Hewlett Packard Enterprise

HPE FlexFabric 12900E & 12900

ACL and QoS

Configuration Guide

Part number: 5998-8338R
Software version: Release 1135 and later
Document version: 6W102-20151124

Acknowledgments

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 varies with 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, the ACL is applied to hardware to process traffic. Example modules are packet filter and QoS.
- If the module is implemented in software, the ACL is applied to software to process traffic. Example modules are routing and login management.

The login management module denies packets that do not match any ACL. Some modules (QoS for example) ignore the action in the matching ACL rule, and they do not base their drop or forwarding decisions on the ACL rules. For information about how a module uses ACLs, see the configuration guide or command reference for the module.

ACL categories

Category	ACL number	IP version	Match criteria
Basic ACLs	2000 to 2999	IPv4	Source IPv4 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.
Ethernet frame header ACLs	4000 to 4999	IPv4	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	User specified match 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.

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.** VPN instance. **2.** More 0s in the source IPv4 address wildcard (more 0s means a narrower IPv4 address range). **3.** Rule configured earlier.
IPv4 advanced ACL	**1.** VPN instance. **2.** Specific protocol number. **3.** More 0s in the source IPv4 address wildcard mask. **4.** More 0s in the destination IPv4 address wildcard. **5.** Narrower TCP/UDP service port number range. **6.** 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 a 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.

Configuration task list

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

Step	Command	Remarks
5. Create or edit a rule.	**rule** [*rule-id*] { **deny** | **permit** } [**counting** | **fragment** | **source** { *source-address source-wildcard* | **any** } | **time-range** *time-range-name* | **vpn-instance** *vpn-instance-name*] *	By default, an IPv4 basic ACL does not contain any rule. If an IPv4 basic ACL is for QoS traffic classification or packet filtering: • Do not specify the **vpn-instance** *vpn-instance-name* option. • Do not specify the **counting** keyword if the ACL is for outbound application. • The ACL takes effect only on packets forwarded at Layer 3 if it is for outbound application.
6. (Optional.) Add or edit a rule comment.	**rule** *rule-id* **comment** *text*	By default, no rule comments are configured.

Configuring an IPv4 advanced ACL

IPv4 advanced ACLs match packets based on source IP addresses, destination IP addresses, packet priorities, protocol numbers, and 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.

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-wildcard* | **any** } | **destination-port** *operator port1* [*port2*] | { **dscp** *dscp* | { **precedence** *precedence* | **tos** *tos* } * } | **fragment** | **icmp-type** { *icmp-type* [*icmp-code*] | *icmp-message* } | **qos-local-id** *local-id-value* | **source** { *source-address source-wildcard* | **any** } | **source-port** *operator port1* [*port2*] | **time-range** *time-range-name* | **vpn-instance** *vpn-instance-name*] *	By default, an IPv4 advanced ACL does not contain any rule. If an IPv4 advanced ACL is for QoS traffic classification or packet filtering: • Do not specify the **vpn-instance** *vpn-instance-name* option. • Do not specify **neq** for the *operator* argument. • Do not specify **gt**, **lt**, or **range** for the *operator* argument, nor specify the **counting** keyword, if the ACL is for outbound application. • The ACL takes effect only on packets forwarded at Layer 3 if it is for outbound application. The **qos-local-id** *local-id-value* option is available in Release 1138P01 and later versions.
6. (Optional.) Add or edit a rule comment.	**rule** *rule-id* **comment** *text*	By default, no rule comments are configured.

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), and link layer protocol type.

To configure an Ethernet frame header ACL:

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

Step	Command	Remarks
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. When an Ethernet frame header ACL is for QoS traffic classification or packet filtering: • With the **lsap** keyword specified , the *lsap-type* argument must be AAAA and the *lsap-type-mask* argument must be FFFF. Otherwise, the ACL cannot be applied successfully. • Do not specify the **lsap**, **type**, and **counting** keywords if the ACL is for outbound QoS traffic classification or packet filtering.
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.
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 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 packets. For example, you can apply an ACL to an interface to filter incoming or outgoing packets.

NOTE:

The ACL-based packet filter feature is available on Layer 2 Ethernet interfaces, Layer 3 Ethernet interfaces, Layer 3 Ethernet subinterfaces, and VLAN interfaces. The term "interface" in this section collectively refers to these types of interfaces. You can use the **port link-mode** command to configure an Ethernet port as a Layer 2 or Layer 3 interface (see *Layer 2—LAN Switching Configuration Guide*).

Applying an ACL to filter packets globally

IMPORTANT:

This feature is available in Release 1137 and later versions.

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Apply an ACL to all physical interfaces to filter packets.	**packet-filter** { *acl-number* | **name** *acl-name* } **global** { **inbound** | **outbound** } [**hardware-count**]	By default, physical interfaces do not filter packets.

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** { *acl-number* | **name** *acl-name* } { **inbound** | **outbound** } [**extension**] [**hardware-count**]	By default, an interface does not filter packets. If you specify the **extension** keyword, the TCAM resources will be used for packet filtering. This keyword is available in Release 1137 and later versions. You can apply IPv4 ACLs or Ethernet frame header ACLs to an interface for packet filtering. In one direction of an interface, you can use only one ACL of each type.

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

IMPORTANT:

The **global** keyword in the following commands is available in Release 1137 and later versions:

- **display packet-filter**.
- **display packet-filter statistics**.
- **display packet-filter verbose**.
- **reset packet-filter statistics**.

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

Task	Command
Display ACL configuration and match statistics.	**display acl** { *acl-number* | **all** | **name** *acl-name* }
Display whether an ACL has been successfully applied to an interface for packet filtering (in standalone mode).	**display packet-filter** { **global** [**inbound** | **outbound**] [**slot** *slot-number*] | **interface** [*interface-type interface-number*] [**inbound** | **outbound**] | **interface vlan-interface** *vlan-interface-number* [**inbound** | **outbound**] [**slot** *slot-number*] }

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

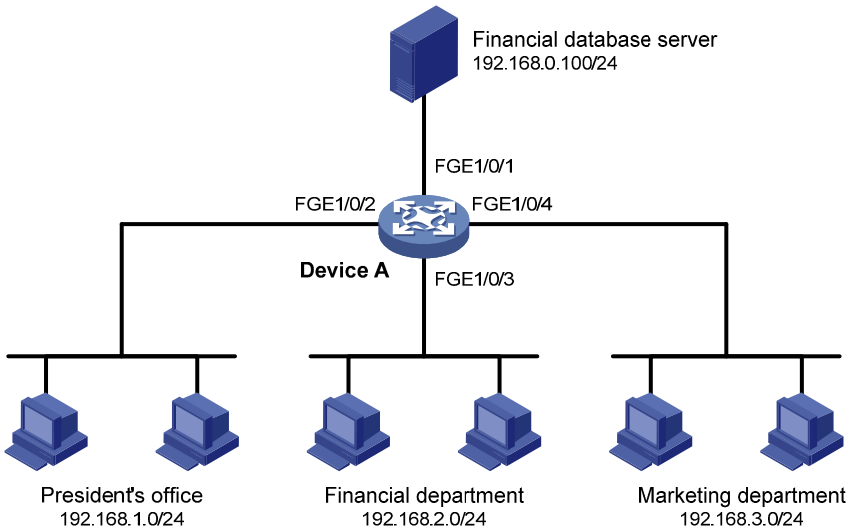
ACL configuration example

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 FortyGigE 1/0/1.

```
[DeviceA] interface fortygige 1/0/1
[DeviceA-FortyGigE1/0/1] packet-filter 3000 outbound
[DeviceA-FortyGigE1/0/1] 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

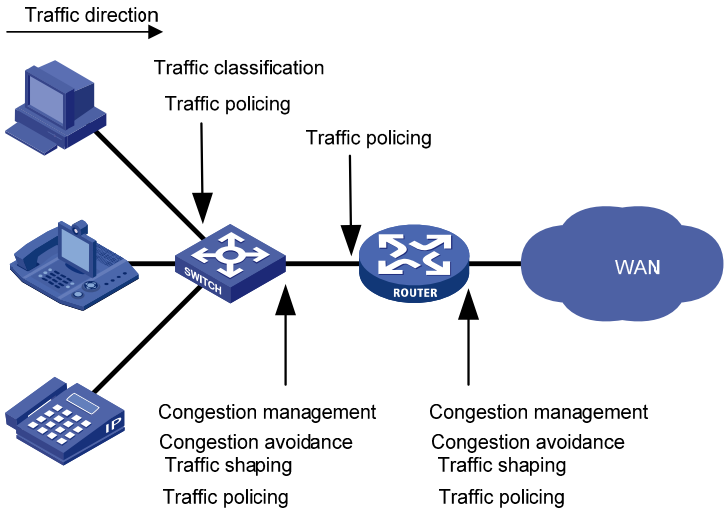
The differentiated service (DiffServ) model is a multiple-service model that can meet diverse QoS requirements. It is easy to implement and extend. DiffServ does not signal the network to reserve resources before sending data, as IntServ does.

QoS techniques overview

The QoS techniques include traffic classification, traffic policing, traffic shaping, rate limit, congestion management, and congestion avoidance. The following section briefly introduces these QoS techniques.

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

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 functions:

- **Traffic classification**—Uses certain 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.

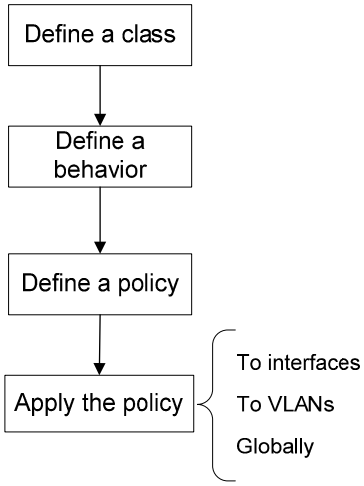
A traffic behavior defines a set of QoS actions to take on packets, such as priority marking and redirect.

By associating a traffic behavior with a traffic class in a QoS policy, you apply the specific set of QoS actions 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 restrictions and guidelines

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

- To configure multiple values for a match criterion, perform the following tasks:
  a. Set the logical operator to OR.
  b. Configure multiple **if-match** commands for the match criterion.
  For the **service-vlan-id** match criterion, 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

To define a traffic class:

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 { *acl-number* | **name** *acl-name* }	Matches an ACL. The value range for the *acl-number* argument is 2000 to 3999 for IPv4 ACLs, and 4000 to 4999 for Ethernet frame header 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.
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 3.
ip-precedence *ip-precedence-value*&<1-8>	Matches IP precedence. 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.

Option	Description
protocol *protocol-name*	Matches a protocol. The *protocol-name* argument can only be **ip**.
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.

Table 3 DSCP keywords and values

Keyword	DSCP value (binary)	DSCP value (decimal)
default	000000	0
af11	001010	10
af12	001100	12
af13	001110	14
af21	010010	18
af22	010100	20
af23	010110	22
af31	011010	26
af32	011100	28
af33	011110	30
af41	100010	34
af42	100100	36
af43	100110	38
cs1	001000	8
cs2	010000	16
cs3	011000	24
cs4	100000	32
cs5	101000	40
cs6	110000	48
cs7	111000	56
ef	101110	46

Defining a traffic behavior

A traffic behavior is a set of QoS actions (such as traffic filtering, shaping, policing, and priority marking) to take 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. The **mode dcbx** keyword specifies that a class-behavior association 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:

- **An interface**—The QoS policy takes effect on the traffic sent or received on the interface.
- **A 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.

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 as follows:

- Add rules to the ACL.
- Delete rules from the ACL.
- Modify rules of the ACL.

QoS policies applied to an interface, a VLAN, and globally are in descending order of priority. In other words, 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

The term "interface" in this section collectively refers to Layer 2 Ethernet interfaces, Layer 3 Ethernet interfaces, and Layer 3 Ethernet subinterfaces. You can use the **port link-mode** command to configure an Ethernet port as a Layer 2 or Layer 3 interface (see *Layer 2—LAN Switching 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 (IS-IS, BGP, and OSPF for example), RIP, 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 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. The switch does not support applying a QoS policy to the outbound direction of a Layer 3 Ethernet subinterface.

Applying the QoS policy to a VLAN

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

Configuration restrictions and guidelines

QoS policies cannot be applied to dynamic VLANs.

Configuration procedure

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 interfaces.

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.

Displaying and maintaining QoS policies

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

Task	Command
Display traffic class configuration (in standalone mode).	**display traffic classifier user-defined** [*classifier-name*] [**slot** *slot-number*]
Display traffic class configuration (in IRF mode).	**display traffic classifier user-defined** [*classifier-name*] [**chassis** *chassis-number* **slot** *slot-number*]
Display traffic behavior configuration (in standalone mode).	**display traffic behavior user-defined** [*behavior-name*] [**slot** *slot-number*]
Display traffic behavior configuration (in IRF mode).	**display traffic behavior user-defined** [*behavior-name*] [**chassis** *chassis-number* **slot** *slot-number*]
Display QoS and ACL resource usage (in standalone mode).	**display qos-acl resource** [**slot** *slot-number*]
Display QoS and ACL resource usage (in IRF mode).	**display qos-acl resource** [**chassis** *chassis-number* **slot** *slot-number*]
Display QoS policy configuration (in standalone mode).	**display qos policy user-defined** [*policy-name* [**classifier** *classifier-name*]] [**slot** *slot-number*]
Display QoS policy configuration (in IRF mode).	**display qos policy user-defined** [*policy-name* [**classifier** *classifier-name*]] [**chassis** *chassis-number* **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 (in standalone mode).	**display qos vlan-policy** { **name** *policy-name* | **vlan** *vlan-id* } [**slot** *slot-number*] [**inbound** | **outbound**]
Display information about QoS policies applied to VLANs (in IRF mode).	**display qos vlan-policy** { **name** *policy-name* | **vlan** [*vlan-id*] } [**chassis** *chassis-number* **slot** *slot-number*] [**inbound** | **outbound**]
Display information about QoS policies applied globally (in standalone mode).	**display qos policy global** [**slot** *slot-number*] [**inbound** | **outbound**]
Display information about QoS policies applied globally (in IRF mode).	**display qos policy global** [**chassis** *chassis-number* **slot** *slot-number*] [**inbound** | **outbound**]
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**]

Configuring priority mapping

Overview

When a packet arrives, depending on your configuration, a device assigns a set of QoS priority parameters to the packet based on either of the following:

- A certain priority field carried in the packet.
- 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 priorities such as 802.11e priority, 802.1p priority, DSCP, IP precedence, local precedence, and 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, IP precedence, and EXP. 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 device 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 device provides various types of priority maps. By looking through a priority map, the device 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 to only IP packets.
- **dscp-dp**—DSCP-to-drop priority mapping table, which is applicable to only IP packets.
- **dscp-dscp**—DSCP-to-DSCP priority mapping table, which is applicable to only IP packets that are forwarded at Layer 3 by the local switch.

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 4 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 without VLAN tags. The priority mapping results are the same as not trusting packet priority, as shown in Table 4.

- Using the DSCP carried in packets for priority mapping.

Table 5 Priority mapping results of trusting the DSCP (when the default dscp-dot1p and dot1p-lp priority mapping tables are used)

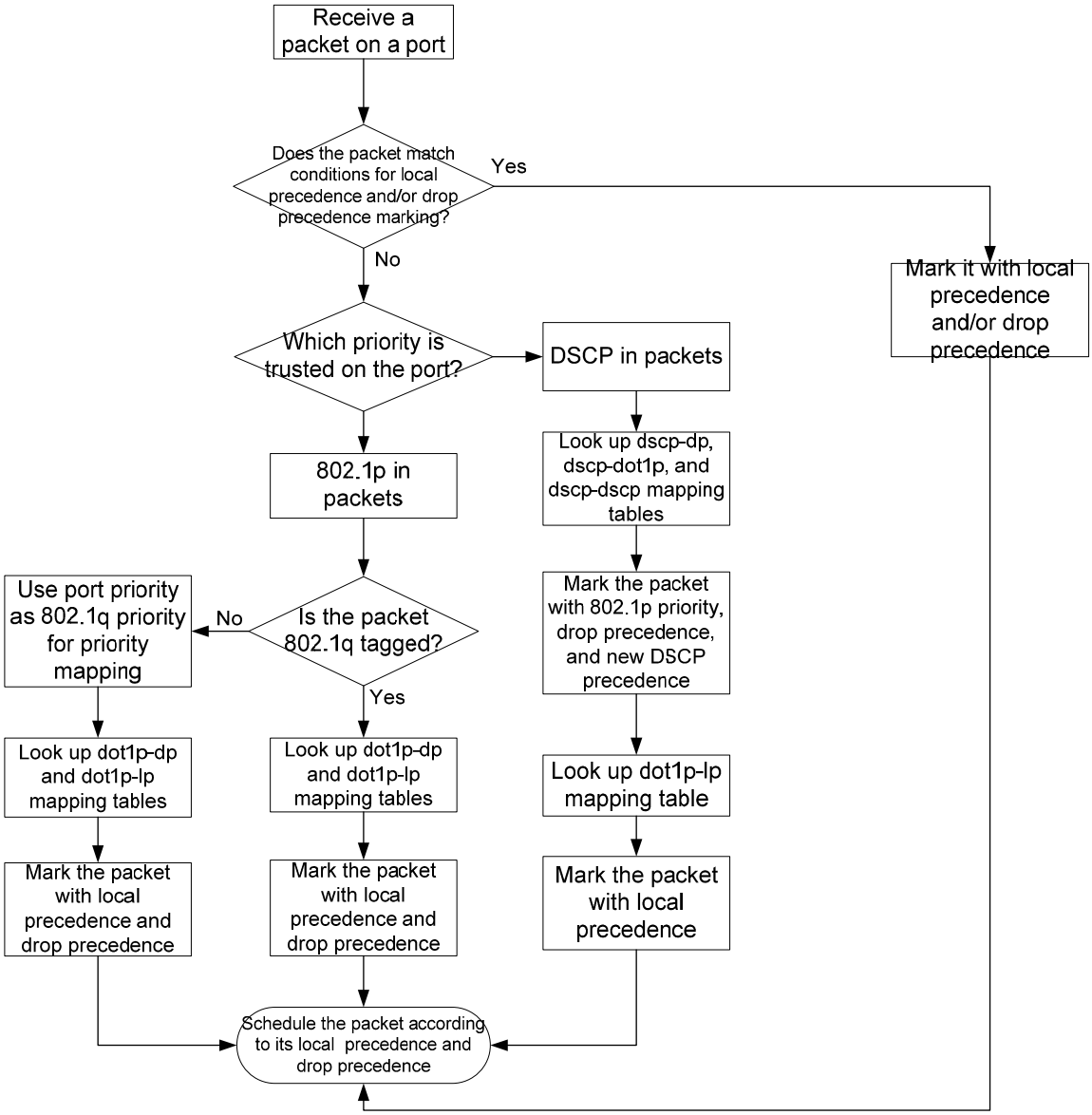
DSCP value carried in packets	Local precedence	Queue ID
0 to 7	2	2
8 to 15	0	0
16 to 23	1	1
24 to 31	3	3
32 to 39	4	4
40 to 47	5	5
48 to 55	6	6
56 to 63	7	7

The priority mapping procedure varies with the priority trust modes. 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 process 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



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

Configuring a priority map

The term "interface" in this section collectively refers to Layer 2 and Layer 3 Ethernet interfaces. You can use the **port link-mode** command to configure an Ethernet port as a Layer 2 or Layer 3 interface (see *Layer 2—LAN Switching Configuration Guide*).

To configure priority maps:

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 a port to trust packet priority for priority mapping

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

When you configure the 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.

To configure the trusted packet priority type on 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 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. **undo qos trust**	Use one of these commands. By default, the interface trusts the 802.1p priority.

Changing the port priority of an interface

If an interface does not trust any packet priorities, the device 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 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 mapping configuration examples

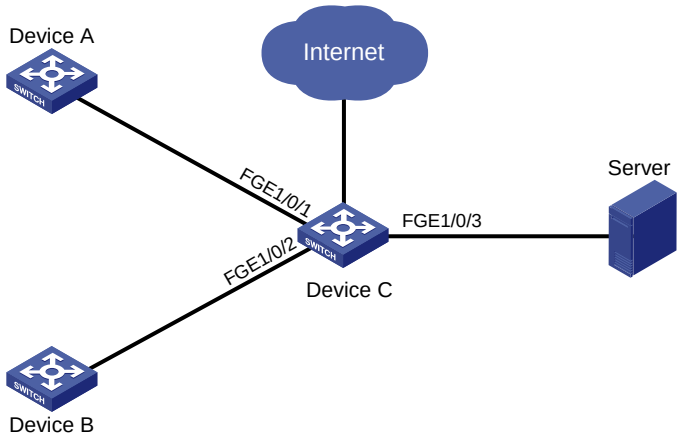
Priority trust mode configuration example

Network requirements

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

Configure Device C to preferentially process packets from Device A to Server when FortyGigE 1/0/3 of Device C is congested.

Figure 5 Network diagram



Configuration procedure

Assign port priorities to FortyGigE 1/0/1 and FortyGigE 1/0/2. Make sure the following:

- The priority of FortyGigE 1/0/1 is higher than that of FortyGigE 1/0/2.
- No trusted packet priority type is configured on FortyGigE 1/0/1 or FortyGigE 1/0/2.

```
<DeviceC> system-view
[DeviceC] interface FortyGigE 1/0/1
[DeviceC-FortyGigE1/0/1] qos priority 3
[DeviceC-FortyGigE1/0/1] quit
[DeviceC] interface FortyGigE 1/0/2
[DeviceC-FortyGigE1/0/2] qos priority 1
[DeviceC-FortyGigE1/0/2] quit
```

Priority mapping table and priority marking configuration example

Network requirements

As shown in Figure 6:

- The marketing department connects to FortyGigE1/0/1 of Device, which sets the 802.1p priority of traffic from the marketing department to 3.
- The R&D department connects to FortyGigE1/0/2 of Device, which sets the 802.1p priority of traffic from the R&D department to 4.
- The management department connects to FortyGigE1/0/3 of Device, which sets the 802.1p priority of traffic from the management department to 5.

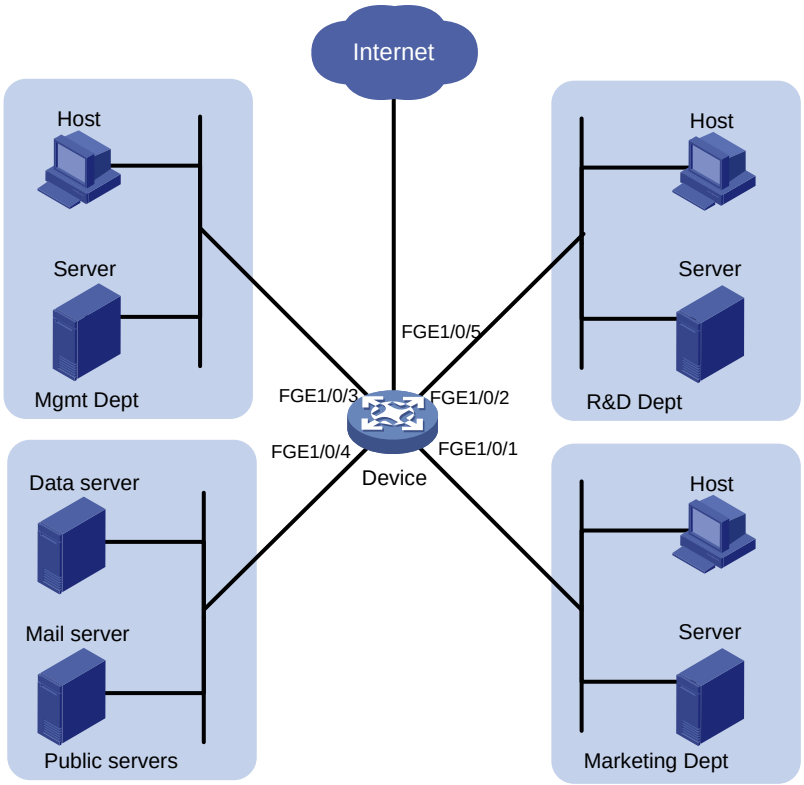
Configure port priority, 802.1p-to-local mapping table, and priority marking to implement the plan as described in Table 6.

Table 6 Configuration plan

Traffic destination	Traffic priority order	Queuing plan		
		Traffic source	Output queue	Queue priority
Public servers	R&D department >	R&D department	6	High

Traffic destination	Traffic priority order	Queuing plan		
		Traffic source	Output queue	Queue priority
	management department > marketing department	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 FortyGigE 1/0/1 to 3.

   ```
   <Device> system-view
   [Device] interface FortyGigE 1/0/1
   [Device-FortyGigE1/0/1] qos priority 3
   [Device-FortyGigE1/0/1] quit
   ```

   # Set the port priority of FortyGigE 1/0/2 to 4.

   ```
   [Device] interface FortyGigE 1/0/2
   [Device-FortyGigE1/0/2] qos priority 4
   [Device-FortyGigE1/0/2] quit
   ```

   # Set the port priority of FortyGigE 1/0/3 to 5.

```
[Device] interface FortyGigE 1/0/3
[Device-FortyGigE1/0/3] qos priority 5
[Device-FortyGigE1/0/3] 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.

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

3. Configure priority marking:

Create ACL 3000 to match HTTP traffic.

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

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

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

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

```
[Device] traffic behavior admin
[Device-behavior-admin] remark local-precedence 6
[Device-behavior-admin] quit
[Device] qos policy admin
[Device-qospolicy-admin] classifier http behavior admin
[Device-qospolicy-admin] quit
[Device] interface FortyGigE 1/0/3
[Device-FortyGigE1/0/3] qos apply policy admin inbound
```

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

```
[Device] traffic behavior market
[Device-behavior-market] remark local-precedence 4
[Device-behavior-market] quit
[Device] qos policy market
[Device-qospolicy-market] classifier http behavior market
[Device-qospolicy-market] quit
[Device] interface FortyGigE 1/0/1
[Device-FortyGigE1/0/1] qos apply policy market inbound
```

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

```
[Device] traffic behavior rd
[Device-behavior-rd] remark local-precedence 2
[Device-behavior-rd] quit
[Device] qos policy rd
[Device-qospolicy-rd] classifier http behavior rd
```

```
[Device-qospolicy-rd] quit
[Device] interface FortyGigE 1/0/2
[Device-FortyGigE1/0/2] 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 traffic conforms to the specification, and is called "conforming traffic." Otherwise, the traffic does not conform to the specification, and is 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. In each evaluation, if the number of tokens in the bucket is enough, the traffic conforms to the specification and the tokens for forwarding the packet are taken away. If the number of tokens in the bucket is not enough, the traffic is excessive.

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.

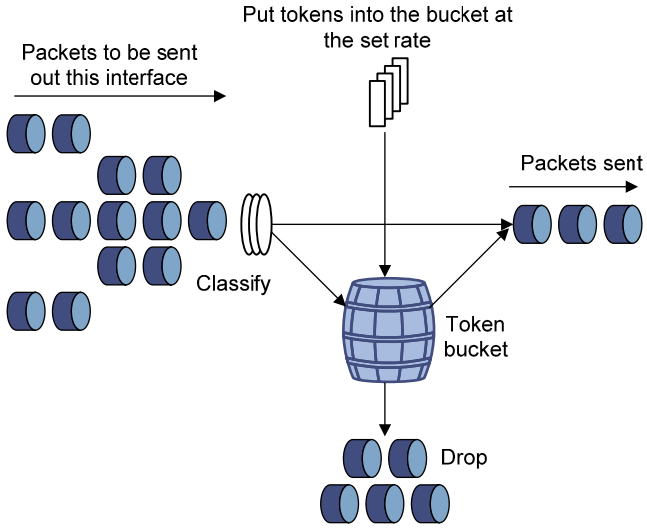
  When a packet arrives, the following rules apply:
  - If bucket C has enough tokens, the packet is colored green.
  - If bucket C does not have enough tokens but bucket E has enough tokens, the packet is colored yellow.
  - If neither bucket C nor bucket E has enough tokens, the packet is colored red.

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 certain traffic entering a network and limit it within a reasonable range. Another application is to discipline the excess traffic to prevent aggressive use of network resources by a certain application. For example, you can limit bandwidth for HTTP packets to less than 50% of the total. If the traffic of a certain 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



Traffic policing is widely used in policing traffic entering the networks of ISPs. It can classify the policed traffic and take pre-defined policing actions on each packet depending on the evaluation result:

- Forwarding the packet if the evaluation result is "conforming."
- Dropping the packet if the evaluation result is "excess."

The switch forwards green and yellow packets and drops red packets.

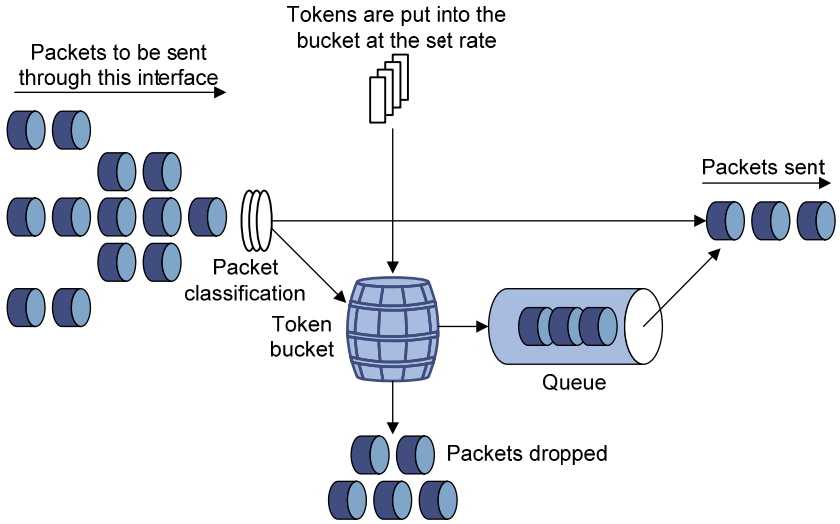
GTS

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:

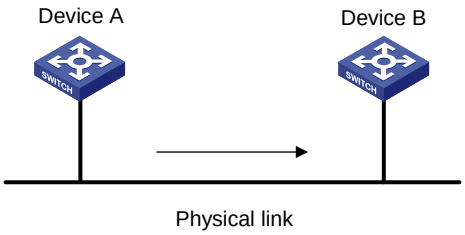
- Packets to be dropped with traffic policing are retained in a buffer or queue with GTS, as shown in Figure 8. When enough tokens are in the token bucket, the buffered packets are sent at an even rate.
- GTS can result in additional delay and traffic policing does not.

Figure 8 GTS



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

Figure 9 GTS application



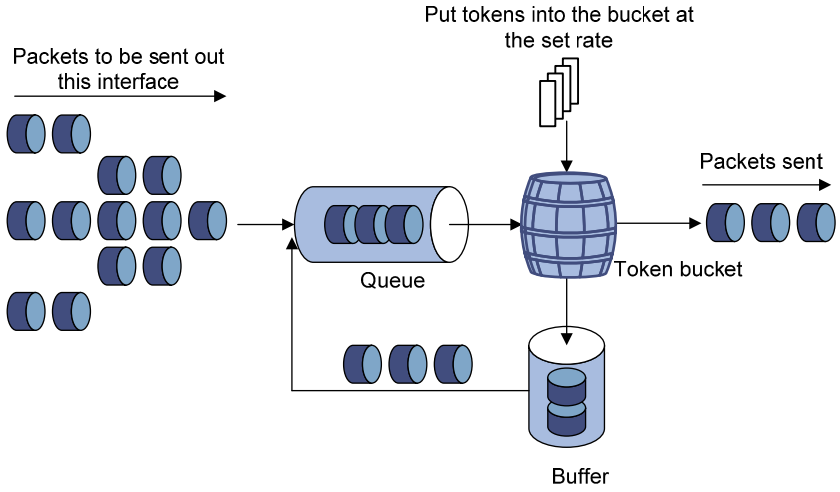
Rate limit

The switch supports rate limiting only outbound traffic.

The rate limit of a physical interface specifies the maximum rate for forwarding 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



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.
6. Configure a traffic policing action.	**car cir** *committed-information-rate* [**cbs** *committed-burst-size* [**ebs** *excess-burst-size*]] **car cir** *committed-information-rate* [**cbs** *committed-burst-size*] **pir** *peak-information-rate* [**ebs** *excess-burst-size*]	Use either of the commands. By default, no traffic policing action is configured.
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 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	Choose one of the application destinations as needed. By default, no QoS policy is applied. Traffic policing actions can be applied only to the inbound direction.
12. (Optional.) Display traffic policing configuration.	**display traffic behavior user-defined** [*behavior-name*]	Available in any view.

Configuring GTS

The term "interface" in this section collectively refers to Layer 2 and Layer 3 Ethernet interfaces. You can use the **port link-mode** command to configure an Ethernet port as a Layer 2 or Layer 3 interface (see *Layer 2—LAN Switching Configuration Guide*).

The switch supports configuring queue-based GTS. In queue-based GTS, you set GTS parameters for packets of a certain queue.

To configure GTS:

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

Configuring the rate limit

The term "interface" in this section collectively refers to Layer 2 and Layer 3 Ethernet interfaces. You can use the **port link-mode** command to configure an Ethernet port as a Layer 2 or Layer 3 interface (see *Layer 2—LAN Switching Configuration Guide*).

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

To configure the rate limit:

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

3.	Configure the rate limit for the interface.	**qos lr 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 (in standalone mode).	**display qos-acl resource** [**slot** *slot-number*]
Display QoS and ACL resource usage (in IRF mode).	**display qos-acl resource** [**chassis** *chassis-number* **slot** *slot-number*]
Display traffic behavior configuration.	**display traffic behavior user-defined** [*behavior-name*]
Display GTS configuration and statistics on an interface.	**display qos gts interface** [*interface-type interface-number*]
Display rate limit configuration and statistics on an interface.	**display qos lr interface** [*interface-type interface-number*]

Traffic policing and GTS configuration example

Network requirements

As shown in Figure 11, the server, Host A, and Host B can access the Internet through Device A and Device B.

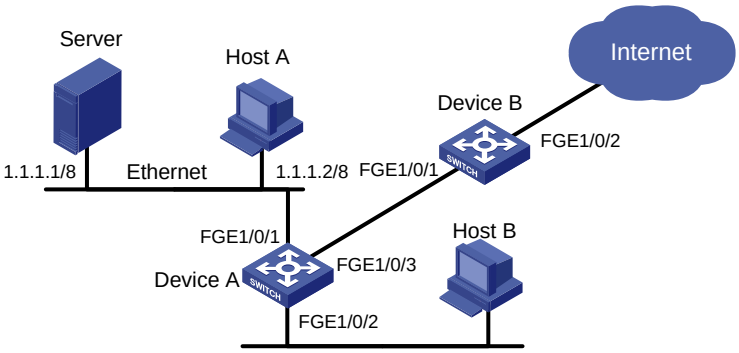
Perform traffic control on FortyGigE 1/0/1 of Device A for incoming traffic from the server and Host A to meet the following requirements:

- Limit the rate of traffic from the server to 102400 kbps: Forward the conforming traffic, and drop the excess traffic.
- Limit the rate of traffic from Host A to 25600 kbps: Forward the conforming traffic, and drop the excess traffic.

Perform traffic control on FortyGigE 1/0/1 and FortyGigE 1/0/2 of Device B to meet the following requirements:

- Limit the total incoming traffic rate of FortyGigE 1/0/1 to 204800 kbps and drop the excess traffic.
- Limit the outgoing HTTP traffic (traffic accessing the Internet) rate of FortyGigE 1/0/2 to 102400 kbps and drop the excess traffic.

Figure 11 Network diagram



Configuration procedures

1. Configure Device A:

Configure ACL 2001 and ACL 2002 to match traffic from Server and Host A, respectively.

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

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

Create a behavior named **server** and configure the CAR action (102400 kbps CIR) for the behavior.

```
[DeviceA] traffic behavior server
[DeviceA-behavior-server] car cir 102400
[DeviceA-behavior-server] quit
```

Create a behavior named **host** and configure the CAR action (25600 kbps CIR) for the behavior.

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

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

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

Apply QoS policy **car** to the incoming traffic of port FortyGigE 1/0/1.

```
[DeviceA] interface FortyGigE 1/0/1
[DeviceA-FortyGigE1/0/1] qos apply policy car inbound
```

2. Configure Device B:

Configure advanced ACL 3001 to match HTTP traffic.

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

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

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

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

```
[DeviceB] traffic classifier class
[DeviceB-classifier-class] if-match any
[DeviceB-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.

```
[DeviceB] traffic behavior car_inbound
[DeviceB-behavior-car_inbound] car cir 204800
[DeviceB-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.

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

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

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

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

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

Apply QoS policy **car_inbound** to the incoming traffic of port FortyGigE 1/0/1.

```
[DeviceB] interface FortyGigE 1/0/1
[DeviceB-FortyGigE1/0/1] qos apply policy car_inbound inbound
```

Apply QoS policy **car_outbound** to the outgoing traffic of port FortyGigE 1/0/2.

```
[DeviceB] interface FortyGigE 1/0/2
[DeviceB-FortyGigE1/0/2] qos apply policy car_outbound outbound
```

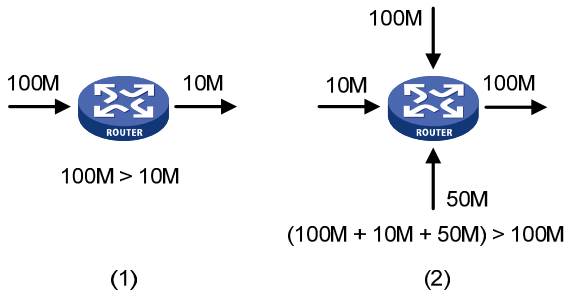
Configuring congestion management

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 brings 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 multi-user 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.

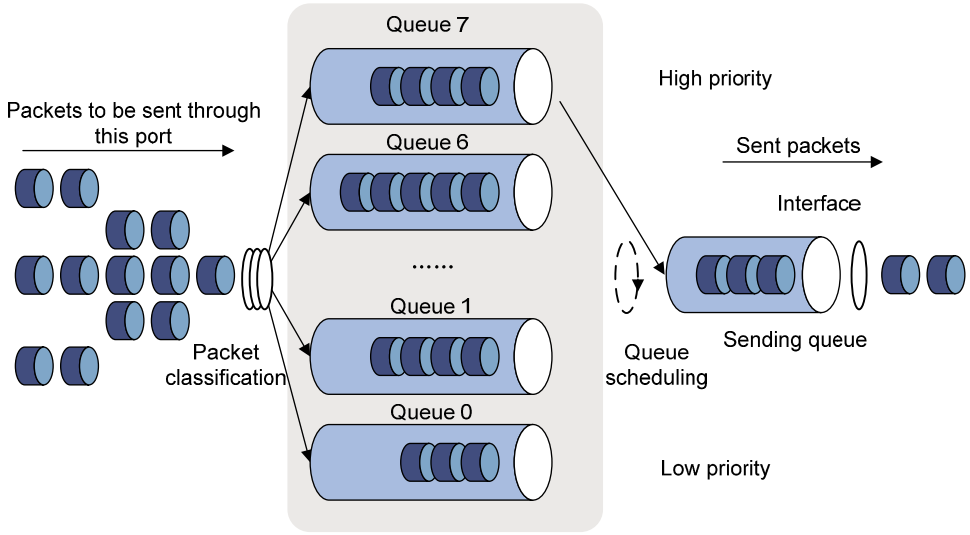
Congestion management uses queuing and scheduling algorithms to classify and sort traffic leaving a port.

Queue scheduling prioritizes packets to transmit high-priority packets preferentially. The switch supports Strict Priority (SP) queuing, Weighted Round Robin (WRR) queuing, Weighted Fair Queuing (WFQ), SP+WRR queuing, and SP+WFQ queuing.

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.

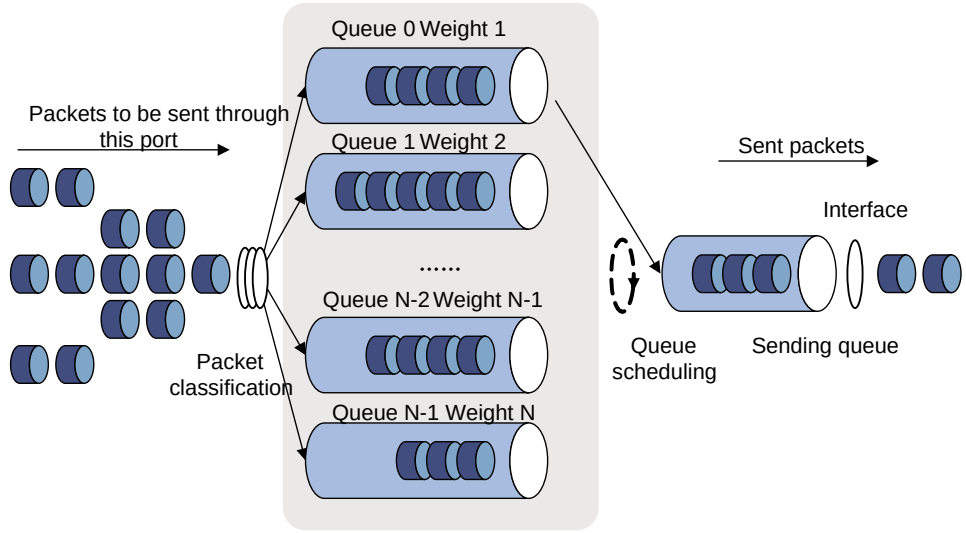
SP queuing schedules the eight queues in the descending order of priority. SP queuing sends packets in the queue with the highest priority first. When the queue with the highest priority is empty, it sends packets in the queue with the second highest priority, and so on. You can assign mission-critical packets to a high priority queue to make sure they are always served first. You can assign common service packets to the low priority queues, so that they are transmitted when the 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. In the worst case, lower priority traffic might never get serviced.

WRR queuing

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

Figure 14 WRR queuing

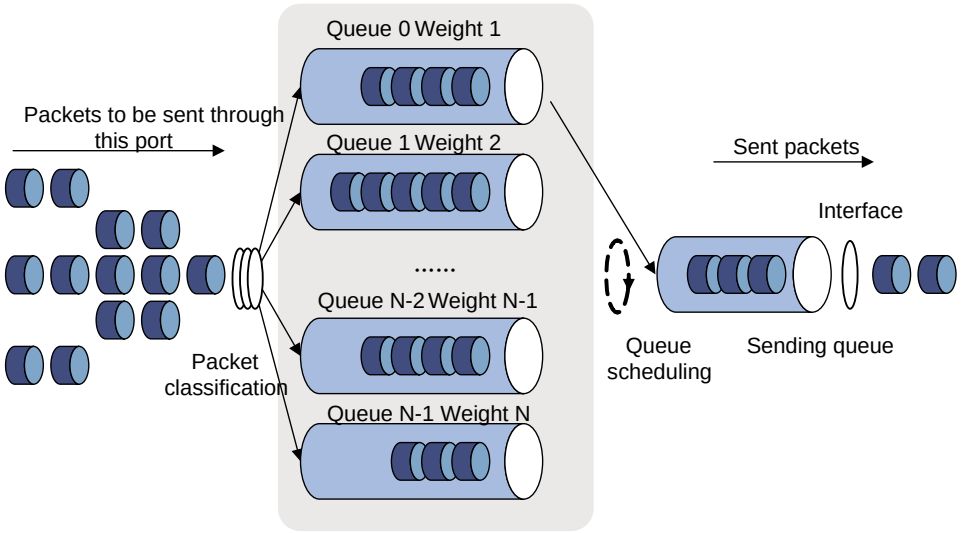


Assume a port provides eight output queues. WRR assigns each queue a weight value (represented by w7, w6, w5, w4, w3, w2, w1, or w0) to decide the proportion of resources assigned to the queue. On a 10 Gbps port, you can set w7, w6, w5, w4, w3, w2, w1, and w0 to 5, 5, 3, 3, 1, 1, 1, and 1, respectively. 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.

Another advantage of WRR queuing is that when the queues are scheduled in turn, the service time for each queue is not fixed. If a queue is empty, the next queue will be scheduled immediately. This maximizes bandwidth usage.

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.

SP+WRR queuing

You can implement SP+WRR queuing by configuring some queues on an interface to use SP queuing and others to use WRR queuing.

With this SP+WRR queuing method, the system schedules queues in the following order:

1. Schedules the queues in the SP group.
2. Schedules queues in the WRR group when all queues in the SP group are empty.

The queues in the SP group are scheduled based on their priorities. The queues in the WRR group are scheduled based on their weights.

SP+WFQ queuing

You can configure SP+WFQ queuing as follows:

- Assign some queues to the SP group.
- Assign others to the WFQ group.

Congestion management configuration task list

To configure hardware congestion management, you can use one of the following methods:

- Configure queue scheduling for each queue in interface view.
- Configure a queue scheduling profile, as described in "Configuring a queue scheduling profile."

Tasks at a glance
Perform one of the following tasks to configure congestion management on a per-port basis: • Configuring SP queuing • Configuring WRR queuing • Configuring WFQ queuing • Configuring SP+WRR queuing • Configuring SP+WFQ queuing
Configuring a queue scheduling profile

Configuring congestion management on a per-port basis

The term "interface" in this section collectively refers to Layer 2 and Layer 3 Ethernet interfaces. You can use the **port link-mode** command to configure an Ethernet port as a Layer 2 or Layer 3 interface (see *Layer 2—LAN Switching Configuration Guide*).

Configuring SP queuing

Configuration procedure

To configure SP queuing:

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

Configuration example

Configure FortyGigE 1/0/1 to use SP queuing:

Enter system view

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

Configure FortyGigE 1/0/1 to use SP queuing.

```
[Sysname] interface FortyGigE 1/0/1
[Sysname-FortyGigE1/0/1] qos sp
```

Configuring WRR queuing

Configuration procedure

To configure WRR queuing:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Enter interface view.	**interface** *interface-type interface-number*	N/A
3. Enable byte-count or packet-based WRR queuing.	**qos wrr** { **byte-count** | **weight** }	The default queuing algorithm on an interface is byte-count SP queuing.
4. Assign a queue to a WRR group, and configure scheduling parameters for the queue.	**qos wrr** *queue-id* **group 1** { **byte-count** | **weight** } *schedule-value*	Select **weight** or **byte-count** according to the WRR type (byte-count or packet-based) you have enabled. By default, the weights of queues 0 through 7 are 1, 2, 3, 4, 5, 6, 7, and 8, respectively.

Configuration example

1. Network requirements

Enable packet-based WRR on FortyGigE 1/0/1. Configure the weights of queues 0 through 7 as 1, 2, 4, 6, 1, 2, 4, and 6, respectively.

2. Configuration procedure

Enter system view.

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

Configure WRR queuing on FortyGigE 1/0/1.

```
[Sysname] interface FortyGigE 1/0/1
[Sysname-FortyGigE1/0/1] qos wrr weight
[Sysname-FortyGigE1/0/1] qos wrr 0 group 1 weight 1
[Sysname-FortyGigE1/0/1] qos wrr 1 group 1 weight 2
[Sysname-FortyGigE1/0/1] qos wrr 2 group 1 weight 4
[Sysname-FortyGigE1/0/1] qos wrr 3 group 1 weight 6
[Sysname-FortyGigE1/0/1] qos wrr 4 group 1 weight 1
[Sysname-FortyGigE1/0/1] qos wrr 5 group 1 weight 2
[Sysname-FortyGigE1/0/1] qos wrr 6 group 1 weight 4
[Sysname-FortyGigE1/0/1] qos wrr 7 group 1 weight 6
```

Configuring WFQ queuing

Configuration procedure

To configure WFQ queuing:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Enter 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 SP queuing.
4.	Assign a queue to a WFQ group, and configure scheduling parameters for the queue.	**qos wfq** *queue-id* **group 1** { **byte-count** | **weight** } *schedule-value*	Select **weight** or **byte-count** according to the WFQ type (byte-count or packet-based) you have enabled. By default, all queues have a weight of 1.
5.	(Optional.) Configure 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 WFQ queuing as follows:

- Configure byte-count WFQ queuing on interface FortyGigE 1/0/1.
- Configure the weights of queues 0 through 7 as 2, 5, 10, 10, 10, 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 FortyGigE 1/0/1.

```
[Sysname] interface FortyGigE 1/0/1
[Sysname-FortyGigE1/0/1] qos wfq byte-count
[Sysname-FortyGigE1/0/1] qos wfq 1 group 1 byte-count 2
[Sysname-FortyGigE1/0/1] qos wfq 3 group 1 byte-count 5
[Sysname-FortyGigE1/0/1] qos wfq 4 group 1 byte-count 10
[Sysname-FortyGigE1/0/1] qos wfq 5 group 1 byte-count 10
[Sysname-FortyGigE1/0/1] qos wfq 6 group 1 byte-count 10
[Sysname-FortyGigE1/0/1] qos wfq 0 group 1 byte-count 1
[Sysname-FortyGigE1/0/1] qos wfq 2 group 1 byte-count 2
[Sysname-FortyGigE1/0/1] qos wfq 7 group 1 byte-count 4
[Sysname-FortyGigE1/0/1] qos bandwidth queue 0 min 100000
[Sysname-FortyGigE1/0/1] qos bandwidth queue 1 min 100000
[Sysname-FortyGigE1/0/1] qos bandwidth queue 2 min 100000
[Sysname-FortyGigE1/0/1] qos bandwidth queue 3 min 100000
[Sysname-FortyGigE1/0/1] qos bandwidth queue 4 min 100000
[Sysname-FortyGigE1/0/1] qos bandwidth queue 5 min 100000
[Sysname-FortyGigE1/0/1] qos bandwidth queue 6 min 100000
[Sysname-FortyGigE1/0/1] 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

Step	Command	Remarks
2. Enter interface view or port group view.	**interface** *interface-type interface-number*	N/A
3. Enable byte-count or packet-based WRR queuing.	**qos wrr** { **byte-count** | **weight** }	The default queuing algorithm on an interface is SP queuing.
4. Assign a queue to the SP queue scheduling 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** { **weight** | **byte-count** } *schedule-value*	Select **weight** or **byte-count** according to the WRR type (byte-count or packet-based) you have enabled. By default, the weights of queues 0 through 7 are 1, 2, 3, 4, 5, 6, 7, and 8, respectively.

Configuration example

1. Network requirements

   Configure SP+WRR queuing as follows:

   - Configure SP+WRR queuing on FortyGigE 1/0/1, and use byte-count WRR.
   - Assign queues 0, 1, 2, and 3 on FortyGigE 1/0/1 to the SP group.
   - Assign queues 4, 5, 6, and 7 on FortyGigE 1/0/1 to the WRR group, with the weights being 1, 2, 1, and 3, respectively.

2. Configuration procedure

   # Enter system view.

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

   # Configure SP+WRR queuing on FortyGigE1/0/1.

   ```
   [Sysname] interface FortyGigE 1/0/1
   [Sysname-FortyGigE1/0/1] qos wrr byte-count
   [Sysname-FortyGigE1/0/1] qos wrr 0 group sp
   [Sysname-FortyGigE1/0/1] qos wrr 1 group sp
   [Sysname-FortyGigE1/0/1] qos wrr 2 group sp
   [Sysname-FortyGigE1/0/1] qos wrr 3 group sp
   [Sysname-FortyGigE1/0/1] qos wrr 4 group 1 byte-count 1
   [Sysname-FortyGigE1/0/1] qos wrr 5 group 1 byte-count 2
   [Sysname-FortyGigE1/0/1] qos wrr 6 group 1 byte-count 1
   [Sysname-FortyGigE1/0/1] qos wrr 7 group 1 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
2. Enter interface view or port group 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 SP queuing.

Step	Command	Remarks
4. Assign a queue to the SP queue scheduling 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** { **weight** | **byte-count** } *schedule-value*	Select **weight** or **byte-count** according to the WFQ type (byte-count or packet-based) you have enabled. By default, all queues have a weight of 1.
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 as follows:

- Configure SP+WFQ queuing on interface FortyGigE 1/0/1, and use packet-based WFQ.
- Assign queues 0, 1, 2, and 3 to the SP group.
- Assign queues 4, 5, 6, and 7 to the WFQ group, with the weights being 1, 2, 1, and 3, respectively.

2. Configuration procedure

Enter system view.

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

Configure SP+WFQ queuing on FortyGigE 1/0/1.

```
[Sysname] interface FortyGigE 1/0/1
[Sysname-FortyGigE1/0/1] qos wfq weight
[Sysname-FortyGigE1/0/1] qos wfq 0 group sp
[Sysname-FortyGigE1/0/1] qos wfq 1 group sp
[Sysname-FortyGigE1/0/1] qos wfq 2 group sp
[Sysname-FortyGigE1/0/1] qos wfq 3 group sp
[Sysname-FortyGigE1/0/1] qos wfq 4 group 1 weight 1
[Sysname-FortyGigE1/0/1] qos bandwidth queue 4 min 128000
[Sysname-FortyGigE1/0/1] qos wfq 5 group 1 weight 2
[Sysname-FortyGigE1/0/1] qos bandwidth queue 5 min 128000
[Sysname-FortyGigE1/0/1] qos wfq 6 group 1 weight 1
[Sysname-FortyGigE1/0/1] qos bandwidth queue 6 min 128000
[Sysname-FortyGigE1/0/1] qos wfq 7 group 1 weight 3
 [Sysname-FortyGigE1/0/1] qos bandwidth queue 7 min 128000
```

Displaying and maintaining congestion management

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*]

Task	Command
Display WFQ queuing configuration.	**display qos queue wfq interface** [*interface-type interface-number*]

Configuring a queue scheduling profile

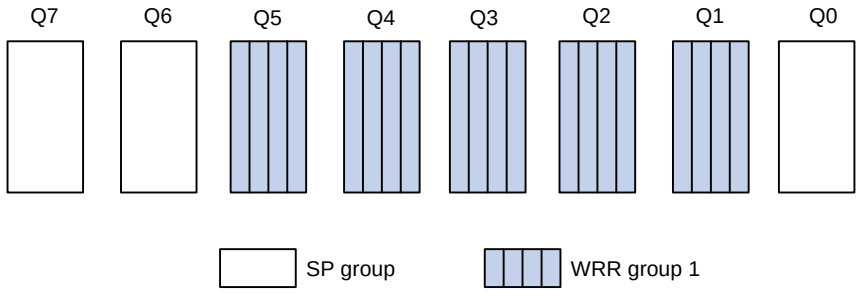
In a queue scheduling profile, you can configure scheduling parameters for each queue. By applying the queue scheduling profile to an interface, you can implement congestion management on the interface.

Queue scheduling profiles support three queue scheduling algorithms: 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.
- Queues in the WRR or WFQ group 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. Packets in queue 0 are sent when queue 7 and queue 6 are both empty.
- When queue 7, queue 6, and queue 0 are all empty, WRR group 1 is scheduled. Queue 1 through queue 5 in WRR group 1 are scheduled according to their weights.

Configuration procedure

To configure a queue scheduling profile, create the queue scheduling profile first, and then enter the queue scheduling profile view to configure its queue scheduling parameters. At last, apply the queue scheduling profile to the specified interface.

When you configure a queue scheduling profile, follow these guidelines:

- Only one queue scheduling profile can be applied to an interface.
- You can modify the scheduling parameters in a queue scheduling profile already applied to an interface.

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 1** { **byte-count** | **weight** } *schedule-value* • Configure a queue to use WFQ: **a.** **queue** *queue-id* **wfq group 1** { **byte-count** | **weight** } *schedule-value* **b.** **bandwidth queue** *queue-id* **min** *bandwidth-value*	You can configure the same queue scheduling algorithm, SP+WRR, or SP+WFQ for all queues. However, you cannot configure WRR+WFQ for queues. In a queue scheduling profile, you can configure different queue scheduling algorithms for different queues. By default, a queue scheduling profiles uses SP queuing for all queues. You can configure the minimum guaranteed bandwidth for only WFQ queues.
4. Return to system view.	**quit**	N/A
5. Enter interface view.	**interface** *interface-type interface-number*	N/A
6. Apply the queue scheduling profile to the interface.	**qos apply qmprofile** *profile-name*	The default queue scheduling profile of an interface depends on the device model.

Displaying and maintaining queue scheduling profiles

Execute **display** commands in any view.

Task	Command
Display the configuration of the specified or all queue scheduling profiles (in standalone mode).	**display qos qmprofile configuration** [*profile-name*] [**slot** *slot-number*]
Display the configuration of the specified or all queue scheduling profiles (in IRF mode).	**display qos qmprofile configuration** [*profile-name*] [**chassis** *chassis-number* **slot** *slot-number*]
Display the queue scheduling profiles already 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 FortyGigE 1/0/1 to meet the following requirements:

- Queue 7 has the highest priority, and its packets are sent preferentially.
- Queue 0 through queue 6 in the WRR group are scheduled according to their weights, which are 2, 4, 6, 8, 10, 12, and 14, respectively.

Configuration procedure

Enter system view.

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

Create queue scheduling profile **qm1**.

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

Configure queue 7 to use SP queuing.

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

Assign queue 0 through queue 6 to the WRR group, with their weights as 2, 4, 6, 8, 10, 12, and 14, respectively.

```
[Sysname-qmprofile-qm1] queue 0 wrr group 1 weight 2
[Sysname-qmprofile-qm1] queue 1 wrr group 1 weight 4
[Sysname-qmprofile-qm1] queue 2 wrr group 1 weight 6
[Sysname-qmprofile-qm1] queue 3 wrr group 1 weight 8
[Sysname-qmprofile-qm1] queue 4 wrr group 1 weight 10
[Sysname-qmprofile-qm1] queue 5 wrr group 1 weight 12
[Sysname-qmprofile-qm1] queue 6 wrr group 1 weight 14
[Sysname-qmprofile-qm1] quit
```

Apply queue scheduling profile **qm1** to interface FortyGigE 1/0/1.

```
[Sysname] interface FortyGigE 1/0/1
[Sysname-FortyGigE1/0/1] qos apply qmprofile qm1
```

After the configuration is completed, interface FortyGigE 1/0/1 performs queue scheduling as specified in queue scheduling profile **qm1**.

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 a specific 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	Choose one of the application destinations as needed. By default, no QoS policy is applied.
12. (Optional.) Display the traffic filtering configuration.	**display traffic behavior user-defined** [*behavior-name*]	Available in any view.

Traffic filtering configuration example

Network requirements

As shown in Figure 17, configure traffic filtering to filter the packets with source port not being 21, and received on FortyGigE 1/0/1.

Figure 17 Network diagram

Configuration procedure

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

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

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

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

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

```
[Device] traffic behavior behavior_1
[Device-behavior-behavior_1] filter deny
[Device-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.

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

Apply the QoS policy named **policy** to the incoming traffic of FortyGigE 1/0/1.

```
[Device] interface FortyGigE 1/0/1
[Device-FortyGigE1/0/1] 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 a DSCP value for a traffic class of IP packets to control the forwarding of these packets.

To configure priority marking, associate a traffic class with a traffic behavior configured with a priority marking action.

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

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* [**insert-before** *before-classifier-name*]	By default, no traffic behavior is configured.
6. Configure a priority marking action.	• Set the DSCP value for packets: **remark dscp** *dscp-value* • Set the local precedence for packets: **remark**[**green** | **red** | **yellow**] **local-precedence** *local-precedence-value*	Use one or more of the commands. By default, no priority marking action is configured. The **remark local-precedence** command applies to only the incoming traffic.
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.

Step	Command	Remarks
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	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.

Support for priority marking actions

When a priority marking QoS policy is applied to an interface, port group, VLAN, or globally, Table 7 shows the switch support for priority marking actions in the inbound and outbound directions.

Table 7 Support for priority marking actions

Action	Inbound	Outbound
DSCP marking	Supported	Not Supported
Local precedence marking	Supported	Not supported

NOTE:

DSCP marking takes effect only on Layer 3 packets.

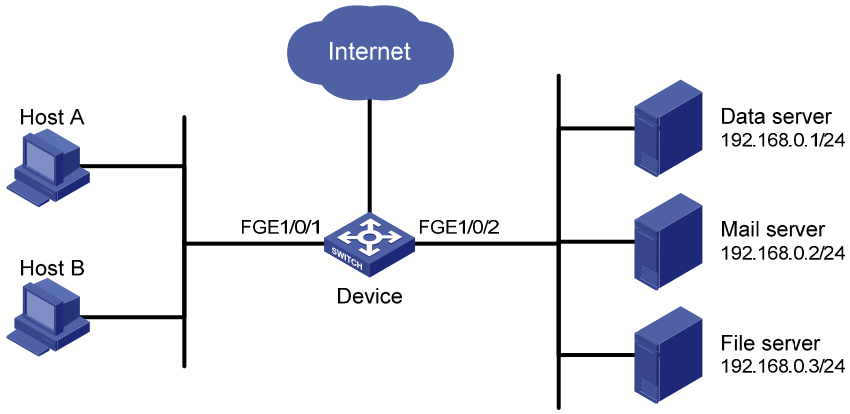
Priority marking configuration example

Network requirements

As shown in Figure 18, configure priority marking on the device 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.

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

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

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

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

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

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

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

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

```
[Device] traffic classifier classifier_mserver
[Device-classifier-classifier_mserver] if-match acl 3001
[Device-classifier-classifier_mserver] quit
```

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

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

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

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

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

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

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

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

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

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

Apply the QoS policy named **policy_server** to the incoming traffic of FortyGigE 1/0/1.

```
[Device] interface FortyGigE 1/0/1
[Device-FortyGigE1/0/1] qos apply policy policy_server inbound
[Device-FortyGigE1/0/1] quit
```

Configuring traffic redirecting

Traffic redirecting is the action of redirecting the packets matching the specific match criteria to a certain 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 to only 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
8. Create a QoS policy and enter QoS policy view.	**qos policy** *policy-name*	By default, no QoS policy exists.

Step	Command	Remarks
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	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.

Traffic redirecting configuration example

Network requirements

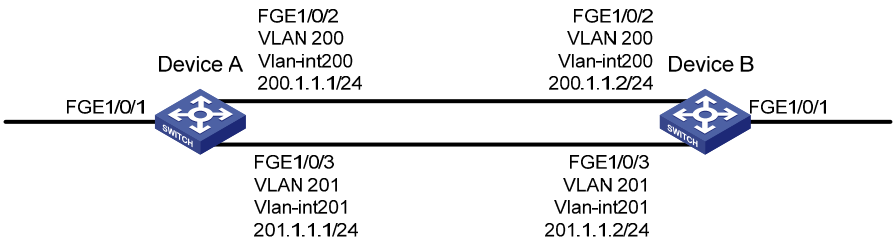
As shown in Figure 19:

- Device A is connected to Device B through two links. At the same time, Device A and Device B are each connected to other devices.
- FortyGigE 1/0/2 of Device A and FortyGigE 1/0/2 of Device B belong to VLAN 200.
- FortyGigE 1/0/3 of Device A and FortyGigE 1/0/3 of Device B belong to VLAN 201.
- On Device 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 Device 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 so that:

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

Figure 19 Network diagram



Configuration procedure

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

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

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

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

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

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

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

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

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

```
[DeviceA] traffic behavior behavior_1
[DeviceA-behavior-behavior_1] redirect interface FortyGigE 1/0/2
[DeviceA-behavior-behavior_1] quit
```

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

```
[DeviceA] traffic behavior behavior_2
[DeviceA-behavior-behavior_2] redirect interface FortyGigE 1/0/3
[DeviceA-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.

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

Apply the QoS policy named **policy** to the incoming traffic of FortyGigE 1/0/1.

```
[DeviceA] interface FortyGigE 1/0/1
[DeviceA-FortyGigE1/0/1] qos apply policy policy inbound
```

Configuring aggregate CAR

An aggregate CAR action is created globally. It 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

To configure aggregate CAR:

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*]] **qos car** *car-name* **aggregative cir** *committed-information-rate* [**cbs** *committed-burst-size*] **pir** *peak-information-rate* [**ebs** *excess-burst-size*]	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** *car-name*	By default, no aggregate CAR action is used.

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*]

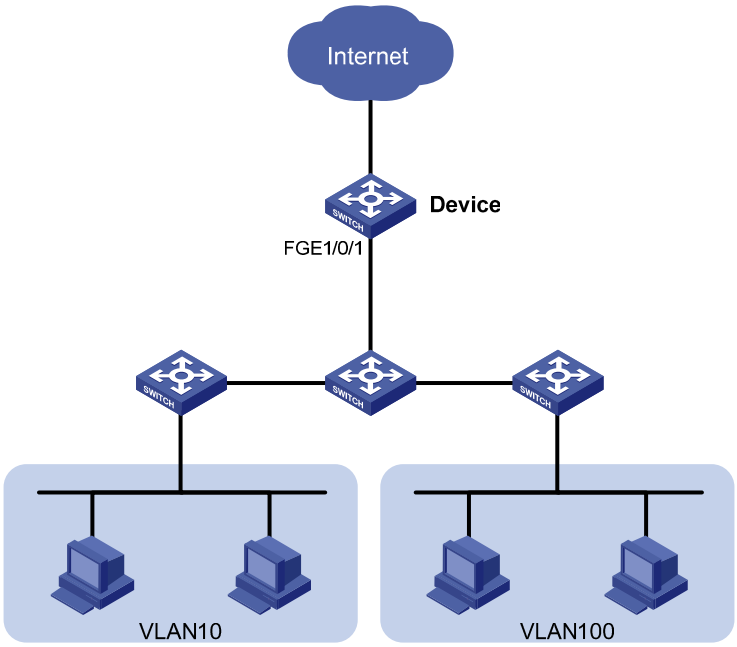
Aggregate CAR configuration example

Network requirements

As shown in Figure 20, configure an aggregate CAR to rate-limit the traffic of VLAN 10 and VLAN 100 received on FortyGigE 1/0/1 by using these parameters:

- The CIR is 2560 kbps.
- The CBS is 20480 bytes.

Figure 20 Network diagram



Configuration procedure

Configure an aggregate CAR according to the rate limit requirements.

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

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

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

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

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

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

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

Apply the QoS policy to the incoming traffic of FortyGigE 1/0/1.

```
[Device] interface FortyGigE 1/0/1
[Device-FortyGigE1/0/1]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	Choose one of the application destinations as needed. By default, no QoS policy is applied. Accounting actions can be applied only to the inbound direction.

Step	Command	Remarks

12. Display traffic accounting configuration.	• In standalone mode:
	  ○ **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**]
	• In IRF mode:
	  ○ **display qos policy global** [**chassis** *chassis-number* **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*] } [**chassis** *chassis-number* **slot** *slot-number*] [**inbound** | **outbound**]
		Available in any view.

Class-based accounting configuration example

Network requirements

As shown in Figure 21, configure class-based accounting to collect statistics for traffic sourced from 1.1.1.1/24 and received on FortyGigE 1/0/1.

Figure 21 Network diagram

Configuration procedure

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

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

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

```
[Device] traffic classifier classifier_1
```

```
[Device-classifier-classifier_1] if-match acl 2000
[Device-classifier-classifier_1] quit
```

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

```
[Device] traffic behavior behavior_1
[Device-behavior-behavior_1] accounting
[Device-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.

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

Apply the QoS policy named **policy** to the incoming traffic of FortyGigE 1/0/1.

```
[Device] interface FortyGigE 1/0/1
[Device-FortyGigE1/0/1] qos apply policy policy inbound
[Device-FortyGigE1/0/1] quit
```

Display traffic statistics to verify the configuration.

```
[Device] display qos policy interface FortyGigE 1/0/1

Interface: FortyGigE1/0/1

  Direction: Inbound

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

Configuring queue-based accounting

Queue-based accounting collects the following traffic statistics on a per-queue basis:

- The total length of a queue.
- The number of packets dropped by a queue.
- The current length of a queue.
- The ratio of the current length to the total length of a queue.

Configuration procedure

To configure queue-based accounting:

Step	Command	Description
1. Enter system view.	**system-view**	N/A
2. Set the packet statistics collection mode to queue.	**statistic mode queue**	The default setting is VSI.

Displaying and maintaining queue-based accounting

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

Task	Command
Display queue-based traffic statistics for interfaces.	**display qos queue-statistics interface** [*interface-type interface-number*] **outbound**
Display the packet statistics collection mode.	**display statistic mode**
Clear queue-based traffic statistics for interfaces.	**reset qos queue-statistics interface** [*interface-type interface-number*] **outbound**
Clear interface statistics (see *Layer 2—LAN Switching Command Reference*).	**reset counters interface** [*interface-type* [*interface-number*]]

Appendixes

Appendix A Default priority maps

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

Table 8 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

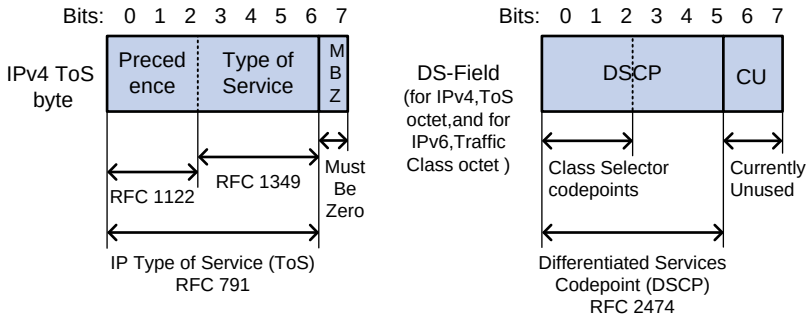
Table 9 Default dscp-dp and dscp-dot1p priority maps

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 22 ToS and DS fields



As shown in Figure 22, 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 10 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 11 DSCP values

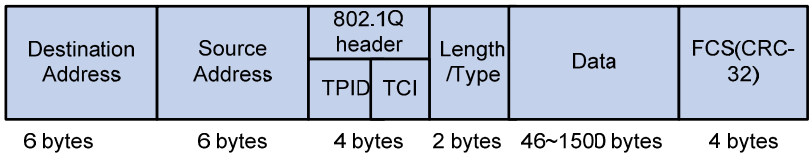
DSCP value (decimal)	DSCP value (binary)	Description
46	101110	ef
10	001010	af11
12	001100	af12
14	001110	af13
18	010010	af21
20	010100	af22
22	010110	af23
26	011010	af31
28	011100	af32

DSCP value (decimal)	DSCP value (binary)	Description
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 and applies to occasions where Layer 3 header analysis is not needed and QoS must be assured at Layer 2.

Figure 23 An Ethernet frame with an 802.1Q tag header



As shown in Figure 23, the four-byte 802.1Q tag header contains the TPID and the TCI fields. The value of the TPID is 0x8100. Figure 24 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 12 shows the values for 802.1p priority.

Figure 24 802.1Q tag header

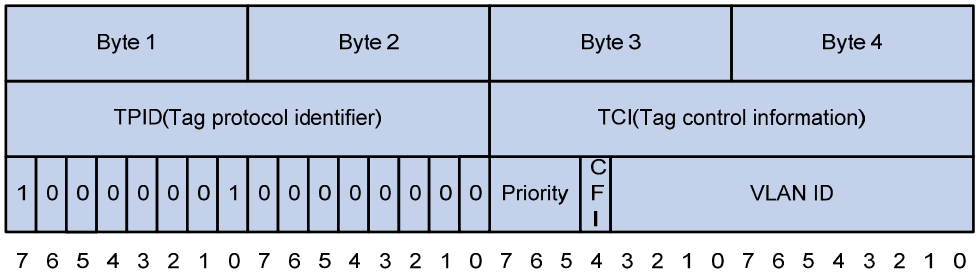


Table 12 Description on 802.1p priority

802.1p priority (decimal)	802.1p priority (binary)	Description
0	000	best-effort
1	001	background

802.1p priority (decimal)	802.1p priority (binary)	Description
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 only takes effect in any 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 range 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.

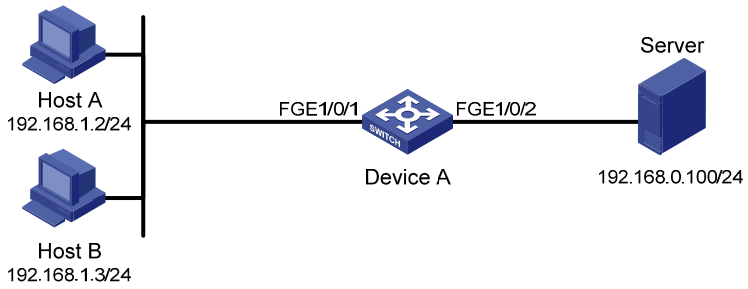
Task	Command
Display time range configuration and status	**display time-range** { *time-range-name* | **all** }

Time range configuration example

Network requirements

As shown in Figure 25, 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 25 Network diagram



Configuration procedure

Create a periodic time range during 8:00 and 18:00 on working days from June 2013 to the end of the year.

```
<DeviceA> system-view
[DeviceA] time-range work 8:0 to 18:0 working-day from 0:0 6/1/2013 to 24:0 12/31/2013
```

Create an IPv4 basic ACL numbered 2001, and configure a rule in the ACL to permit only packets 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 FortyGigE 1/0/2.

```
[DeviceA] interface FortyGigE 1/0/2
[DeviceA-FortyGigE1/0/2] packet-filter 2001 outbound
[DeviceA-FortyGigE1/0/2] quit
```

Verifying the configuration

Display time range configuration and status on Device A.

```
Current time is 13:19:14 7/30/2013 Tuesday

Time-range: work (Active)
 08:00 to 18:00 working-day
 from 00:00 6/1/2013 to 00:00 1/1/2014
```

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 26. When congestion occurs, the following rules apply:

  a. An interface first uses the relevant queues of the fixed area to store packets.

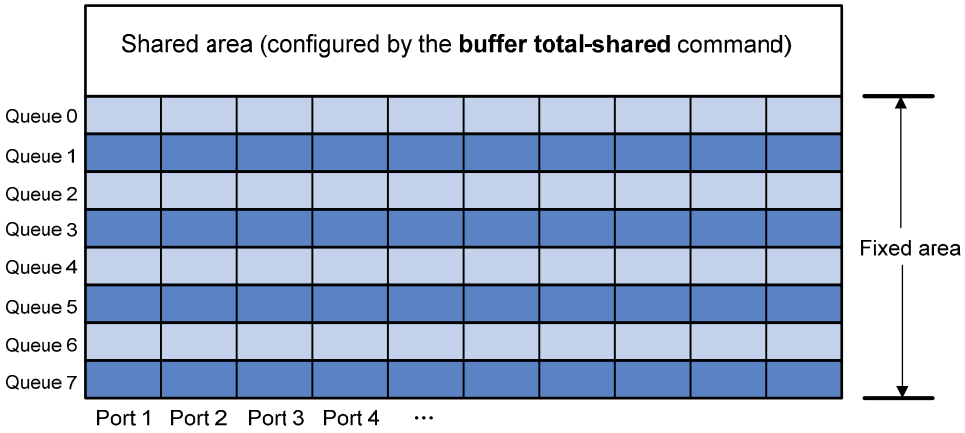
  b. When a queue is full, the interface uses the space for the queue in the shared area.

  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 26. 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 26 Fixed area and shared area



Configuration task list

Tasks at a glance
(Required.) Enabling the Burst feature
(Optional.) Configuring data buffer monitoring

Enabling the Burst feature

The Burst feature 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.

By enabling the Burst feature, you can improve the processing performance of the switch operating in these scenarios to reduce packet loss.

The Burst feature might affect the QoS performance of the switch.

Configuration prerequisites

Make sure you are fully aware of the impact when enabling the Burst feature.

Configuration procedure

To enable the Burst feature:

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

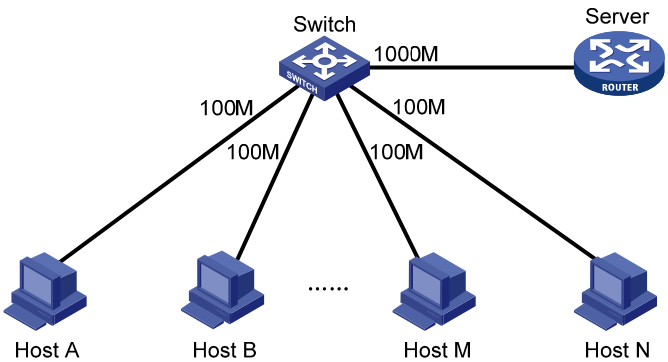
Burst configuration example

Network requirements

As shown in Figure 27, a server connects to the switch through a 1000 Mbps Ethernet interface. The server sends high-volume broadcast or multicast traffic to the hosts irregularly. Each host connects to the switch through a 100 Mbps network adapter.

Configure the switch to process high-volume traffic from the server to guarantee that packets can reach the hosts.

Figure 27 Network diagram



Configuration procedure

Enter system view.

```
<Switch> system-view
```

Enable the Burst feature.

```
[Switch] burst-mode enable
```

Configuring data buffer monitoring

IMPORTANT:

This feature is available in Release 1137 and later versions.

The data buffer on a switch is shared by all interfaces for buffering packets during periods of congestion.

This feature allows you to identify the interfaces that use an excessive amount of data buffer space. Then, you can diagnose those interfaces for anomalies.

You can set a per-interface buffer usage threshold. The buffer usage threshold for a queue is the same as the per-interface threshold value. The switch automatically records buffer usage for each interface. When a queue on an interface uses more buffer space than the set threshold, the system counts one threshold violation for the queue.

To configure data buffer monitoring:

Step	Command	Remarks
1. Enter system view.	**system-view**	N/A
2. Set the packet statistics collection mode to queue.	**statistic mode queue**	The default setting is vsi. For more information about packet statistics collection modes, see "Configuring queue-based accounting."
3. Set a per-interface buffer usage threshold.	• In standalone mode: **buffer usage threshold slot** *slot-number* **ratio** *ratio* • In IRF mode: **buffer usage threshold chassis** *chassis-number* **slot** *slot-number* **ratio** *ratio*	The default setting is 100%.
4. Return to user view.	**quit**	N/A
5. Display buffer usage statistics for interfaces.	**display buffer usage interface** [*interface-type* [*interface-number*]]	Available in any view.

Document conventions and icons

Conventions

This section describes the conventions used in the documentation.

Port numbering in examples

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

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 Boldface. 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

Convention	Description
	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 access controller engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a wireless terminator unit.
	Represents a wireless terminator.
	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.

Support and other resources

Accessing Hewlett Packard Enterprise Support

- For live assistance, go to the Contact Hewlett Packard Enterprise Worldwide website:

  www.hpe.com/assistance
- To access documentation and support services, go to the Hewlett Packard Enterprise Support Center website:

  www.hpe.com/support/hpesc

Information to collect

- Technical support registration number (if applicable)
- Product name, model or version, and serial number
- Operating system name and version
- Firmware version
- Error messages
- Product-specific reports and logs
- Add-on products or components
- Third-party products or components

Accessing updates

- Some software products provide a mechanism for accessing software updates through the product interface. Review your product documentation to identify the recommended software update method.
- To download product updates, go to either of the following:
  - Hewlett Packard Enterprise Support Center **Get connected with updates** page:

    www.hpe.com/support/e-updates
  - Software Depot website:

    www.hpe.com/support/softwaredepot
- To view and update your entitlements, and to link your contracts, Care Packs, and warranties with your profile, go to the Hewlett Packard Enterprise Support Center **More Information on Access to Support Materials** page:

www.hpe.com/support/AccessToSupportMaterials

IMPORTANT:

Access to some updates might require product entitlement when accessed through the Hewlett Packard Enterprise Support Center. You must have an HP Passport set up with relevant entitlements.

Websites

Website	Link
Networking websites	
Hewlett Packard Enterprise Networking Information Library	www.hpe.com/networking/resourcefinder
Hewlett Packard Enterprise Networking website	www.hpe.com/info/networking
Hewlett Packard Enterprise Networking My Support	www.hpe.com/networking/support
General websites	
Hewlett Packard Enterprise Information Library	www.hpe.com/info/enterprise/docs
Hewlett Packard Enterprise Support Center	www.hpe.com/support/hpesc
Contact Hewlett Packard Enterprise Worldwide	www.hpe.com/assistance
Subscription Service/Support Alerts	www.hpe.com/support/e-updates
Software Depot	www.hpe.com/support/softwaredepot
Customer Self Repair (not applicable to all devices)	www.hpe.com/support/selfrepair
Insight Remote Support (not applicable to all devices)	www.hpe.com/info/insightremotesupport/docs

Customer self repair

Hewlett Packard Enterprise customer self repair (CSR) programs allow you to repair your product. If a CSR part needs to be replaced, it will be shipped directly to you so that you can install it at your convenience. Some parts do not qualify for CSR. Your Hewlett Packard Enterprise authorized service provider will determine whether a repair can be accomplished by CSR.

For more information about CSR, contact your local service provider or go to the CSR website:

www.hpe.com/support/selfrepair

Remote support

Remote support is available with supported devices as part of your warranty, Care Pack Service, or contractual support agreement. It provides intelligent event diagnosis, and automatic, secure submission of hardware event notifications to Hewlett Packard Enterprise, which will initiate a fast and accurate resolution based on your product's service level. Hewlett Packard Enterprise strongly recommends that you register your device for remote support.

For more information and device support details, go to the following website:

www.hpe.com/info/insightremotesupport/docs

Documentation feedback

Hewlett Packard Enterprise is committed to providing documentation that meets your needs. To help us improve the documentation, send any errors, suggestions, or comments to Documentation Feedback (docsfeedback@hpe.com). When submitting your feedback, include the document title, part number, edition, and publication date located on the front cover of the document. For online help content, include the product name, product version, help edition, and publication date located on the legal notices page.

Index

T

U

V

W