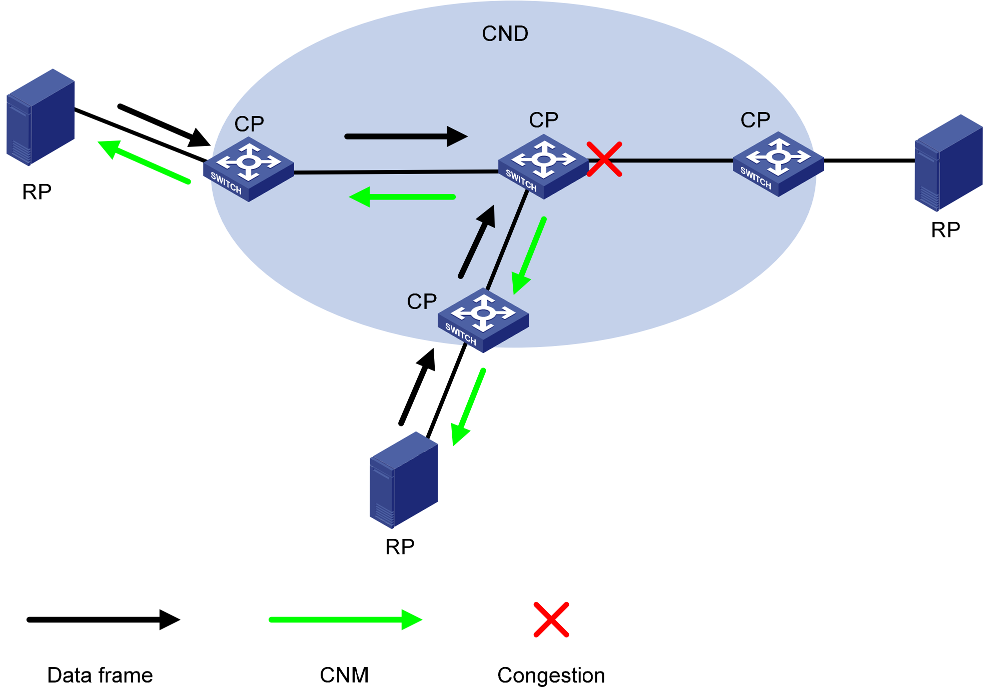
	Octet	Length
Version	1	4 bits
ReservedV	1, 2	6 bits
Quantized Feedback	2	6 bits
Congestion Point Identifier (CPID)	3	8
cnmQOffset	11	2
cnmQDelta	13	2
Encapsulated priority	15	2
Encapsulated destination MAC address	17	6
Encapsulated MSDU length	23	2
Encapsulated MSDU	25	0–64

How QCN works

Figure 32 shows how QCN works.

- The CP periodically samples frames from queues that are enabled with QCN and sends CNMs to the RPs when congestion occurs.
- The RPs reduce their transmission rates when receiving CNMs. The RPs also periodically probe the bandwidth and increase their transmission rates if they fail to receive CNMs for a specific period of time.

Figure 32 How QCN works



QCN algorithm

The QCN algorithm includes the CP algorithm and the RP algorithm.

CP algorithm

The CP measures the queue size by periodically sampling frames and computes the congestion state based on the sampling result.

As shown in Figure 33, the CP algorithm includes the following parameters:

- **Q**—Indicates the instantaneous queue size at the sampling point.
- **Qeq**—Indicates the desired queue size.
- **Qold**—Indicates the queue size at the previous sampling point.
- **Fb**—Indicates the extent of congestion in the form of a quantized value.

The following formulas apply:

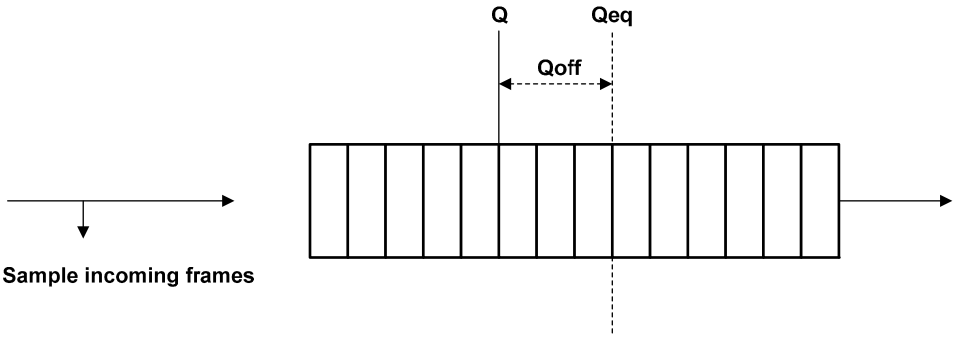
- Qoff = Q – Qeq
- Qδ = Q – Qold
- Fb = – (Qoff + wQδ)

where w is a constant to control the weight of Qδ in determining the value of Fb.

The CP determines whether to generate CNMs based on the Fb value.

- When Fb ≥ 0, no congestion occurs, and the CP does not generate a CNM.
- When Fb < 0, congestion occurs, and the CP generates an CNM containing the QntzFb. QntzFb is the quantized value of |Fb| and is calculated according to the following rules:
  - If Fb < – Qeq x (2 x w + 1), QntzFb takes the maximum value of 63.
  - Otherwise, QntzFb = – Fb x 63/(Qeq x (2 x w + 1)).

Figure 33 Congestion detection



RP algorithm

An RP decreases its transmission rate based on the value of |Fb| in the received CNM. The greater the Fb value, the lower the RP reduces its transmission rate. After the RP reduces its transmission rate, the RP gradually increases the transmission rate to the original level.

CND

A CND is a set of RPs and CPs enabled with QCN for a CNPV. CNDs are identified based on CNPVs. Devices enabled with QCN for a CNPV are assigned to the corresponding CND. A CNPV-based CND prevents traffic from outside the CND from entering the CND. If a frame from outside the CND includes the CNPV, the 802.1p priority value of the frame is mapped to a configured alternate priority value.

CND defense mode

Each interface on a device in a CND has a defense mode, which is statically configured or negotiated through LLDP.

The following defense modes are available:

- **disabled**—Disables congestion notification and performs priority mapping according to the priority mapping table.
- **edge**—Maps the priority of incoming frames with a CNPV to an alternate priority and removes CN tags before sending out the frames.
- **interior**—Does not alter the priority of incoming frames with a CNPV and removes CN tags before sending out the frames.
- **interiorReady**—Does not alter the priority of incoming frames with a CNPV and retains CN tags when sending out the frames.

Priority mapping

Incoming frames with a CNPV are assigned to the corresponding output queue enabled with QCN. Traffic with other priority values cannot enter that output queue. Priority-to-queue mappings are determined by the QoS priority mapping table (see "Configuring priority mapping").

Modifying the priority mapping table for traffic with specific CNPVs might cause the system to fail to detect congestion.

When you map multiple 802.1p priorities to one queue, all packets with these 802.1p priorities will be included when determining congestion conditions. Therefore, do not map 802.1p priorities not enabled with QCN to a queue enabled with QCN.

Marking actions configured in QoS policies affect priority mapping. For information about marking actions, see "Configuring priority marking."

The priority trust mode must be configured as the 802.1p priority. For information about configuring trust modes, see "Configuring priority mapping."

The default port priority cannot be the same as the CNPV. For information about port priority, see "Configuring priority mapping."

Protocols and standards

IEEE 802.1Qau, Congestion notification

QCN configuration task list

Tasks at a glance
(Required.) Enabling QCN globally
Configuring CND settings • (Required.) Configuring global CND settings • (Optional.) Configuring CND settings for an interface
(Optional.) Configuring congestion detection parameters

Enabling QCN globally

QCN settings take effect only after you enable QCN globally.

Configuration prerequisites

Before you enable QCN globally, enable LLDP. For more information about LLDP, see *Layer 2—LAN Switching Configuration Guide*.

Configuration procedure

To enable QCN globally:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Enable QCN globally.	**qcn-enable**	By default, QCN is disabled globally. When QCN is disabled globally, the following events occur: • All QCN settings become invalid but still exist. • The switch stops LLDP negotiation and does not process or carry CN TLVs in LLDP packets.

Configuring CND settings

You can configure CND settings both globally or for a specific interface. The interface-level CND settings take precedence over global settings.

Configuring global CND settings

Perform this task to assign a switch to a CND identified by the specified CNPV.

After you assign a switch to a CND, the switch can detect congestion for packets within the CND.

You can assign a switch to multiple CNDs by specifying multiple CNPVs for the switch. For example, a switch can be assigned to CND 1, CND 2, and CND 3 and have an alternate priority of 0 in all three CNDs. The following table shows priority mappings:

dot1p	CNPV	Alternate priority
0	N/A	N/A
1	1	0
2	2	0
3	3	0
4	N/A	N/A
5	N/A	N/A
6	N/A	N/A
7	N/A	N/A

To configure global CND settings:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Configure global CND settings.	**qcn priority** *priority-value* { **admin** [**defense-mode** { **disabled** | **edge** | **interior** | **interior-ready** } **alternate** *alternate-value*] | **auto** }	By default, a switch does not belong to any CND.

Configuring CND settings for an interface

You can configure interface CND settings to meet your granular requirements.

You must assign a switch to a CND before you configure CND settings for individual interfaces.

To configure CND settings for an interface:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Enter interface view.	**interface** *interface-type interface-number*	N/A
3. Configure CND settings for the interface.	**qcn port priority** *priority-value* { **admin** [**defense-mode** { **disabled** | **edge** | **interior** | **interior-ready** } **alternate** *alternate-value*] | **auto** }	By default, the global CND settings apply.

Configuring congestion detection parameters

Perform this task to detect congestion for packets in a CND. You configure congestion detection parameters in a profile.

Before you configure congestion detection parameters, you must assign the switch to the CND.

To configure congestion detection parameters:

Step		Command	Remarks
1.	Enter system view.	**system-view**	N/A
2.	Create a profile.	**qcn profile** *profile-id* **set-point** *length-value* **weight** *weight-value*	By default, no user-created profiles exist. The system automatically creates the default profile (profile 0), which has a desired queue length of 26000 bytes and a weight value of 1. You cannot modify the default profile.
3.	Bind the profile to a CND.	**qcn priority** *priority-value* **profile** *profile-id*	By default, the default profile is bound to a CND.

Displaying and maintaining QCN

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display global CND settings.	**display qcn global** [**slot** *slot-number*]
Display the CND settings for an interface.	**display qcn global** [*interface-type interface-number*]
Display profile settings.	**display qcn profile** [*profile-id* | **default**] [**slot** *slot-number*]
Display CP statistics for an interface.	**display qcn cp interface** [*interface-type interface-number*] [**priority** *priority-value*]
Clear CP statistics for an interface.	**reset qcn cp interface** [*interface-type interface-number*] [**priority** *priority-value*]

QCN configuration examples

Basic QCN configuration example

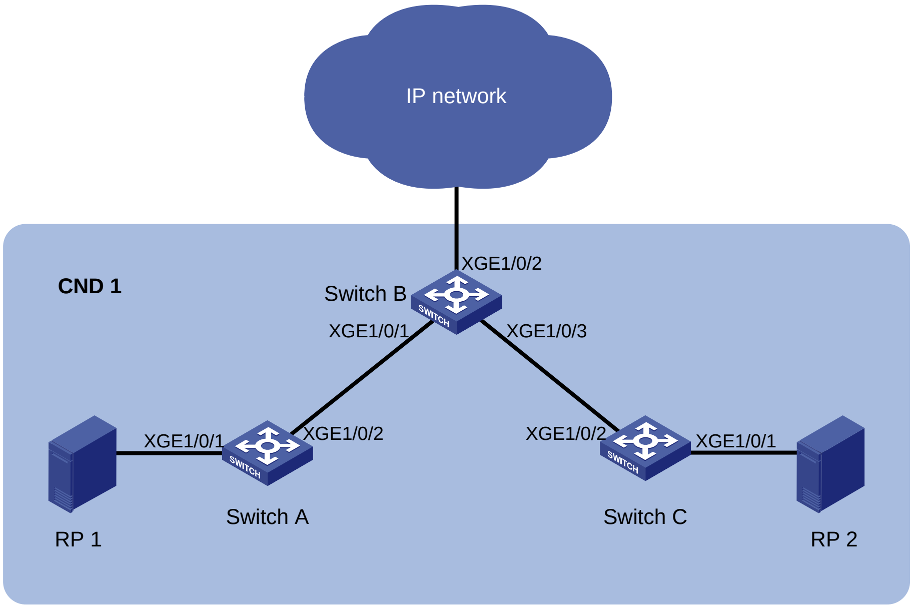
Network requirements

As shown in Figure 34, RP 1 and RP 2 are in the same VLAN and both support QCN.

Configure QCN for CNPV 1 to meet the following requirements:

- Switch A, Switch B, and Switch C detect congestion for traffic with 802.1p priority 1.
- Switch A, Switch B, and Switch C do not detect congestion for all other traffic.

Figure 34 Network diagram



Configuration procedure

1. Configure Switch A:

# Create VLAN 100, and assign Ten-GigabitEthernet 1/0/1 to the VLAN.

```
<SwitchA> system-view
[SwitchA] vlan 100
[SwitchA-vlan100] port ten-gigabitethernet 1/0/1
[SwitchA-vlan100] quit
```

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

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

# Enable LLDP globally.

```
[SwitchA] lldp global enable
```

# Enable CN TLV advertising on Ten-GigabitEthernet 1/0/1.

```
[SwitchA] interface ten-gigabitethernet 1/0/1
[SwitchA-Ten-GigabitEthernet1/0/1] lldp tlv-enable dot1-tlv congestion-notification
[SwitchA-Ten-GigabitEthernet1/0/1] quit
```

# Enable CN TLV advertising on Ten-GigabitEthernet 1/0/2.

```
[SwitchA] interface ten-gigabitethernet 1/0/2
[SwitchA-Ten-GigabitEthernet1/0/2] lldp tlv-enable dot1-tlv congestion-notification
[SwitchA-Ten-GigabitEthernet1/0/2] quit
```

# Enable QCN globally.

```
[SwitchA] qcn enable
```

# Assign the switch to the CND with CNPV 1, and configure all interfaces to negotiate the defense mode and alternate priority by using LLDP.

```
[SwitchA] qcn priority 1 auto
```

2. Configure Switch B:

   # Create VLAN 100.

```
<SwitchB> system-view
[SwitchB] vlan 100
[SwitchB-vlan100] quit
```

   # Configure the following interfaces as trunk ports, and assign all of them to VLAN 100:

   - Ten-GigabitEthernet 1/0/1.
   - Ten-GigabitEthernet 1/0/2.
   - Ten-GigabitEthernet 1/0/3.

```
[SwitchB] interface ten-gigabitethernet 1/0/1
[SwitchB-Ten-GigabitEthernet1/0/1] port link-type trunk
[SwitchB-Ten-GigabitEthernet1/0/1] port trunk permit vlan 100
[SwitchB-Ten-GigabitEthernet1/0/1] quit
[SwitchB] interface ten-gigabitethernet 1/0/2
[SwitchB-Ten-GigabitEthernet1/0/2] port link-type trunk
[SwitchB-Ten-GigabitEthernet1/0/2] port trunk permit vlan 100
[SwitchB-Ten-GigabitEthernet1/0/2] quit
[SwitchB] interface ten-gigabitethernet 1/0/3
[SwitchB-Ten-GigabitEthernet1/0/3] port link-type trunk
[SwitchB-Ten-GigabitEthernet1/0/3] port trunk permit vlan 100
[SwitchB-Ten-GigabitEthernet1/0/3] quit
```

   # Enable LLDP globally.

```
[SwitchB] lldp global enable
```

   # Enable CN TLV advertising on Ten-GigabitEthernet 1/0/1 and Ten-GigabitEthernet 1/0/2.

```
[SwitchB] interface ten-gigabitethernet 1/0/1
[SwitchB-Ten-GigabitEthernet1/0/1] lldp tlv-enable dot1-tlv congestion-notification
[SwitchB-Ten-GigabitEthernet1/0/1] quit
[SwitchB] interface ten-gigabitethernet 1/0/3
[SwitchB-Ten-GigabitEthernet1/0/3] lldp tlv-enable dot1-tlv congestion-notification
[SwitchB-Ten-GigabitEthernet1/0/3] quit
```

   # Enable QCN globally.

```
[SwitchB] qcn enable
```

   # Assign the switch to the CND with CNPV 1.

```
[SwitchB] qcn priority 1 auto
```

   # Configure the CND defense mode **edge** and alternate value 0 for interface Ten-GigabitEthernet 1/0/2.

```
[SwitchB-Ten-GigabitEthernet1/0/2] qcn port priority 1 admin defense-mode edge
alternate 0
[SwitchB-Ten-GigabitEthernet1/0/2] quit
```

   # Assign the switch to the CND with CNPV 1, and configure all interfaces to negotiate the defense mode and alternate priority by using LLDP.

```
[SwitchB] qcn priority 1 auto
```

3. Configure Switch C in the same way Switch A is configured. (Details not shown.)

Verifying the configuration

# Display the CND settings for interfaces on Switch A.

```
[SwitchA] display qcn interface
Interface: Ten-GigabitEthernet1/0/1
 CNPV  Mode   Defense-mode     Alternate
---------------------------------------------------
 1     comp   interior         0

Interface: Ten-GigabitEthernet1/0/2
 CNPV  Mode   Defense-mode     Alternate
---------------------------------------------------
 1     comp   interior         0
```

# Display the CND settings for interfaces on Switch B.

```
[SwitchB] display qcn interface
Interface: Ten-GigabitEthernet1/0/1
 CNPV  Mode   Defense-mode     Alternate
---------------------------------------------------
 1     comp   interior         0

Interface: Ten-GigabitEthernet1/0/2
 CNPV  Mode   Defense-mode     Alternate
---------------------------------------------------
 1     admin  edge             0

Interface: Ten-GigabitEthernet1/0/3
 CNPV  Mode   Defense-mode     Alternate
---------------------------------------------------
 1     comp   interior         0
```

# Display the CND settings for interfaces on Switch C.

```
[SwitchC] display qcn interface
Interface: Ten-GigabitEthernet1/0/1
 CNPV  Mode   Defense-mode     Alternate
---------------------------------------------------
 1     comp   interior         0

Interface: Ten-GigabitEthernet1/0/2
 CNPV  Mode   Defense-mode     Alternate
---------------------------------------------------
 1     comp   interior         0
```

MultiCND QCN configuration example

Network requirements

As shown in Figure 35:

- RP 1 and RP 2 are in the same VLAN.
- RP 3 and RP 4 are in the same VLAN.
- RP 1, RP 2, Switch A, Switch B, and Switch C form a CND with CNPV 1.
- RP 3, RP 4, Switch C, Switch D, and Switch E form a CND with CNPV 5.

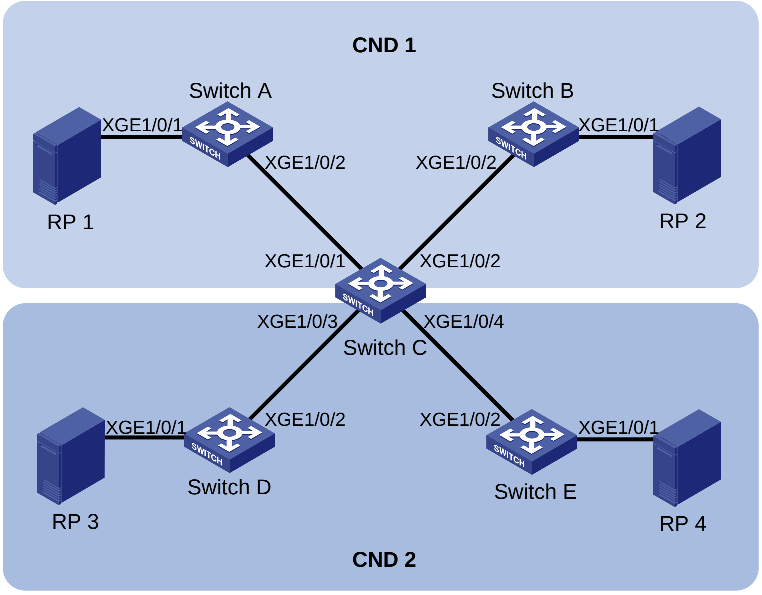
Configure QCN for CNPV 1 to meet the following requirements:

- Switch A, Switch B, and Switch C detect congestion for traffic with 802.1p priority 1.
- Switch A and Switch B do not detect congestion for traffic with 802.1p priority 5.

Configure QCN for CNPV 5 to meet the following requirements:

- Switch C, Switch D, and Switch E detect congestion for traffic with 802.1p priority 5.
- Switch D and Switch E do not detect congestion for traffic with 802.1p priority 1.

Figure 35 Network diagram



Configuration procedure

1. Configure Switch A:

# Create VLAN 100, and assign Ten-GigabitEthernet 1/0/1 to the VLAN.

```
<SwitchA> system-view
[SwitchA] vlan 100
[SwitchA-vlan100] port ten-gigabitethernet 1/0/1
[SwitchA-vlan100] quit
```

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

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

# Enable LLDP globally.

```
[SwitchA] lldp global enable
```

# Enable CN TLV advertising on Ten-GigabitEthernet 1/0/1.

```
[SwitchA] interface ten-gigabitethernet 1/0/1
[SwitchA-Ten-GigabitEthernet1/0/1] lldp tlv-enable dot1-tlv congestion-notification
[SwitchA-Ten-GigabitEthernet1/0/1] quit
```

# Enable CN TLV advertising on Ten-GigabitEthernet 1/0/2.

```
[SwitchA] interface ten-gigabitethernet 1/0/2
[SwitchA-Ten-GigabitEthernet1/0/2] lldp tlv-enable dot1-tlv congestion-notification
```

```
[SwitchA-Ten-GigabitEthernet1/0/2] quit
```

# Enable QCN globally.

```
[SwitchA] qcn enable
```

# Assign the switch to the CND with CNPV 1, and configure all interfaces to negotiate the defense mode and alternate priority by using LLDP.

```
[SwitchA] qcn priority 1 auto
```

2. Configure Switch B in the same way Switch A is configured. (Details not shown.)
3. Configure Switch C:

# Create VLAN 100 and VLAN 200.

```
<SwitchC> system-view
[SwitchC] vlan 100
[SwitchC-vlan100] quit
[SwitchC] vlan 200
[SwitchC-vlan200] quit
```

# Configure the following interfaces as trunk ports, and assign all of them to VLAN 100 and VLAN 200:

- Ten-GigabitEthernet 1/0/1.
- Ten-GigabitEthernet 1/0/2.
- Ten-GigabitEthernet 1/0/3.
- Ten-GigabitEthernet 1/0/4.

```
[SwitchC] interface ten-gigabitethernet 1/0/1
[SwitchC-Ten-GigabitEthernet1/0/1] port link-type trunk
[SwitchC-Ten-GigabitEthernet1/0/1] port trunk permit vlan 100 200
[SwitchC-Ten-GigabitEthernet1/0/1] quit
[SwitchC] interface ten-gigabitethernet 1/0/2
[SwitchC-Ten-GigabitEthernet1/0/2] port link-type trunk
[SwitchC-Ten-GigabitEthernet1/0/2] port trunk permit vlan 100 200
[SwitchC-Ten-GigabitEthernet1/0/2] quit
[SwitchC] interface ten-gigabitethernet 1/0/3
[SwitchC-Ten-GigabitEthernet1/0/3] port link-type trunk
[SwitchC-Ten-GigabitEthernet1/0/3] port trunk permit vlan 100 200
[SwitchC-Ten-GigabitEthernet1/0/3] quit
[SwitchC] interface ten-gigabitethernet 1/0/4
[SwitchC-Ten-GigabitEthernet1/0/4] port link-type trunk
[SwitchC-Ten-GigabitEthernet1/0/4] port trunk permit vlan 100 200
[SwitchC-Ten-GigabitEthernet1/0/4] quit
```

# Enable LLDP globally.

```
[SwitchC] lldp global enable
```

# Enable CN TLV advertising on the following interfaces:

- Ten-GigabitEthernet 1/0/1.
- Ten-GigabitEthernet 1/0/2.
- Ten-GigabitEthernet 1/0/3.
- Ten-GigabitEthernet 1/0/4.

```
[SwitchC] interface ten-gigabitethernet 1/0/1
[SwitchC-Ten-GigabitEthernet1/0/1] lldp tlv-enable dot1-tlv congestion-notification
```

```
[SwitchC-Ten-GigabitEthernet1/0/1] quit
[SwitchC] interface ten-gigabitethernet 1/0/2
[SwitchC-Ten-GigabitEthernet1/0/2] lldp tlv-enable dot1-tlv congestion-notification
[SwitchC-Ten-GigabitEthernet1/0/2] quit
[SwitchC] interface ten-gigabitethernet 1/0/3
[SwitchC-Ten-GigabitEthernet1/0/3] lldp tlv-enable dot1-tlv congestion-notification
[SwitchC-Ten-GigabitEthernet1/0/3] quit
[SwitchC] interface ten-gigabitethernet 1/0/4
[SwitchC-Ten-GigabitEthernet1/0/4] lldp tlv-enable dot1-tlv congestion-notification
[SwitchC-Ten-GigabitEthernet1/0/4] quit
```

# Enable QCN globally.

```
[SwitchC] qcn enable
```

# Assign the switch to the CNDs with CNPV 1 and CNPV 5.

```
[SwitchC] qcn priority 1 auto
[SwitchC] qcn priority 5 admin defense-mode interior-ready alternate 4
```

# Configure the CND defense mode **edge** and alternate value 4 for Ten-GigabitEthernet 1/0/1 and Ten-GigabitEthernet 1/0/2.

```
[SwitchC] interface ten-gigabitethernet 1/0/1
[SwitchC-Ten-GigabitEthernet1/0/1] qcn port priority 5 admin defense-mode edge
alternate 4
[SwitchC-Ten-GigabitEthernet1/0/1] quit
[SwitchC] interface ten-gigabitethernet 1/0/2
[SwitchC-Ten-GigabitEthernet1/0/2] qcn port priority 5 admin defense-mode edge
alternate 4
[SwitchC-Ten-GigabitEthernet1/0/2] quit
```

# Assign the switch to the CND with CNPV 1, and configure all interfaces to negotiate the defense mode and alternate priority by using LLDP.

```
[SwitchC] qcn priority 1 auto
```

4. Configure Switch D:

   # Create VLAN 200, and assign Ten-GigabitEthernet 1/0/1 to the VLAN.

   ```
   <SwitchD> system-view
   [SwitchD] vlan 200
   [SwitchD-vlan200] port ten-gigabitethernet 1/0/1
   [SwitchD-vlan200] quit
   ```

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

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

   # Enable QCN globally.

   ```
   [SwitchD] qcn enable
   ```

   # Assign the switch to the CND with CNPV 5.

   ```
   [SwitchD] qcn priority 5 admin defense-mode interior-ready alternate 4
   ```

5. Configure Switch E in the same way Switch D is configured. (Details not shown.)

Verifying the configuration

# Display the CND settings for interfaces on Switch A.

```
Interface: Ten-GigabitEthernet1/0/1
 CNPV  Mode   Defense-mode     Alternate
---------------------------------------------------
 1     comp   interior         0

Interface: Ten-GigabitEthernet1/0/2
 CNPV  Mode   Defense-mode     Alternate
---------------------------------------------------
 1     comp   interior         0
```

# Display the CND settings for interfaces on Switch B.

```
[SwitchB] display qcn interface
Interface: Ten-GigabitEthernet1/0/1
 CNPV  Mode   Defense-mode     Alternate
---------------------------------------------------
 1     comp   interior         0

Interface: Ten-GigabitEthernet1/0/2
 CNPV  Mode   Defense-mode     Alternate
---------------------------------------------------
 1     comp   interior         0
```

# Display the CND settings for interfaces on Switch C.

```
[SwitchC] display qcn interface
Interface: Ten-GigabitEthernet1/0/1
 CNPV  Mode   Defense-mode     Alternate
---------------------------------------------------
 1     comp   interior-ready   0
 5     admin  edge             4

Interface: Ten-GigabitEthernet1/0/2
 CNPV  Mode   Defense-mode     Alternate
---------------------------------------------------
 1     comp   interior-ready   0
 5     admin  edge             4

Interface: Ten-GigabitEthernet1/0/3
 CNPV  Mode   Defense-mode     Alternate
---------------------------------------------------
 1     comp   edge             0
 5     comp   interior-ready   4

Interface: Ten-GigabitEthernet1/0/4
 CNPV  Mode   Defense-mode     Alternate
---------------------------------------------------
 1     comp   edge             0
 5     comp   interior-ready   4
```

# Display the CND settings for interfaces on Switch D.

```
[SwitchD] display qcn interface
```

```
Interface: Ten-GigabitEthernet1/0/1
 CNPV  Mode   Defense-mode     Alternate
---------------------------------------------------
 5     comp   interior-ready   4

Interface: Ten-GigabitEthernet1/0/2
 CNPV  Mode   Defense-mode     Alternate
---------------------------------------------------
 5     comp   interior-ready   4
```

# Display the CND settings for interfaces on Switch E.

```
[SwitchE] display qcn interface
Interface: Ten-GigabitEthernet1/0/1
 CNPV  Mode   Defense-mode     Alternate
---------------------------------------------------
 5     comp   interior-ready   4

Interface: Ten-GigabitEthernet1/0/2
 CNPV  Mode   Defense-mode     Alternate
---------------------------------------------------
 5     comp   interior-ready   4
```

Support and other resources

Contacting HP

For worldwide technical support information, see the HP support website:

http://www.hp.com/support

Before contacting HP, collect the following information:

- Product model names and numbers
- Technical support registration number (if applicable)
- Product serial numbers
- Error messages
- Operating system type and revision level
- Detailed questions

Subscription service

HP recommends that you register your product at the Subscriber's Choice for Business website:

http://www.hp.com/go/wwalerts

After registering, you will receive email notification of product enhancements, new driver versions, firmware updates, and other product resources.

Related information

Documents

To find related documents, browse to the Manuals page of the HP Business Support Center website:

http://www.hp.com/support/manuals

- For related documentation, navigate to the Networking section, and select a networking category.
- For a complete list of acronyms and their definitions, see *HP FlexNetwork Technology Acronyms*.

Websites

- HP.com http://www.hp.com
- HP Networking http://www.hp.com/go/networking
- HP manuals http://www.hp.com/support/manuals
- HP download drivers and software http://www.hp.com/support/downloads
- HP software depot http://www.software.hp.com
- HP Education http://www.hp.com/learn

Conventions

This section describes the conventions used in this documentation set.

Command conventions

Convention	Description
Boldface	**Bold** text represents commands and keywords that you enter literally as shown.
Italic	*Italic* text represents arguments that you replace with actual values.
[ ]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x | y | ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x | y | ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x | y | ... } *	Asterisk-marked braces enclose a set of required syntax choices separated by vertical bars, from which you select at least one.
[x | y | ...] *	Asterisk-marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.

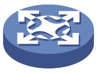
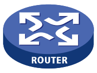
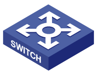
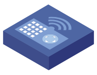
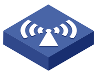
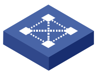
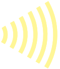
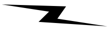
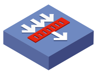
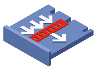
GUI conventions

Convention	Description
Boldface	Window names, button names, field names, and menu items are in bold text. For example, the **New User** window appears; click **OK**.
>	Multi-level menus are separated by angle brackets. For example, **File** > **Create** > **Folder**.

Symbols

Convention	Description
⚠ WARNING	An alert that calls attention to important information that if not understood or followed can result in personal injury.
△ CAUTION	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
ⓘ IMPORTANT	An alert that calls attention to essential information.
NOTE	An alert that contains additional or supplementary information.
💡 TIP	An alert that provides helpful information.

Network topology icons

	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the switching engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a mesh access point.
	Represents omnidirectional signals.
	Represents directional signals.
	Represents a security product, such as a firewall, UTM, multiservice security gateway, or load-balancing device.
	Represents a security card, such as a firewall, load-balancing, NetStream, SSL VPN, IPS, or ACG card.

Port numbering in examples

The port numbers in this document are for illustration only and might be unavailable on your device.

Index