Configuring QoS

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for QoS

Before configuring standard QoS, you must have a thorough understanding of these items:

- The types of applications used and the traffic patterns on your network.
- Traffic characteristics and needs of your network. For example, is the traffic on your network bursty? Do you need to reserve bandwidth for voice and video streams?
- Bandwidth requirements and speed of the network.
- Location of congestion points in the network.

General QoS Guidelines

These are the general QoS guidelines:

- Control traffic (such as spanning-tree bridge protocol data units [BPDUs] and routing update packets) received by the switch are subject to all ingress QoS processing.
- You are likely to lose data when you change queue settings; therefore, try to make changes when traffic is at a minimum.

Restrictions for QoS

The following are the restrictions for QoS:

- The switch does not support classifying of traffic using class maps (**class-map** global configuration command).
- Ingress queueing is not supported.
- Interface restrictions:
  - Enable only cos trust at interface level.
  - Enable SRR shaping and sharing at interface level.
  - Enable Priority queueing at interface level.

Information About QoS

QoS Implementation

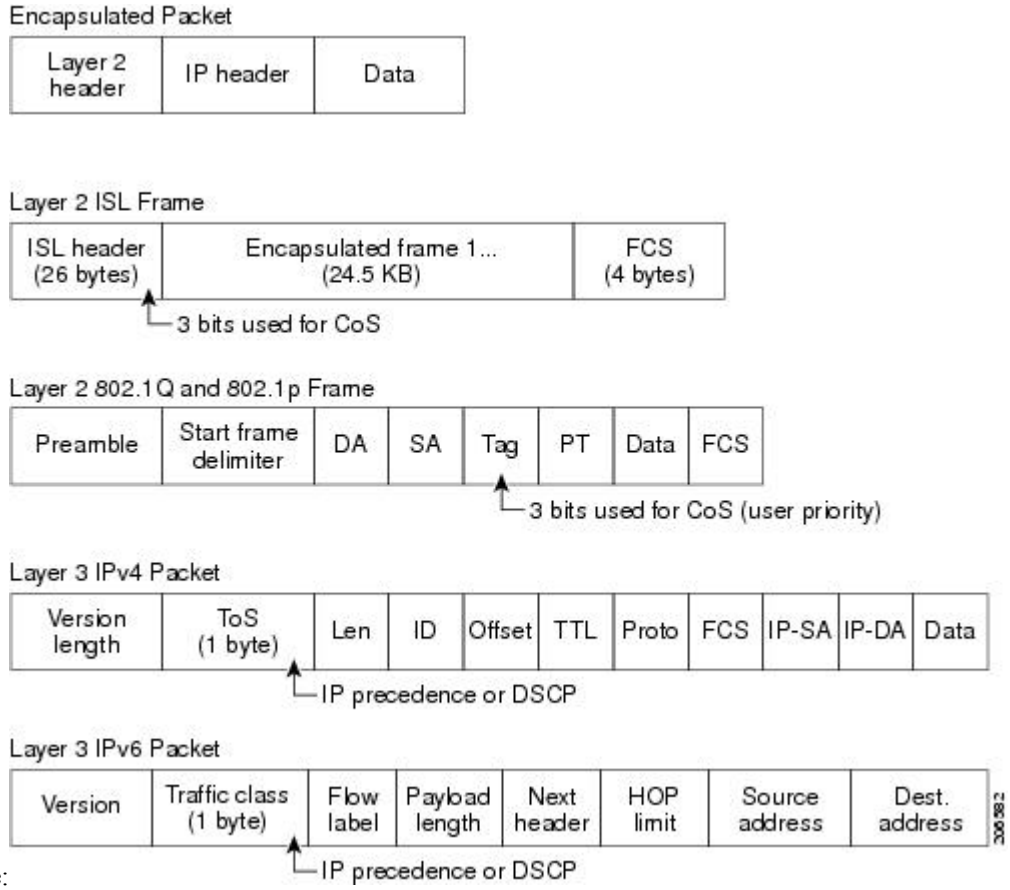
Typically, networks operate on a best-effort delivery basis, which means that all traffic has equal priority and an equal chance of being delivered in a timely manner. When congestion occurs, all traffic has an equal chance of being dropped.

When you configure the QoS feature, you can select specific network traffic, prioritize it according to its relative importance, and use congestion-management and congestion-avoidance techniques to provide preferential treatment. Implementing QoS in your network makes network performance more predictable and bandwidth utilization more effective.

The QoS implementation is based on the Differentiated Services (Diff-Serv) architecture, a standard from the Internet Engineering Task Force (IETF). This architecture specifies that each packet is classified upon entry into the network.

Figure 1: QoS Classification Layers in Frames and Packets

The special bits in the Layer 2 frame or a Layer 3 packet are shown in the following figure:



Layer 2 Frame Prioritization Bits

Layer 2 802.1Q frame headers have a 2-byte Tag Control Information field that carries the CoS value in the three most-significant bits, which are called the User Priority bits. On ports configured as Layer 2 802.1Q trunks, all traffic is in 802.1Q frames except for traffic in the native VLAN.

Other frame types cannot carry Layer 2 CoS values.

Layer 2 CoS values range from 0 for low priority to 7 for high priority.

Layer 3 Packet Prioritization Bits

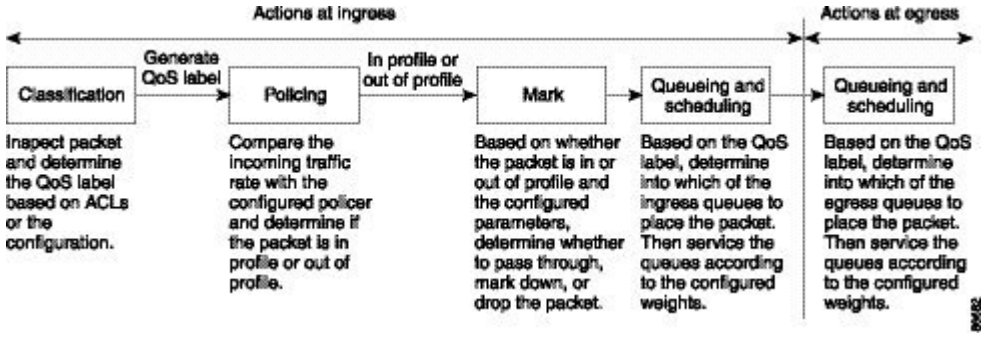
Layer 3 IP packets can carry either an IP precedence value or a Differentiated Services Code Point (DSCP) value. QoS supports the use of either value because DSCP values are backward-compatible with IP precedence values.

IP precedence values range from 0 to 7. DSCP values range from 0 to 63.

QoS Basic Model

To implement QoS, the switch must distinguish packets or flows from one another (classify), assign a label to indicate the given quality of service as the packets move through the switch, make the packets comply with the configured resource usage limits (police and mark), and provide different treatment (queue and schedule) in all situations where resource contention exists. The switch also needs to ensure that traffic sent from it meets a specific traffic profile (shape).

Figure 2: QoS Basic Wired Model



Actions at Ingress Port

Actions at the ingress port include classifying traffic, policing, marking, and scheduling:

- Classifying a distinct path for a packet by associating it with a QoS label. The switch maps the CoS or DSCP in the packet to a QoS label to distinguish one kind of traffic from another. The QoS label that is generated identifies all future QoS actions to be performed on this packet.
- Policing determines whether a packet is in or out of profile by comparing the rate of the incoming traffic to the configured policer. The policer limits the bandwidth consumed by a flow of traffic. The result is passed to the marker.
- Marking evaluates the policer and configuration information for the action to be taken when a packet is out of profile and determines what to do with the packet (pass through a packet without modification, marking down the QoS label in the packet, or dropping the packet).

Note Queueing and scheduling are only supported at egress and not at ingress on the switch.

Actions at Egress Port

Actions at the egress port include queueing and scheduling:

- Queueing evaluates the QoS packet label and the corresponding DSCP or CoS value before selecting which of the four egress queues to use. Because congestion can occur when multiple ingress ports simultaneously send data to an egress port, WTD differentiates traffic classes and subjects the packets to different thresholds based on the QoS label. If the threshold is exceeded, the packet is dropped.
- Scheduling services the four egress queues based on their configured SRR shared or shaped weights. One of the queues (queue 1) can be the priority queue, which is serviced until empty before the other queues are serviced.

Mapping Tables Overview

During QoS processing, the switch represents the priority of all traffic (including non-IP traffic) with a QoS label based on the DSCP or CoS value from the classification stage.

The following table describes QoS processing and mapping tables.

Table 1: QoS Processing and Mapping Tables

QoS Processing Stage	Mapping Table Usage
Classification	During the classification stage, QoS uses configurable mapping tables to derive a corresponding DSCP or CoS value from a received CoS, DSCP, or IP precedence value. These maps include the CoS-to-DSCP map and the IP-precedence-to-DSCP map. You configure these maps by using the **mls qos map cos-dscp** and the **mls qos map ip-prec-dscp** global configuration commands. On an ingress port configured in the DSCP-trusted state, if the DSCP values are different between the QoS domains, you can apply the configurable DSCP-to-DSCP-mutation map to the port that is on the boundary between the two QoS domains. You configure this map by using the **mls qos map dscp-mutation** global configuration command.
Policing	During policing stage, QoS can assign another DSCP value to an IP or a non-IP packet (if the packet is out of profile and the policer specifies a marked-down value). This configurable map is called the policed-DSCP map. You configure this map by using the **mls qos map policed-dscp** global configuration command.
Pre-scheduling	Before the traffic reaches the scheduling stage, QoS stores the packet in an egress queue according to the QoS label. The QoS label is based on the DSCP or the CoS value in the packet and selects the queue through the DSCP output queue threshold maps or through the CoS output queue threshold maps. In addition to an egress queue, the QOS label also identifies the WTD threshold value. You configure these maps by using the **mls qos srr-queue** { **output**} **dscp-map** and the **mls qos srr-queue** { **output**} **cos-map** global configuration commands.

The CoS-to-DSCP, DSCP-to-CoS, and the IP-precedence-to-DSCP maps have default values that might or might not be appropriate for your network.

The default DSCP-to-DSCP-mutation map and the default policed-DSCP map are null maps; they map an incoming DSCP value to the same DSCP value. The DSCP-to-DSCP-mutation map is the only map you apply to a specific port. All other maps apply to the entire switch.

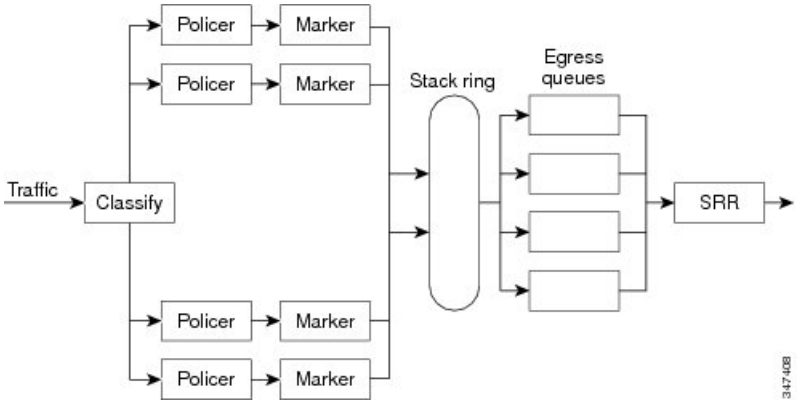
Related Topics

Configuring DSCP Maps
Queueing and Scheduling on Egress Queues

Queueing and Scheduling Overview

The switch has queues at specific points to help prevent congestion.

Figure 3: Egress Queue Location on Switch



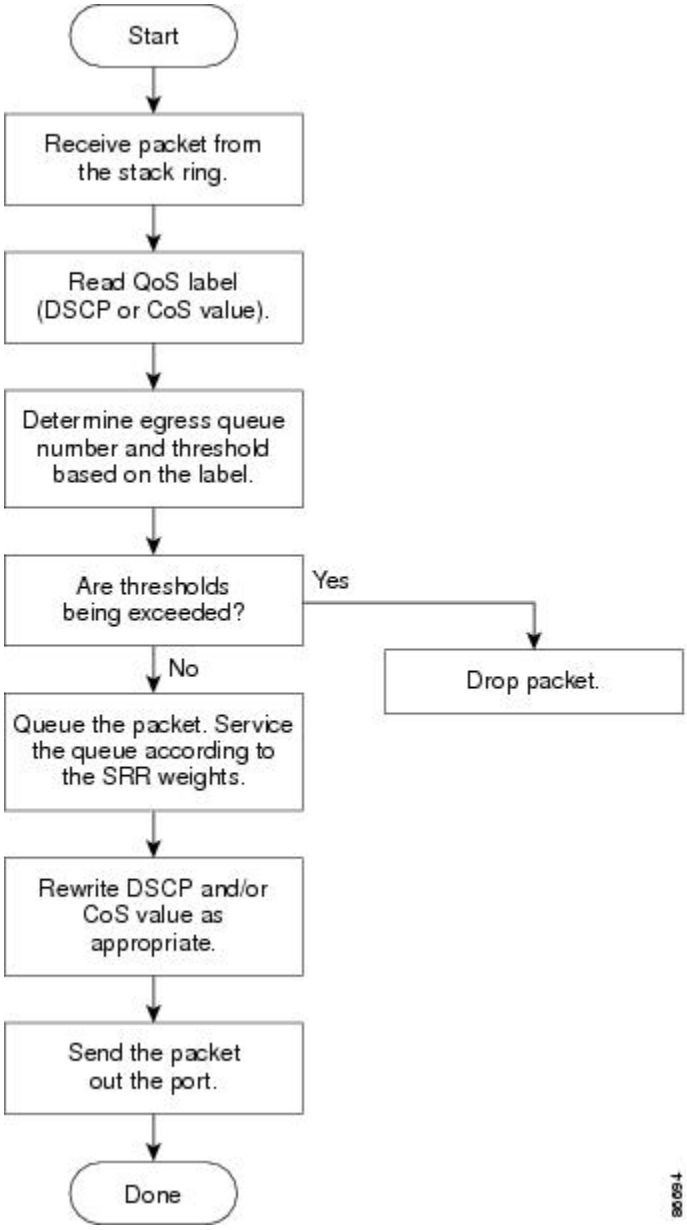
Note The switch supports 4 egress queues by default and there is an option to enable a total of 8 egress queues. The 8 egress queue configuration is only supported on a standalone switch.

The Catalyst 2960-L switches support Scheduled Round Robin (SRR). They do not support Weighted Round Robin (WRR). Currently, you can configure SRR with **wrr** commands instead of **srr** commands. From Cisco IOS Release 15.2(5)E2 and later, the **wrr** commands will be replaced with the **srr** commands on the switch.

Queueing and Scheduling on Egress Queues

The following figure shows queueing and scheduling flowcharts for egress ports on the switch.

Figure 4: Queueing and Scheduling Flowchart for Egress Ports on the Switch



Note If the priority queue is enabled, SRR services it until it is empty before servicing the other three queues.

Egress Expedite Queue

Each port supports four egress queues, one of which (queue 1) can be the egress expedite queue. These queues are assigned to a queue-set. All traffic exiting the switch flows through one of these four queues and is subjected to a threshold based on the QoS label assigned to the packet.

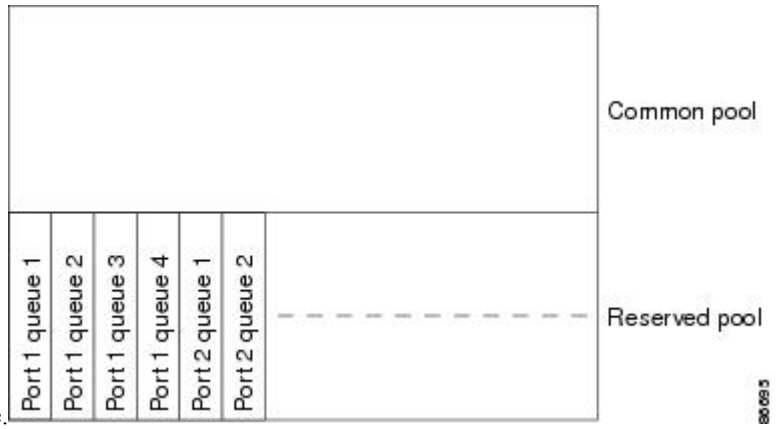
Note If the expedite queue is enabled, SRR services it until it is empty before servicing the other three queues.

Egress Queue Buffer Allocation

The following figure shows the egress queue buffer.

Figure 5: Egress Queue Buffer Allocation

The buffer space is divided between the common pool and the reserved pool. The switch uses a buffer allocation scheme to reserve a minimum amount of buffers for each egress queue, to prevent any queue or port from consuming all the buffers and depriving other queues, and to control whether to grant buffer space to a requesting queue. The switch detects whether the target queue has not consumed more buffers than its reserved amount (under-limit), whether it has consumed all of its maximum buffers (over limit), and whether the common pool is empty (no free buffers) or not empty (free buffers). If the queue is not over-limit, the switch can allocate buffer space from the common pool (if it is not empty). If there are no free buffers in the common pool or if the queue is over-limit, the switch drops the frame.



Buffer and Memory Allocation

You guarantee the availability of buffers, set drop thresholds, and configure the maximum memory allocation for a queue-set by using the **mls qos queue-set output** *qset-id* **threshold** *queue-id drop-threshold1 drop-threshold2 reserved-threshold maximum-threshold* global configuration command. Each threshold value is a percentage of the queue's allocated memory, which you specify by using the **mls qos queue-set output** *qset-id* **buffers** *allocation1 ... allocation4* global configuration command. The sum of all the allocated buffers represents the reserved pool, and the remaining buffers are part of the common pool.

Through buffer allocation, you can ensure that high-priority traffic is buffered. For example, if the buffer space is 400, you can allocate 70 percent of it to queue 1 and 10 percent to queues 2 through 4. Queue 1 then has 280 buffers allocated to it, and queues 2 through 4 each have 40 buffers allocated to them.

You can guarantee that the allocated buffers are reserved for a specific queue in a queue-set. For example, if there are 100 buffers for a queue, you can reserve 50 percent (50 buffers). The switch returns the remaining 50 buffers to the common pool. You also can enable a queue in the full condition to obtain more buffers than are reserved for it by setting a maximum threshold. The switch can allocate the needed buffers from the common pool if the common pool is not empty.

Queues and WTD Thresholds

You can assign each packet that flows through the switch to a queue and to a threshold.

Specifically, you map DSCP or CoS values to an egress queue and map DSCP or CoS values to a threshold ID. You use the **mls qos srr-queue output dscp-map queue** *queue-id* {*dscp1...dscp8* | **threshold** *threshold-id dscp1...dscp8*} or the **mls qos srr-queue output cos-map queue** *queue-id* {*cos1...cos8* | **threshold** *threshold-id cos1...cos8*} global configuration command. You can display the DSCP output queue threshold map and the CoS output queue threshold map by using the **show mls qos maps** privileged EXEC command.

The queues use WTD to support distinct drop percentages for different traffic classes. Each queue has three drop thresholds: two configurable (*explicit*) WTD thresholds and one nonconfigurable (*implicit*) threshold preset to the queue-full state. You assign the two WTD threshold percentages for threshold ID 1 and ID 2. The drop threshold for threshold ID 3 is preset to the queue-full state, and you cannot modify it. You map a port to queue-set by using the **queue-set qset-id** interface configuration command. Modify the queue-set configuration to change the WTD threshold percentages.

Note The switch supports 4 egress queues by default, although there is an option to enable a total of 8 egress queues. Use the **mls qos srr-queue output queues 8** global configuration command to enable all 8 egress queues. Once 8 egress queues are enabled, you are able to configure thresholds and buffers for all 8 queues. The 8 egress queue configuration is only supported on a standalone switch.

Related Topics

Weighted Tail Drop

Shaped or Shared Mode

SRR services each queue-set in shared or shaped mode. You map a port to a queue-set by using the **queue-set** *qset-id* interface configuration command. You assign shared or shaped weights to the port by using the **srr-queue bandwidth share** *weight1 weight2 weight3 weight4* or the **srr-queue bandwidth shape** *weight1 weight2 weight3 weight4* interface configuration command.

The buffer allocation together with the SRR weight ratios control how much data can be buffered and sent before packets are dropped. The weight ratio is the ratio of the frequency in which the SRR scheduler sends packets from each queue.

All four queues participate in the SRR unless the expedite queue is enabled, in which case the first bandwidth weight is ignored and is not used in the ratio calculation. The expedite queue is a priority queue, and it is serviced until empty before the other queues are serviced. You enable the expedite queue by using the **priority-queue out** interface configuration command.

You can combine the commands described in this section to prioritize traffic by placing packets with particular CoSs into certain queues, by allocating a large queue size or by servicing the queue more frequently, and by adjusting queue thresholds so that packets with lower priorities are dropped.

Note The egress queue default settings are suitable for most situations. You should change them only when you have a thorough understanding of the egress queues and if these settings do not meet your QoS solution.

Related Topics

Configuring Egress Queue Characteristics, on page 33
SRR Shaping and Sharing

Packet Modification

A packet is classified and queued to provide QoS. The following packet modifications can occur during the process to provide QoS:

- For IP and non-IP packets, classification involves assigning a QoS label to a packet based on the CoS of the received packet. However, the packet is not modified at this stage; only an indication of the assigned CoS value is carried along.
- If you configure the port to trust the CoS of the incoming frame and it is an IP packet, the CoS value in the frame is not changed.

Standard QoS Default Configuration

QoS is disabled by default.

When QoS is disabled, there is no concept of trusted or untrusted ports because the packets are not modified. The CoS, DSCP, and IP precedence values in the packet are not changed.

Traffic is switched in pass-through mode. The packets are switched without any rewrites and classified as best effort without any policing.

When QoS is enabled using the **mls qos** global configuration command and all other QoS settings are at their defaults, traffic is classified as best effort (the DSCP and CoS value is set to 0) without any policing. No policy maps are configured. The default port trust state on all ports is untrusted.

Related Topics

Enabling QoS Globally, on page 11
Default Egress Queue Configuration, on page 10

Default Egress Queue Configuration

The following tables describe the default egress queue configurations.

The following table shows the default egress queue configuration for each queue-set when QoS is enabled. All ports are mapped to queue-set 1. The port bandwidth limit is set to 100 percent and rate unlimited. Note that for the SRR shaped weights (absolute) feature, a shaped weight of zero indicates that the queue is operating in shared mode. Note that for the SRR shared weights feature, one quarter of the bandwidth is allocated to each queue.

Table 2: Default Egress Queue Configuration

Feature	Queue 1	Queue 2	Queue 3	Queue 4
Buffer allocation	25 percent	25 percent	25 percent	25 percent
WTD drop threshold 1	100 percent	200 percent	100 percent	100 percent
WTD drop threshold 2	100 percent	200 percent	100 percent	100 percent
Reserved threshold	50 percent	50 percent	50 percent	50 percent
Maximum threshold	400 percent	400 percent	400 percent	400 percent

Feature	Queue 1	Queue 2	Queue 3	Queue 4
SRR shaped weights (absolute)	25	0	0	0
SRR shared weights	25	25	25	25

The following table shows the default CoS output queue threshold map when QoS is enabled.

Table 3: Default CoS Output Queue Threshold Map

CoS Value	Queue ID–Threshold ID
0, 1	2–1
2, 3	3–1
4	4–1
5	1–1
6, 7	4–1

Related Topics

Enabling QoS Globally, on page 11
Standard QoS Default Configuration, on page 10

Default Mapping Table Configuration

The default DSCP-to-DSCP-mutation map is a null map, which maps an incoming DSCP value to the same DSCP value.

The default policed-DSCP map is a null map, which maps an incoming DSCP value to the same DSCP value (no markdown).

Related Topics

Default CoS-to-DSCP Map
Default IP-Precedence-to-DSCP Map
Default DSCP-to-CoS Map

How to Configure QoS

Enabling QoS Globally

By default, QoS is disabled on the switch.

The following procedure to enable QoS globally is required.

SUMMARY STEPS

1. **configure terminal**

2. **mls qos**
3. **end**
4. **show mls qos**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**mls qos** **Example:** `Switch(config)# mls qos`	Enables QoS globally. QoS operates with the default settings described in the related topic sections below. **Note** To disable QoS, use the **no mls qos** global configuration command.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show mls qos** **Example:** `Switch# show mls qos`	Verifies the QoS configuration.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Standard QoS Default Configuration, on page 10
Default Egress Queue Configuration, on page 10

Enabling VLAN-Based QoS on Physical Ports

By default, VLAN-based QoS is disabled on all physical switch ports. You can enable VLAN-based QoS on a switch port.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **mls qos vlan-based**
4. **end**
5. **show mls qos interface** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Specifies the physical port, and enter interface configuration mode.
Step 3	**mls qos vlan-based** **Example:** `Switch(config-if)# mls qos vlan-based`	Enables VLAN-based QoS on the port. **Note** Use the **no mls qos vlan-based** interface configuration command to disable VLAN-based QoS on the physical port.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show mls qos interface** *interface-id* **Example:** `Switch# show mls qos interface gigabitethernet 1/0/1`	Verifies if VLAN-based QoS is enabled on the physical port.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring a QoS Policy

Configuring a QoS policy typically requires the following tasks:

- Classifying traffic into classes
- Configuring policies applied to those traffic classes
- Attaching policies to ports

These sections describe how to classify, police, and mark traffic. Depending on your network configuration, you must perform one or more of the modules in this section.

Related Topics

Policing and Marking Overview
Classification Overview

Classifying Traffic by Using ACLs

You can classify IP traffic by using IPv4 standard ACLS, IPv4 extended ACLs, or IPv6 ACLs.

You can classify non-IP traffic by using Layer 2 MAC ACLs.

Creating an IP Standard ACL for IPv4 Traffic

Before you begin

Before you perform this task, determine which access lists you will be using for your QoS configuration.

SUMMARY STEPS

1. **configure terminal**
2. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
3. **end**
4. **show access-lists**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**access-list** *access-list-number* {**deny**	**permit**} *source* [*source-wildcard*] **Example:** `Switch(config)# access-list 1`	Creates an IP standard ACL, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number. The range is 1 to 99 and 1300 to 1999.

	Command or Action	Purpose
	`permit 192.2.255.0 1.1.1.255`	• Use the **permit** keyword to permit a certain type of traffic if the conditions are matched. Use the **deny** keyword to deny a certain type of traffic if conditions are matched. • For *source*, enter the network or host from which the packet is being sent. You can use the **any** keyword as an abbreviation for 0.0.0.0 255.255.255.255. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. When you create an access list, remember that by default the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end. **Note** To delete an access list, use the **no access-list** *access-list-number* global configuration command.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show access-lists** **Example:** `Switch# show access-lists`	Verifies your entries.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy-running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Access Control Lists
QoS ACL Guidelines
Examples: Classifying Traffic by Using ACLs, on page 45

Creating an IP Extended ACL for IPv4 Traffic

Before you begin

Before you perform this task, determine which access lists you will be using for your QoS configuration.

SUMMARY STEPS

1. **configure terminal**
2. **access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard*
3. **end**
4. **show access-lists**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**access-list** *access-list-number* {**deny**	**permit**} *protocol source source-wildcard destination destination-wildcard* **Example:** `Switch(config)# access-list 100 permit ip any any dscp 32`	Creates an IP extended ACL, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number. The range is 100 to 199 and 2000 to 2699. • Use the **permit** keyword to permit a certain type of traffic if the conditions are matched. Use the **deny** keyword to deny a certain type of traffic if conditions are matched. • For *protocol*, enter the name or number of an IP protocol. Use the question mark (?) to see a list of available protocol keywords. • For *source*, enter the network or host from which the packet is being sent. You specify this by using dotted decimal notation, by using the **any** keyword as an abbreviation for *source* 0.0.0.0 *source-wildcard* 255.255.255.255, or by using the **host** keyword for *source* 0.0.0.0. • For *source-wildcard*, enter the wildcard bits by placing ones in the bit positions that you want to ignore. You specify the wildcard by using dotted decimal notation, by using the **any** keyword as an abbreviation for *source* 0.0.0.0 *source-wildcard* 255.255.255.255, or by using the **host** keyword for *source* 0.0.0.0.

	Command or Action	Purpose
		• For *destination*, enter the network or host to which the packet is being sent. You have the same options for specifying the *destination and destination-wildcard* as those described by *source* and *source-wildcard*. When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end. **Note** To delete an access list, use the **no access-list** *access-list-number* global configuration command.
Step 3	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show access-lists** **Example:** `Switch# show access-lists`	Verifies your entries.
Step 5	**copy running-config startup-config** **Example:** `Switch# copy-running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Access Control Lists
QoS ACL Guidelines
Examples: Classifying Traffic by Using ACLs, on page 45

Creating an IPv6 ACL for IPv6 Traffic

Before you begin

Before you perform this task, determine which access lists you will be using for your QoS configuration.

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 access-list** *access-list-name*
3. {**deny** | **permit**} *protocol* {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix/ prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dscp** *value*] [**fragments**] [**log**] [**log-input**] [**routing**] [**sequence** *value*] [**time-range** *name*]

4. **end**
5. **show ipv6 access-list**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ipv6 access-list** *access-list-name* **Example:** `Switch(config)# ipv6 access-list ipv6_Name_ACL`	Creates an IPv6 ACL and enters IPv6 access-list configuration mode. Accesses list names cannot contain a space or quotation mark or begin with a numeric. **Note** To delete an access list, use the **no ipv6 access-list** *access-list-number* global configuration command.
Step 3	{**deny** | **permit**} *protocol* {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix/ prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dscp** *value*] [**fragments**] [**log**] [**log-input**] [**routing**] [**sequence** *value*] [**time-range** *name*] **Example:** `Switch(config-ipv6-acl)# permit ip host 10::1 host 11::2 host`	Enters **deny** or **permit** to specify whether to deny or permit the packet if conditions are matched. These are the conditions: For *protocol*, enter the name or number of an Internet protocol: **ahp**, **esp**, **icmp**, **ipv6**, **pcp**, **stcp**, **tcp**, or **udp**, or an integer in the range 0 to 255 representing an IPv6 protocol number. • The *source-ipv6-prefix/prefix-length* or *destination-ipv6-prefix/ prefix-length* is the source or destination IPv6 network or class of networks for which to set deny or permit conditions, specified in hexadecimal and using 16-bit values between colons (see RFC 2373). • Enter **any** as an abbreviation for the IPv6 prefix ::/0. • For **host** *source-ipv6-address* or *destination-ipv6-address*, enter the source or destination IPv6 host address for which to set deny or permit conditions, specified in hexadecimal using 16-bit values between colons. • (Optional) For *operator*, specify an operand that compares the source or destination ports of the specified protocol. Operands are **lt** (less than), **gt** (greater than), **eq** (equal), **neq** (not equal), and **range**. If the operator follows the *source-ipv6-prefix/prefix-length* argument, it must

	Command or Action	Purpose
		match the source port. If the operator follows the *destination-ipv6- prefix/prefix-length* argument, it must match the destination port. • (Optional) The *port-number* is a decimal number from 0 to 65535 or the name of a TCP or UDP port. You can use TCP port names only when filtering TCP. You can use UDP port names only when filtering UDP. • (Optional) Enter **dscp** *value* to match a differentiated services code point value against the traffic class value in the Traffic Class field of each IPv6 packet header. The acceptable range is from 0 to 63. • (Optional) Enter **fragments** to check noninitial fragments. This keyword is visible only if the protocol is IPv6. • (Optional) Enter **log** to cause a logging message to be sent to the console about the packet that matches the entry. Enter **log-input** to include the input interface in the log entry. Logging is supported only for router ACLs. • (Optional) Enter **routing** to specify that IPv6 packets be routed. • (Optional) Enter **sequence** *value* to specify the sequence number for the access list statement. The acceptable range is from 1 to 4294967295. • (Optional) Enter **time-range** *name* to specify the time range that applies to the deny or permit statement.
Step 4	**end** **Example:** `Switch(config-ipv6-acl)# end`	Returns to privileged EXEC mode.
Step 5	**show ipv6 access-list** **Example:** `Switch# show ipv6 access-list`	Verifies the access list configuration.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy-running-config`	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	`startup-config`	

Related Topics

Access Control Lists
QoS ACL Guidelines
Examples: Classifying Traffic by Using ACLs, on page 45
QoS ACL IPv6 Guidelines

Creating a Layer 2 MAC ACL for Non-IP Traffic

Before you begin

Before you perform this task, determine that Layer 2 MAC access lists are required for your QoS configuration.

SUMMARY STEPS

1. **configure terminal**
2. **mac access-list extended** *name*
3. {**permit** | **deny**} {**host** *src-MAC-addr mask* | **any** | **host** *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask*]
4. **end**
5. **show access-lists** [*access-list-number* | *access-list-name*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**mac access-list extended** *name* **Example:** `Switch(config)# mac access-list extended maclist1`	Creates a Layer 2 MAC ACL by specifying the name of the list. After entering this command, the mode changes to extended MAC ACL configuration. **Note** To delete an access list, use the **no mac access-list extended** *access-list-name* global configuration command.
Step 3	{**permit**	**deny**} {**host** *src-MAC-addr mask*	**any**	**host** *dst-MAC-addr*	*dst-MAC-addr mask*} [*type mask*] **Example:** `Switch(config-ext-mac1) # permit 0001.0000.0001`	Specifies the type of traffic to permit or deny if the conditions are matched, entering the command as many times as necessary. • For *src-MAC-addr*, enter the MAC address of the host from which the packet is being sent. You specify this

	Command or Action	Purpose

	`0.0.0 0002.0000.0001 0.0.0`
	
	`Switch(config-ext-mac1) # permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp`
		by using the hexadecimal format (H.H.H), by using the **any** keyword as an abbreviation for *source* 0.0.0, *source-wildcard* ffff.ffff.ffff, or by using the **host** keyword for *source* 0.0.0.
		• For *mask*, enter the wildcard bits by placing ones in the bit positions that you want to ignore.
		• For *dst-MAC-addr*, enter the MAC address of the host to which the packet is being sent. You specify this by using the hexadecimal format (H.H.H), by using the **any** keyword as an abbreviation for *source* 0.0.0, *source-wildcard* ffff.ffff.ffff, or by using the **host** keyword for *source* 0.0.0.
		• (Optional) For *type mask*, specify the Ethertype number of a packet with Ethernet II or SNAP encapsulation to identify the protocol of the packet. For *type*, the range is from 0 to 65535, typically specified in hexadecimal. For *mask*, enter the *don't care* bits applied to the Ethertype before testing for a match.
		
		When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.

Step 4	**end** **Example:** `Switch(config-ext-mac1)# end`	Returns to privileged EXEC mode.
Step 5	**show access-lists** [*access-list-number* | *access-list-name*] **Example:** `Switch# show access-lists`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy-running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Access Control Lists
QoS ACL Guidelines
Examples: Classifying Traffic by Using ACLs, on page 45

Classifying Traffic by Using Class Maps

You use the **class-map** global configuration command to name and to isolate a specific traffic flow (or class) from all other traffic. The class map defines the criteria to use to match against a specific traffic flow to further classify it. Match statements can include criteria such as an ACL, IP precedence values, or DSCP values. The match criterion is defined with one match statement entered within the class-map configuration mode.

Note You can also create class maps during policy map creation by using the **class** policy-map configuration command.

SUMMARY STEPS

1. **configure terminal**
2. Use one of the following:
   - **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
   - **access-list** *access-list-number* {**deny** | **permit**} *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*]
   - **ipv6 access-list** *access-list-name* {**deny** | **permit**} *protocol* {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix/ prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dscp** *value*] [**fragments**] [**log**] [**log-input**] [**routing**] [**sequence** *value*] [**time-range** *name*]
   - **mac access-list extended** *name* {**permit** | **deny**} {**host** *src-MAC-addr mask* | **any** | **host** *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask*]
3. **class-map** [**match-all** | **match-any**] *class-map-name*
4. **match** {**access-group** *acl-index-or-name* | **ip dscp** *dscp-list* | **ip precedence** *ip-precedence-list*}
5. **end**
6. **show class-map**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	Use one of the following: • **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] • **access-list** *access-list-number* {**deny** | **permit**} *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*] • **ipv6 access-list** *access-list-name* {**deny** | **permit**} *protocol* {*source-ipv6-prefix/prefix-length* | **any** | **host**	Creates an IP standard or extended ACL, an IPv6 ACL for IP traffic, or a Layer 2 MAC ACL for non-IP traffic, repeating the command as many times as necessary. When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.

	Command or Action	Purpose
	source-ipv6-address} [*operator* [*port-number*]] {*destination-ipv6-prefix/ prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dscp** *value*] [**fragments**] [**log**] [**log-input**] [**routing**] [**sequence** *value*] [**time-range** *name*] • **mac access-list extended** *name* {**permit** | **deny**} {**host** *src-MAC-addr mask* | **any** | **host** *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask*] **Example:** `Switch(config)# access-list 103 permit ip any any dscp 10`	
Step 3	**class-map** [**match-all** | **match-any**] *class-map-name* **Example:** `Switch(config)# class-map class1`	Creates a class map, and enters class-map configuration mode. By default, no class maps are defined. • (Optional) Use the **match-all** keyword to perform a logical-AND of all matching statements under this class map. All match criteria in the class map must be matched. • (Optional) Use the **match-any** keyword to perform a logical-OR of all matching statements under this class map. One or more match criteria must be matched. • For *class-map-name*, specify the name of the class map. If neither the **match-all** or **match-any** keyword is specified, the default is **match-all**. **Note** To delete an existing class map, use the **no class-map** [**match-all** | **match-any**] *class-map-name* global configuration command.
Step 4	**match** {**access-group** *acl-index-or-name* | **ip dscp** *dscp-list* | **ip precedence** *ip-precedence-list*} **Example:** `Switch(config-cmap)# match ip dscp 10 11 12`	Defines the match criterion to classify traffic. By default, no match criterion is defined. Only one match criterion per class map is supported, and only one ACL per class map is supported. • For **access-group** *acl-index-or-name,* specify the number or name of the ACL created in Step 2. • To filter IPv6 traffic with the **match access-group** command, create an IPv6 ACL, as described in Step 2. • For **ip dscp** *dscp-list*, enter a list of up to eight IP DSCP values to match against incoming packets. Separate each value with a space. The range is 0 to 63.

	Command or Action	Purpose
		• For **ip precedence** *ip-precedence-list*, enter a list of up to eight IP-precedence values to match against incoming packets. Separate each value with a space. The range is 0 to 7. **Note** To remove a match criterion, use the **no match** {**access-group** *acl-index-or-name* | **ip dscp** | **ip precedence**} class-map configuration command.
Step 5	**end** **Example:** `Switch(config-cmap)# end`	Returns to privileged EXEC mode.
Step 6	**show class-map** **Example:** `Switch# show class-map`	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** `Switch# copy-running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Classifying, Policing, and Marking Traffic on Physical Ports by Using Policy Maps, on page 26
Classifying, Policing, and Marking Traffic on SVIs by Using Hierarchical Policy Maps
Examples: Classifying Traffic by Using Class Maps, on page 46

Classifying Traffic by Using Class Maps and Filtering IPv6 Traffic

Note IPv6 QoS is not supported on switches running the LAN base feature set.

To apply the primary match criteria to only IPv4 traffic, use the **match protocol** command with the **ip** keyword. To apply the primary match criteria to only IPv6 traffic, use the **match protocol** command with the **ipv6** keyword.

SUMMARY STEPS

1. **configure terminal**
2. **class-map** {**match-all**} *class-map-name*
3. **match protocol** [*ip* | *ipv6*]
4. **match** {**ip dscp** *dscp-list* | **ip precedence** *ip-precedence-list*}

5. **end**
6. **show class-map**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**class-map** {**match-all**} *class-map-name* **Example:** `Switch(config)# class-map cm-1`	Creates a class map, and enters class-map configuration mode. By default, no class maps are defined. When you use the **match protocol** command, only the **match-all** keyword is supported. • For *class-map-name*, specify the name of the class map. If neither the **match-all** or **match-any** keyword is specified, the default is **match-all**. **Note** To delete an existing class map, use the **no class-map** [**match-all** | **match-any**] *class-map-name* global configuration command.
Step 3	**match protocol** [*ip* | *ipv6*] **Example:** `Switch(config-cmap)# match protocol ip`	(Optional) Specifies the IP protocol to which the class map applies: • Use the argument *ip* to specify IPv4 traffic and *ipv6* to specify IPv6 traffic. • When you use the **match protocol** command, only the **match-all** keyword is supported for the **class-map** command.
Step 4	**match** {**ip dscp** *dscp-list* | **ip precedence** *ip-precedence-list*} **Example:** `Switch(config-cmap)# match ip dscp 10`	Defines the match criterion to classify traffic. By default, no match criterion is defined. • For **ip dscp** *dscp-list*, enter a list of up to eight IP DSCP values to match against incoming packets. Separate each value with a space. The range is 0 to 63. • For **ip precedence** *ip-precedence-list*, enter a list of up to eight IP-precedence values to match against incoming packets. Separate each value with a space. The range is 0 to 7.

	Command or Action	Purpose
		Note To remove a match criterion, use the no match {**access-group** *acl-index-or-name* | **ip dscp** | **ip precedence**} class-map configuration command.
Step 5	**end** **Example:** `Switch(config-cmap)# end`	Returns to privileged EXEC mode.
Step 6	**show class-map** **Example:** `Switch# show class-map`	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** `Switch# copy-running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Examples: Classifying Traffic by Using Class Maps, on page 46

Classifying, Policing, and Marking Traffic on Physical Ports by Using Policy Maps

You can configure a policy map on a physical port that specifies which traffic class to act on. Actions can include trusting the CoS, DSCP, or IP precedence values in the traffic class; setting a specific DSCP or IP precedence value in the traffic class; and specifying the traffic bandwidth limitations for each matched traffic class (policer) and the action to take when the traffic is out of profile (marking).

A policy map also has these characteristics:

- A policy map can contain multiple class statements, each with different match criteria and policers.
- A policy map can contain a predefined default traffic class explicitly placed at the end of the map.
- A separate policy-map class can exist for each type of traffic received through a port.

Follow these guidelines when configuring policy maps on physical ports:

- You can attach only one policy map per ingress port.
- If you configure the IP-precedence-to-DSCP map by using the **mls qos map ip-prec-dscp** *dscp1...dscp8* global configuration command, the settings only affect packets on ingress interfaces that are configured to trust the IP precedence value. In a policy map, if you set the packet IP precedence value to a new value by using the **set ip precedence** *new-precedence* policy-map class configuration command, the egress DSCP value is not affected by the IP-precedence-to-DSCP map. If you want the egress DSCP value to be different than the ingress value, use the **set dscp new-dscp** policy-map class configuration command.

- If you enter or have used the **set ip dscp** command, the switch changes this command to **set dscp** in its configuration.
- You can use the **set ip precedence** or the **set precedence** policy-map class configuration command to change the packet IP precedence value. This setting appears as set ip precedence in the switch configuration.
- A policy-map and a port trust state can both run on a physical interface. The policy-map is applied before the port trust state.
- When you configure a default traffic class by using the **class class-default** policy-map configuration command, unclassified traffic (traffic that does not meet the match criteria specified in the traffic classes) is treated as the default traffic class (class-default).

SUMMARY STEPS

1. **configure terminal**
2. **class-map** [**match-all** | **match-any**] *class-map-name*
3. **policy-map** *policy-map-name*
4. **class** [*class-map-name* | **class-default**]
5. **trust** [**cos** | **dscp** | **ip-precedence**]
6. **set** {**dscp** *new-dscp* | **ip precedence** *new-precedence*}
7. **police** *rate-bps burst-byte* [**exceed-action** {**drop** | **policed-dscp-transmit**}]
8. **exit**
9. **exit**
10. **interface** *interface-id*
11. **service-policy input** *policy-map-name*
12. **end**
13. **show policy-map** [*policy-map-name* [**class** *class-map-name*]]
14. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**class-map** [**match-all**	**match-any**] *class-map-name* **Example:** `Switch(config)# class-map ipclass1`	Creates a class map, and enters class-map configuration mode. By default, no class maps are defined. • (Optional) Use the **match-all** keyword to perform a logical-AND of all matching statements under this class map. All match criteria in the class map must be matched.

	Command or Action	Purpose
		• (Optional) Use the **match-any** keyword to perform a logical-OR of all matching statements under this class map. One or more match criteria must be matched. • For *class-map-name*, specify the name of the class map. If neither the **match-all** or **match-any** keyword is specified, the default is **match-all**.
Step 3	**policy-map** *policy-map-name* **Example:** `Switch(config-cmap)# policy-map flowit`	Creates a policy map by entering the policy map name, and enters policy-map configuration mode. By default, no policy maps are defined. The default behavior of a policy map is to set the DSCP to 0 if the packet is an IP packet and to set the CoS to 0 if the packet is tagged. No policing is performed. **Note** To delete an existing policy map, use the **no policy-map** *policy-map-name* global configuration command.
Step 4	**class** [*class-map-name* | **class-default**] **Example:** `Switch(config-pmap)# class ipclass1`	Defines a traffic classification, and enters policy-map class configuration mode. By default, no policy map class-maps are defined. If a traffic class has already been defined by using the **class-map** global configuration command, specify its name for *class-map-name* in this command. A **class-default** traffic class is pre-defined and can be added to any policy. It is always placed at the end of a policy map. With an implied **match any** included in the **class-default** class, all packets that have not already matched the other traffic classes will match **class-default**. **Note** To delete an existing class map, use the **no class** *class-map-name* policy-map configuration command.
Step 5	**trust** [**cos** | **dscp** | **ip-precedence**] **Example:** `Switch(config-pmap-c)# trust dscp`	Configures the trust state, which QoS uses to generate a CoS-based or DSCP-based QoS label. This command is mutually exclusive with the **set** command within the same policy map. If you enter the **trust** command, go to Step 6. By default, the port is not trusted. If no keyword is specified when the command is entered, the default is **dscp**. The keywords have these meanings:

	Command or Action	Purpose
		• **cos**—QoS derives the DSCP value by using the received or default port CoS value and the CoS-to-DSCP map. • **dscp**—QoS derives the DSCP value by using the DSCP value from the ingress packet. For non-IP packets that are tagged, QoS derives the DSCP value by using the received CoS value; for non-IP packets that are untagged, QoS derives the DSCP value by using the default port CoS value. In either case, the DSCP value is derived from the CoS-to-DSCP map. • **ip-precedence**—QoS derives the DSCP value by using the IP precedence value from the ingress packet and the IP-precedence-to-DSCP map. For non-IP packets that are tagged, QoS derives the DSCP value by using the received CoS value; for non-IP packets that are untagged, QoS derives the DSCP value by using the default port CoS value. In either case, the DSCP value is derived from the CoS-to-DSCP map. **Note** To return to the untrusted state, use the **no trust** policy-map configuration command
Step 6	**set** {**dscp** *new-dscp* | **ip precedence** *new-precedence*} **Example:** `Switch(config-pmap-c)# set dscp 45`	Classifies IP traffic by setting a new value in the packet. • For **dscp** *new-dscp*, enter a new DSCP value to be assigned to the classified traffic. The range is 0 to 63. • For **ip precedence** *new-precedence,* enter a new IP-precedence value to be assigned to the classified traffic. The range is 0 to 7. **Note** To remove an assigned DSCP or IP precedence value, use the **no set** {**dscp** *new-dscp* | **ip** precedence *new-precedence*} policy-map configuration command.
Step 7	**police** *rate-bps burst-byte* [**exceed-action** {**drop** | **policed-dscp-transmit**}] **Example:** `Switch(config-pmap-c)# police 100000 80000 drop`	Defines a policer for the classified traffic. By default, no policer is defined. • For *rate-bps,* specify average traffic rate in bits per second (b/s). The range is 8000 to 10000000000. • For *burst-byte,* specify the normal burst size in bytes. The range is 8000 to 1000000. • (Optional) Specifies the action to take when the rates are exceeded. Use the **exceed-action drop** keywords to drop the packet. Use the **exceed-action policed-dscp-transmit** keywords to mark down the

	Command or Action	Purpose
		DSCP value (by using the policed-DSCP map) and to send the packet. **Note** To remove an existing policer, use the **no police** *rate-bps burst-byte* [**exceed-action** {**drop** | **policed-dscp-transmit**}] policy-map configuration command.
Step 8	**exit** **Example:** `Switch(config-pmap-c)# exit`	Returns to policy map configuration mode.
Step 9	**exit** **Example:** `Switch(config-pmap)# exit`	Returns to global configuration mode.
Step 10	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 2/0/1`	Specifies the port to attach to the policy map, and enters interface configuration mode. Valid interfaces include physical ports.
Step 11	**service-policy input** *policy-map-name* **Example:** `Switch(config-if)# service-policy input flowit`	Specifies the policy-map name, and applies it to an ingress port. Only one policy map per ingress port is supported. **Note** To remove the policy map and port association, use the **no service-policy** *input policy-map-name* interface configuration command.
Step 12	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 13	**show policy-map** [*policy-map-name* [**class** *class-map-name*]] **Example:** `Switch# show policy-map`	Verifies your entries.

	Command or Action	Purpose
Step 14	**copy running-config startup-config** **Example:** `Switch# copy-running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Policing and Marking Overview
Physical Port Policing
Classifying Traffic by Using Class Maps, on page 22
Policy Map on Physical Port
Examples: Classifying, Policing, and Marking Traffic on Physical Ports Using Policy Maps, on page 48
Policy Map on Physical Port Guidelines

Classifying, Policing, and Marking Traffic by Using Aggregate Policers

By using an aggregate policer, you can create a policer that is shared by multiple traffic classes within the same policy map. However, you cannot use the aggregate policer across different policy maps or ports.

You can configure aggregate policers only in nonhierarchical policy maps on physical ports.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos aggregate-policer** *aggregate-policer-name rate-bps burst-byte* **exceed-action** {**drop** | **policed-dscp-transmit**}
3. **class-map** [**match-all** | **match-any**] *class-map-name*
4. **policy-map** *policy-map-name*
5. **class** [*class-map-name* | **class-default**]
6. **police aggregate** *aggregate-policer-name*
7. **exit**
8. **interface** *interface-id*
9. **service-policy input** *policy-map-name*
10. **end**
11. **show mls qos aggregate-policer** [*aggregate-policer-name*]
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**mls qos aggregate-policer** *aggregate-policer-name rate-bps burst-byte* **exceed-action** {**drop** | **policed-dscp-transmit**} **Example:** `Switch(config)# mls qos aggregate-police transmit1 48000 8000 exceed-action policed-dscp-transmit`	Defines the policer parameters that can be applied to multiple traffic classes within the same policy map. By default, no aggregate policer is defined. • For *aggregate-policer-name*, specify the name of the aggregate policer. • For *rate-bps,* specify average traffic rate in bits per second (b/s). The range is 8000 to 10000000000. • For *burst-byte,* specify the normal burst size in bytes. The range is 8000 to 1000000. • Specifies the action to take when the rates are exceeded. Use the **exceed-action drop** keywords to drop the packet. Use the **exceed-action policed-dscp-transmit** keywords to mark down the DSCP value (by using the policed-DSCP map) and to send the packet.
Step 3	**class-map** [**match-all** | **match-any**] *class-map-name* **Example:** `Switch(config)# class-map ipclass1`	Creates a class map to classify traffic as necessary.
Step 4	**policy-map** *policy-map-name* **Example:** `Switch(config-cmap)# policy-map aggflow1`	Creates a policy map by entering the policy map name, and enters policy-map configuration mode.
Step 5	**class** [*class-map-name* | **class-default**] **Example:** `Switch(config-cmap-p)# class ipclass1`	Defines a traffic classification, and enters policy-map class configuration mode.
Step 6	**police aggregate** *aggregate-policer-name* **Example:** `Switch(configure-cmap-p)# police aggregate transmit1`	Applies an aggregate policer to multiple classes in the same policy map. For *aggregate-policer-name*, enter the name specified in Step 2. To remove the specified aggregate policer from a policy map, use the **no police aggregate** *aggregate-policer-name* policy map configuration command. To delete an aggregate policer and its parameters, use the **no mls qos aggregate-policer** *aggregate-policer-name* global configuration command.

	Command or Action	Purpose
Step 7	**exit** **Example:** `Switch(configure-cmap-p)# exit`	Returns to global configuration mode.
Step 8	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 2/0/1`	Specifies the port to attach to the policy map, and enters interface configuration mode. Valid interfaces include physical ports.
Step 9	**service-policy input** *policy-map-name* **Example:** `Switch(config-if)# service-policy input aggflow1`	Specifies the policy-map name, and applies it to an ingress port. Only one policy map per ingress port is supported.
Step 10	**end** **Example:** `Switch(configure-if)# end`	Returns to privileged EXEC mode.
Step 11	**show mls qos aggregate-policer** [*aggregate-policer-name*] **Example:** `Switch# show mls qos aggregate-policer transmit1`	Verifies your entries.
Step 12	**copy running-config startup-config** **Example:** `Switch# copy-running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Policing and Marking Overview

Examples: Classifying, Policing, and Marking Traffic by Using Aggregate Policers, on page 49

Configuring Egress Queue Characteristics

Depending on the complexity of your network and your QoS solution, you might need to perform all of the tasks in the following modules. You need to make decisions about these characteristics:

- Which packets are mapped by DSCP or CoS value to each queue and threshold ID?
- What drop percentage thresholds apply to the queue-set (four egress queues per port), and how much reserved and maximum memory is needed for the traffic type?

- How much of the fixed buffer space is allocated to the queue-set?
- Does the bandwidth of the port need to be rate limited?
- How often should the egress queues be serviced and which technique (shaped, shared, or both) should be used?

Related Topics

Shaped or Shared Mode, on page 9

Configuration Guidelines

Follow these guidelines when the expedite queue is enabled or the egress queues are serviced based on their SRR weights:

- If the egress expedite queue is enabled, it overrides the SRR shaped and shared weights for queue 1.
- If the egress expedite queue is disabled and the SRR shaped and shared weights are configured, the shaped mode overrides the shared mode for queue 1, and SRR services this queue in shaped mode.
- If the egress expedite queue is disabled and the SRR shaped weights are not configured, SRR services this queue in shared mode.

Allocating Buffer Space to and Setting WTD Thresholds for an Egress Queue-Set

You can guarantee the availability of buffers, set WTD thresholds, and configure the maximum allocation for a queue-set by using the **mls qos queue-set output** *qset-id* **threshold** *queue-id drop-threshold1 drop-threshold2 reserved-threshold maximum-threshold* global configuration command.

Each threshold value is a percentage of the queue's allocated buffers, which you specify by using the **mls qos queue-set output** *qset-id* **buffers** *allocation1 ... allocation4* global configuration command. The queues use WTD to support distinct drop percentages for different traffic classes.

Note The egress queue default settings are suitable for most situations. You should change them only when you have a thorough understanding of the egress queues and if these settings do not meet your QoS solution.

Beginning in privileged EXEC mode, follow these steps to configure the memory allocation and to drop thresholds for a queue-set. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **queue-set** *qset-id*
4. **end**
5. **show mls qos interface** [*interface-id*] **buffers**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Specifies the port of the outbound traffic, and enter interface configuration mode.
Step 3	**queue-set** *qset-id* **Example:** `Switch(config-id)# queue-set 2`	Maps the port to a queue-set. For *qset-id*, enter the ID of the queue-set specified in Step 2. The range is 1 to 2. The default is 1.
Step 4	**end** **Example:** `Switch(config-id)# end`	Returns to privileged EXEC mode.
Step 5	**show mls qos interface** [*interface-id*] **buffers** **Example:** `Switch# show mls qos interface buffers`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy-running-config startup-config`	(Optional) Saves your entries in the configuration file. To return to the default setting, use the **no mls qos queue-set output** *qset-id* **buffers** global configuration command. To return to the default WTD threshold percentages, use the **no mls qos queue-set output** *qset-id* **threshold** [*queue-id*] global configuration command.

Related Topics

Queueing and Scheduling on Egress Queues
Examples: Configuring Egress Queue Characteristics, on page 52

Mapping DSCP or CoS Values to an Egress Queue and to a Threshold ID

You can prioritize traffic by placing packets with particular DSCPs or costs of service into certain queues and adjusting the queue thresholds so that packets with lower priorities are dropped.

Note The egress queue default settings are suitable for most situations. You should change them only when you have a thorough understanding of egress queues and if these settings do not meet your QoS solution.

Beginning in privileged EXEC mode, follow these steps to map DSCP or CoS values to an egress queue and to a threshold ID. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. Use one of the following:
   - **mls qos srr-queue output dscp-map queue** *queue-id* **threshold** *threshold-id dscp1...dscp8*
   - **mls qos srr-queue output cos-map queue** *queue-id* **threshold** *threshold-id cos1...cos8*
3. **mls qos srr-queue output cos-map queue** *queue-id* **threshold** *threshold-id cos1...cos8*
4. **end**
5. **show mls qos maps**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	Use one of the following: • **mls qos srr-queue output dscp-map queue** *queue-id* **threshold** *threshold-id dscp1...dscp8* • **mls qos srr-queue output cos-map queue** *queue-id* **threshold** *threshold-id cos1...cos8* **Example:** `Switch(config)# mls qos srr-queue output dscp-map queue 1 threshold 2 10 11`	Maps DSCP or CoS values to an egress queue and to a threshold ID. By default, DSCP values 0–15 are mapped to queue 2 and threshold 1. DSCP values 16–31 are mapped to queue 3 and threshold 1. DSCP values 32–39 and 48–63 are mapped to queue 4 and threshold 1. DSCP values 40–47 are mapped to queue 1 and threshold 1. By default, CoS values 0 and 1 are mapped to queue 2 and threshold 1. CoS values 2 and 3 are mapped to queue 3 and threshold 1. CoS values 4, 6, and 7 are mapped to queue 4 and threshold 1. CoS value 5 is mapped to queue 1 and threshold 1. • For *queue-id*, the range is 1 to 4. **Note** If you enabled 8 egress queues using the **mls qos srr-queue output queues 8** global configuration command, then the *queue-id* range would be from 1 to 8.

	Command or Action	Purpose
		• For *threshold-id*, the range is 1 to 2. The drop-threshold percentage for threshold 3 is predefined. It is set to the queue-full state. • For *dscp1...dscp8*, enter up to eight values, and separate each value with a space. The range is 0 to 63. • For *cos1...cos8*, enter up to eight values, and separate each value with a space. The range is 0 to 7. **Note** To return to the default DSCP output queue threshold map or the default CoS output queue threshold map, use the **no mls qos srr-queue output dscp-map** or the **no mls qos srr-queue output cos-map** global configuration command.
Step 3	**mls qos srr-queue output cos-map queue** *queue-id* **threshold** *threshold-id cos1...cos8* **Example:** `Switch(config)# mls qos srr-queue output cos-map queue 3 threshold 1 2 3`	Maps CoS values to an egress queue and to a threshold ID. By default, CoS values 0 and 1 are mapped to queue 2 and threshold 1. CoS values 2 and 3 are mapped to queue 3 and threshold 1. CoS values 4, 6, and 7 are mapped to queue 4 and threshold 1. CoS value 5 is mapped to queue 1 and threshold 1. • For *queue-id*, the range is 1 to 4. • For *threshold-id*, the range is 1 to 2. The drop-threshold percentage for threshold 3 is predefined. It is set to the queue-full state. • For *cos1...cos8*, enter up to eight values, and separate each value with a space. The range is 0 to 7. **Note** To return to the default CoS output queue threshold map, use the **no mls qos srr-queue output cos-map** global configuration command.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show mls qos maps** **Example:** `Switch# show mls qos maps`	Verifies your entries. The DSCP output queue threshold map appears as a matrix. The d1 column specifies the most-significant digit of the DSCP number; the d2 row specifies the least-significant digit in the DSCP number. The intersection of the d1 and the d2 values provides the queue ID and threshold ID; for example, queue 2 and threshold 1 (02-01).

	Command or Action	Purpose
		The CoS output queue threshold map shows the CoS value in the top row and the corresponding queue ID and threshold ID in the second row; for example, queue 2 and threshold 2 (2-2).
Step 6	**copy running-config startup-config** **Example:** `Switch# copy-running-config startup-config`	(Optional) Saves your entries in the configuration file. To return to the default DSCP output queue threshold map or the default CoS output queue threshold map, use the **no mls qos srr-queue output dscp-map** or the **no mls qos srr-queue output cos-map** global configuration command.

Related Topics

Queueing and Scheduling on Egress Queues
Examples: Configuring Egress Queue Characteristics, on page 52

Configuring SRR Shaped Weights on Egress Queues

You can specify how much of the available bandwidth is allocated to each queue. The ratio of the weights is the ratio of frequency in which the SRR scheduler sends packets from each queue.

You can configure the egress queues for shaped or shared weights, or both. Use shaping to smooth bursty traffic or to provide a smoother output over time.

Beginning in privileged EXEC mode, follow these steps to assign the shaped weights and to enable bandwidth shaping on the four egress queues mapped to a port. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **srr-queue bandwidth shape** *weight1 weight2 weight3 weight4*
4. **end**
5. **show mls qos interface** *interface-id* **queueing**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface`	Specifies the port of the outbound traffic, and enters interface configuration mode.

	Command or Action	Purpose
	`gigabitethernet2/0/1`	
Step 3	**srr-queue bandwidth shape** *weight1 weight2 weight3 weight4* **Example:** `Switch(config-if)# srr-queue bandwidth shape 8 0 0 0`	Assigns SRR weights to the egress queues. By default, weight1 is set to 25; weight2, weight3, and weight4 are set to 0, and these queues are in shared mode. For *weight1 weight2 weight3 weight4*, enter the weights to control the percentage of the port that is shaped. The inverse ratio (1/weight) controls the shaping bandwidth for this queue. Separate each value with a space. The range is 0 to 65535. If you configure a weight of 0, the corresponding queue operates in shared mode. The weight specified with the **srr-queue bandwidth shape** command is ignored, and the weights specified with the **srr-queue bandwidth share** interface configuration command for a queue come into effect. When configuring queues in the same queue-set for both shaping and sharing, make sure that you configure the lowest number queue for shaping. The shaped mode overrides the shared mode. To return to the default setting, use the **no srr-queue bandwidth shape** interface configuration command.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show mls qos interface** *interface-id* **queueing** **Example:** `Switch# show mls qos interface interface-id queuing`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file. To return to the default setting, use the **no srr-queue bandwidth shape** interface configuration command.

Related Topics

Queueing and Scheduling on Egress Queues
Examples: Configuring Egress Queue Characteristics, on page 52
SRR Shaping and Sharing

Configuring SRR Shared Weights on Egress Queues

In shared mode, the queues share the bandwidth among them according to the configured weights. The bandwidth is guaranteed at this level but not limited to it. For example, if a queue empties and does not require a share of the link, the remaining queues can expand into the unused bandwidth and share it among them. With sharing, the ratio of the weights controls the frequency of dequeuing; the absolute values are meaningless.

Note The egress queue default settings are suitable for most situations. You should change them only when you have a thorough understanding of the egress queues and if these settings do not meet your QoS solution.

Beginning in privileged EXEC mode, follow these steps to assign the shared weights and to enable bandwidth sharing on the four egress queues mapped to a port. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **srr-queue bandwidth share** *weight1 weight2 weight3 weight4*
4. **end**
5. **show mls qos interface** *interface-id* **queueing**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/1`	Specifies the port of the outbound traffic, and enters interface configuration mode.
Step 3	**srr-queue bandwidth share** *weight1 weight2 weight3 weight4* **Example:** `Switch(config-id)# srr-queue bandwidth share 1 2 3 4`	Assigns SRR weights to the egress queues. By default, all four weights are 25 (1/4 of the bandwidth is allocated to each queue). For *weight1 weight2 weight3 weight4*, enter the weights to control the ratio of the frequency in which the SRR scheduler sends packets. Separate each value with a space. The range is 1 to 255. To return to the default setting, use the **no srr-queue bandwidth share** interface configuration command.

	Command or Action	Purpose
Step 4	**end** **Example:** `Switch(config-id)# end`	Returns to privileged EXEC mode.
Step 5	**show mls qos interface** *interface-id* **queueing** **Example:** `Switch# show mls qos interface interface_id queuing`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy-running-config startup-config`	(Optional) Saves your entries in the configuration file. To return to the default setting, use the **no srr-queue bandwidth share** interface configuration command.

Related Topics

Queueing and Scheduling on Egress Queues
Examples: Configuring Egress Queue Characteristics, on page 52
SRR Shaping and Sharing

Configuring the Egress Expedite Queue

You can ensure that certain packets have priority over all others by queuing them in the egress expedite queue. SRR services this queue until it is empty before servicing the other queues.

Beginning in privileged EXEC mode, follow these steps to enable the egress expedite queue. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos**
3. **interface** *interface-id*
4. **priority-queue out**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**mls qos** **Example:** `Switch(config)# mls qos`	Enables QoS on a switch.
Step 3	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Specifies the egress port, and enters interface configuration mode.
Step 4	**priority-queue out** **Example:** `Switch(config-if)# priority-queue out`	Enables the egress expedite queue, which is disabled by default. When you configure this command, the SRR weight and queue size ratios are affected because there is one fewer queue participating in SRR. This means that *weight1* in the **srr-queue bandwidth shape** or the **srr-queue bandwidth share** command is ignored (not used in the ratio calculation). **Note** To disable the egress expedite queue, use the **no priority-queue out** interface configuration command.
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:**	(Optional) Saves your entries in the configuration file. To disable the egress expedite queue, use the **no priority-queue out** interface configuration command.

	Command or Action	Purpose
	`Switch# copy running-config startup-config`	

Related Topics

Queueing and Scheduling on Egress Queues

Examples: Configuring Egress Queue Characteristics, on page 52

Limiting the Bandwidth on an Egress Interface

You can limit the bandwidth on an egress port. For example, if a customer pays only for a small percentage of a high-speed link, you can limit the bandwidth to that amount.

Note The egress queue default settings are suitable for most situations. You should change them only when you have a thorough understanding of the egress queues and if these settings do not meet your QoS solution.

Beginning in privileged EXEC mode, follow these steps to limit the bandwidth on an egress port. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **srr-queue bandwidth limit** *weight1*
4. **end**
5. **show mls qos interface** [*interface-id*] **queueing**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/1`	Specifies the port to be rate-limited, and enters interface configuration mode.

	Command or Action	Purpose
Step 3	**srr-queue bandwidth limit** *weight1* **Example:** `Switch(config-if)# srr-queue bandwidth limit 80`	Specifies the percentage of the port speed to which the port should be limited. The range is 10 to 90. By default, the port is not rate-limited and is set to 100 percent. **Note** To return to the default setting, use the **no srr-queue bandwidth limit** interface configuration command.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show mls qos interface** [*interface-id*] **queueing** **Example:** `Switch# show mls qos interface interface_id queueing`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy-running-config startup-config`	(Optional) Saves your entries in the configuration file. To return to the default setting, use the **no srr-queue bandwidth limit** interface configuration command.

Related Topics

Queueing and Scheduling on Egress Queues

Examples: Configuring Egress Queue Characteristics, on page 52

Monitoring Standard QoS

Table 4: Commands for Monitoring Standard QoS on the Switch

Command	Description
show mls qos	Displays global QoS configuration information.
show mls qos interface [*interface-id*] [**queueing** | **statistics**] **show mls qos interface** [*interface-id*] [| **queueing** | **statistics**]	Displays QoS information at the port level, including the queueing strategy, and the ingress and egress statistics.
show mls qos maps [**cos-dscp** | **cos-output-q**]	Displays QoS mapping information.
show running-config | include rewrite	Displays the DSCP transparency setting.

Configuration Examples for QoS

Example: Configuring Port to the DSCP-Trusted State and Modifying the DSCP-to-DSCP-Mutation Map

This example shows how to configure a port to the DSCP-trusted state and to modify the DSCP-to-DSCP-mutation map (named *gi1/0/2-mutation*) so that incoming DSCP values 10 to 13 are mapped to DSCP 30:

```
Switch(config)# mls qos map dscp-mutation gigabitethernet1/0/2-mutation
10 11 12 13 to 30
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# mls qos trust dscp
Switch(config-if)# mls qos dscp-mutation gigabitethernet1/0/2-mutation
Switch(config-if)# end
```

Related Topics

Configuring the DSCP Trust State on a Port Bordering Another QoS Domain

Examples: Classifying Traffic by Using ACLs

This example shows how to allow access for only those hosts on the three specified networks. The wildcard bits apply to the host portions of the network addresses. Any host with a source address that does not match the access list statements is rejected.

```
Switch(config)# access-list 1 permit 192.5.255.0 0.0.0.255
Switch(config)# access-list 1 permit 128.88.0.0 0.0.255.255
Switch(config)# access-list 1 permit 36.0.0.0 0.0.0.255
! (Note: all other access implicitly denied)
```

This example shows how to create an ACL that permits IP traffic from any source to any destination that has the DSCP value set to 32:

```
Switch(config)# access-list 100 permit ip any any dscp 32
```

This example shows how to create an ACL that permits IP traffic from a source host at 10.1.1.1 to a destination host at 10.1.1.2 with a precedence value of 5:

```
Switch(config)# access-list 100 permit ip host 10.1.1.1 host 10.1.1.2 precedence 5
```

This example shows how to create an ACL that permits PIM traffic from any source to a destination group address of 224.0.0.2 with a DSCP set to 32:

```
Switch(config)# access-list 102 permit pim any 224.0.0.2 dscp 32
```

This example shows how to create an ACL that permits IPv6 traffic from any source to any destination that has the DSCP value set to 32:

```
Switch(config)# ipv6 access-list 100 permit ip any any dscp 32
```

This example shows how to create an ACL that permits IPv6 traffic from a source host at 10.1.1.1 to a destination host at 10.1.1.2 with a precedence value of 5:

```
Switch(config)# ipv6 access-list ipv6_Name_ACL permit ip host 10::1 host 10.1.1.2
precedence 5
```

This example shows how to create a Layer 2 MAC ACL with two permit statements. The first statement allows traffic from the host with MAC address 0001.0000.0001 to the host with MAC address 0002.0000.0001. The second statement allows only Ethertype XNS-IDP traffic from the host with MAC address 0001.0000.0002 to the host with MAC address 0002.0000.0002.

```
Switch(config)# mac access-list extended maclist1
Switch(config-ext-macl)# permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0
Switch(config-ext-macl)# permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp
! (Note: all other access implicitly denied)
```

Related Topics

Creating an IP Standard ACL for IPv4 Traffic, on page 14
Creating an IP Extended ACL for IPv4 Traffic, on page 16
Creating an IPv6 ACL for IPv6 Traffic, on page 17
Creating a Layer 2 MAC ACL for Non-IP Traffic, on page 20

Examples: Classifying Traffic by Using Class Maps

This example shows how to configure the class map called *class1*. The *class1* has one match criterion, which is access list 103. It permits traffic from any host to any destination that matches a DSCP value of 10.

```
Switch(config)# access-list 103 permit ip any any dscp 10
Switch(config)# class-map class1
Switch(config-cmap)# match access-group 103
Switch(config-cmap)# end
Switch#
```

This example shows how to create a class map called *class2*, which matches incoming traffic with DSCP values of 10, 11, and 12.

```
Switch(config)# class-map class2
Switch(config-cmap)# match ip dscp 10 11 12
Switch(config-cmap)# end
Switch#
```

This example shows how to create a class map called *class3*, which matches incoming traffic with IP-precedence values of 5, 6, and 7:

```
Switch(config)# class-map class3
```

```
Switch(config-cmap)# match ip precedence 5 6 7
Switch(config-cmap)# end
Switch#
```

This example shows how to configure a class map to match IP DSCP and IPv6:

```
Switch(config)# Class-map cm-1
Switch(config-cmap)# match ip dscp 10
Switch(config-cmap)# match protocol ipv6
Switch(config-cmap)# exit
Switch(config)# Class-map cm-2
Switch(config-cmap)# match ip dscp 20
Switch(config-cmap)# match protocol ip
Switch(config-cmap)# exit
Switch(config)# Policy-map pm1
Switch(config-pmap)# class cm-1
Switch(config-pmap-c)# set dscp 4
Switch(config-pmap-c)# exit
Switch(config-pmap)# class cm-2
Switch(config-pmap-c)# set dscp 6
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface G1/0/1
Switch(config-if)# service-policy input pm1
```

This example shows how to configure a class map that applies to both IPv4 and IPv6 traffic:

```
Switch(config)# ip access-list 101 permit ip any any
Switch(config)# ipv6 access-list ipv6-any permit ip any any
Switch(config)# Class-map cm-1
Switch(config-cmap)# match access-group 101
Switch(config-cmap)# exit
Switch(config)# class-map cm-2
Switch(config-cmap)# match access-group name ipv6-any
Switch(config-cmap)# exit
Switch(config)# Policy-map pm1
Switch(config-pmap)# class cm-1
Switch(config-pmap-c)# set dscp 4
Switch(config-pmap-c)# exit
Switch(config-pmap)# class cm-2
Switch(config-pmap-c)# set dscp 6
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface G0/1
Switch(config-if)# switch mode access
Switch(config-if)# service-policy input pm1
```

Related Topics

Classifying Traffic by Using Class Maps, on page 22
Classifying Traffic by Using Class Maps and Filtering IPv6 Traffic, on page 24

Examples: Classifying, Policing, and Marking Traffic on Physical Ports Using Policy Maps

This example shows how to create a policy map and attach it to an ingress port. In the configuration, the IP standard ACL permits traffic from network 10.1.0.0. For traffic matching this classification, the DSCP value in the incoming packet is trusted. If the matched traffic exceeds an average traffic rate of 48000 b/s and a normal burst size of 8000 bytes, its DSCP is marked down (based on the policed-DSCP map) and sent:

```
Switch(config)# access-list 1 permit 10.1.0.0 0.0.255.255
Switch(config)# class-map ipclass1
Switch(config-cmap)# match access-group 1
Switch(config-cmap)# exit
Switch(config)# policy-map flow1t
Switch(config-pmap)# class ipclass1
Switch(config-pmap-c)# trust dscp
Switch(config-pmap-c)# police 1000000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# service-policy input flow1t
```

This example shows how to create a Layer 2 MAC ACL with two permit statements and attach it to an ingress port. The first permit statement allows traffic from the host with MAC address 0001.0000.0001 destined for the host with MAC address 0002.0000.0001. The second permit statement allows only Ethertype XNS-IDP traffic from the host with MAC address 0001.0000.0002 destined for the host with MAC address 0002.0000.0002.

```
Switch(config)# mac access-list extended maclist1
Switch(config-ext-mac)# permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0
Switch(config-ext-mac)# permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp
Switch(config-ext-mac)# exit
Switch(config)# mac access-list extended maclist2
Switch(config-ext-mac)# permit 0001.0000.0003 0.0.0 0002.0000.0003 0.0.0
Switch(config-ext-mac)# permit 0001.0000.0004 0.0.0 0002.0000.0004 0.0.0 aarp
Switch(config-ext-mac)# exit
Switch(config)# class-map macclass1
Switch(config-cmap)# match access-group maclist1
Switch(config-cmap)# exit
Switch(config)# policy-map macpolicy1
Switch(config-pmap)# class macclass1
Switch(config-pmap-c)# set dscp 63
Switch(config-pmap-c)# exit
Switch(config-pmap)# class macclass2 maclist2
Switch(config-pmap-c)# set dscp 45
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# mls qos trust cos
Switch(config-if)# service-policy input macpolicy1
```

This example shows how to create a class map that applies to both IPv4 and IPv6 traffic with the default class applied to unclassified traffic:

```
Switch(config)# ip access-list 101 permit ip any any
Switch(config)# ipv6 access-list ipv6-any permit ip any any
```

```
Switch(config)# class-map cm-1
Switch(config-cmap)# match access-group 101
Switch(config-cmap)# exit
Switch(config)# class-map cm-2
Switch(config-cmap)# match access-group name ipv6-any
Switch(config-cmap)# exit
Switch(config)# policy-map pm1
Switch(config-pmap)# class cm-1
Switch(config-pmap-c)# set dscp 4
Switch(config-pmap-c)# exit
Switch(config-pmap)# class cm-2
Switch(config-pmap-c)# set dscp 6
Switch(config-pmap-c)# exit
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# set dscp 10
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface G0/1
Switch(config-if)# switch mode access
Switch(config-if)# service-policy input pm1
```

Related Topics

Classifying, Policing, and Marking Traffic on Physical Ports by Using Policy Maps, on page 26
Policy Map on Physical Port

Examples: Classifying, Policing, and Marking Traffic by Using Aggregate Policers

This example shows how to create an aggregate policer and attach it to multiple classes within a policy map. In the configuration, the IP ACLs permit traffic from network 10.1.0.0 and from host 11.3.1.1. For traffic coming from network 10.1.0.0, the DSCP in the incoming packets is trusted. For traffic coming from host 11.3.1.1, the DSCP in the packet is changed to 56. The traffic rate from the 10.1.0.0 network and from host 11.3.1.1 is policed. If the traffic exceeds an average rate of 48000 b/s and a normal burst size of 8000 bytes, its DSCP is marked down (based on the policed-DSCP map) and sent. The policy map is attached to an ingress port.

```
Switch(config)# access-list 1 permit 10.1.0.0 0.0.255.255
Switch(config)# access-list 2 permit 11.3.1.1
Switch(config)# mls qos aggregate-police transmit1 48000 8000 exceed-action
policed-dscp-transmit
Switch(config)# class-map ipclass1
Switch(config-cmap)# match access-group 1
Switch(config-cmap)# exit
Switch(config)# class-map ipclass2
Switch(config-cmap)# match access-group 2
Switch(config-cmap)# exit
Switch(config)# policy-map aggflow1
Switch(config-pmap)# class ipclass1
Switch(config-pmap-c)# trust dscp
Switch(config-pmap-c)# police aggregate transmit1
Switch(config-pmap-c)# exit
Switch(config-pmap)# class ipclass2
Switch(config-pmap-c)# set dscp 56
Switch(config-pmap-c)# police aggregate transmit1
Switch(config-pmap-c)# exit
Switch(config-pmap)# class class-default
```

```
Switch(config-pmap-c)# set dscp 10
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# service-policy input aggflow1
Switch(config-if)# exit
```

Related Topics

Classifying, Policing, and Marking Traffic by Using Aggregate Policers, on page 31

Examples: Configuring DSCP Maps

This example shows how to modify and display the CoS-to-DSCP map:

```
Switch(config)# mls qos map cos-dscp 10 15 20 25 30 35 40 45
Switch(config)# end
Switch# show mls qos maps cos-dscp

Cos-dscp map:
        cos:   0  1  2  3  4  5  6  7
     --------------------------------
       dscp:   10 15 20 25 30 35 40 45
```

This example shows how to modify and display the IP-precedence-to-DSCP map:

```
Switch(config)# mls qos map ip-prec-dscp 10 15 20 25 30 35 40 45
Switch(config)# end
Switch# show mls qos maps ip-prec-dscp

IpPrecedence-dscp map:
     ipprec:   0  1  2  3  4  5  6  7
     --------------------------------
       dscp:   10 15 20 25 30 35 40 45
```

This example shows how to map DSCP 50 to 57 to a marked-down DSCP value of 0:

```
Switch(config)# mls qos map policed-dscp 50 51 52 53 54 55 56 57 to 0
Switch(config)# end
Switch# show mls qos maps policed-dscp
Policed-dscp map:
     d1 :  d2 0  1  2  3  4  5  6  7  8  9
     ---------------------------------------
      0 :    00 01 02 03 04 05 06 07 08 09
      1 :    10 11 12 13 14 15 16 17 18 19
      2 :    20 21 22 23 24 25 26 27 28 29
      3 :    30 31 32 33 34 35 36 37 38 39
      4 :    40 41 42 43 44 45 46 47 48 49
      5 :    00 00 00 00 00 00 00 00 58 59
      6 :    60 61 62 63
```

Note In this policed-DSCP map, the marked-down DSCP values are shown in the body of the matrix. The d1 column specifies the most-significant digit of the original DSCP; the d2 row specifies the least-significant digit of the original DSCP. The intersection of the d1 and d2 values provides the marked-down value. For example, an original DSCP value of 53 corresponds to a marked-down DSCP value of 0.

This example shows how to map DSCP values 0, 8, 16, 24, 32, 40, 48, and 50 to CoS value 0 and to display the map:

```
Switch(config)# mls qos map dscp-cos 0 8 16 24 32 40 48 50 to 0
Switch(config)# end
Switch# show mls qos maps dscp-cos
Dscp-cos map:
     d1 :  d2 0  1  2  3  4  5  6  7  8  9
     ---------------------------------------
      0 :    00 00 00 00 00 00 00 00 00 01
      1 :    01 01 01 01 01 01 00 02 02 02
      2 :    02 02 02 02 00 03 03 03 03 03
      3 :    03 03 00 04 04 04 04 04 04 04
      4 :    00 05 05 05 05 05 05 05 00 06
      5 :    00 06 06 06 06 06 07 07 07 07
      6 :    07 07 07 07
```

Note In the above DSCP-to-CoS map, the CoS values are shown in the body of the matrix. The d1 column specifies the most-significant digit of the DSCP; the d2 row specifies the least-significant digit of the DSCP. The intersection of the d1 and d2 values provides the CoS value. For example, in the DSCP-to-CoS map, a DSCP value of 08 corresponds to a CoS value of 0.

This example shows how to define the DSCP-to-DSCP-mutation map. All the entries that are not explicitly configured are not modified (remains as specified in the null map):

```
Switch(config)# mls qos map dscp-mutation mutation1 1 2 3 4 5 6 7 to 0
Switch(config)# mls qos map dscp-mutation mutation1 8 9 10 11 12 13 to 10
Switch(config)# mls qos map dscp-mutation mutation1 20 21 22 to 20
Switch(config)# mls qos map dscp-mutation mutation1 30 31 32 33 34 to 30
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# mls qos trust dscp
Switch(config-if)# mls qos dscp-mutation mutation1
Switch(config-if)# end
Switch# show mls qos maps dscp-mutation mutation1
Dscp-dscp mutation map:
   mutation1:
     d1 :  d2 0  1  2  3  4  5  6  7  8  9
     ---------------------------------------
      0 :    00 00 00 00 00 00 00 00 10 10
      1 :    10 10 10 10 14 15 16 17 18 19
      2 :    20 20 20 23 24 25 26 27 28 29
      3 :    30 30 30 30 30 35 36 37 38 39
      4 :    40 41 42 43 44 45 46 47 48 49
      5 :    50 51 52 53 54 55 56 57 58 59
      6 :    60 61 62 63
```

Note In the above DSCP-to-DSCP-mutation map, the mutated values are shown in the body of the matrix. The d1 column specifies the most-significant digit of the original DSCP; the d2 row specifies the least-significant digit of the original DSCP. The intersection of the d1 and d2 values provides the mutated value. For example, a DSCP value of 12 corresponds to a mutated value of 10.

Related Topics

- Configuring the CoS-to-DSCP Map
- Configuring the IP-Precedence-to-DSCP Map
- Configuring the Policed-DSCP Map
- Configuring the DSCP-to-CoS Map
- Configuring the DSCP-to-DSCP-Mutation Map

Examples: Configuring Egress Queue Characteristics

This example shows how to configure bandwidth shaping on queue 1. Because the weight ratios for queues 2, 3, and 4 are set to 0, these queues operate in shared mode. The bandwidth weight for queue 1 is 1/8, which is 12.5 percent:

```
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# srr-queue bandwidth shape 8 0 0 0
```

This example shows how to configure the weight ratio of the SRR scheduler running on an egress port. Four queues are used, and the bandwidth ratio allocated for each queue in shared mode is 1/(1+2+3+4), 2/(1+2+3+4), 3/(1+2+3+4), and 4/(1+2+3+4), which is 10 percent, 20 percent, 30 percent, and 40 percent for queues 1, 2, 3, and 4. This means that queue 4 has four times the bandwidth of queue 1, twice the bandwidth of queue 2, and one-and-a-third times the bandwidth of queue 3.

```
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# srr-queue bandwidth share 1 2 3 4
```

This example shows how to enable the egress expedite queue when the SRR weights are configured. The egress expedite queue overrides the configured SRR weights.

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# srr-queue bandwidth shape 25 0 0 0
Switch(config-if)# srr-queue bandwidth share 30 20 25 25
Switch(config-if)# priority-queue out
Switch(config-if)# end
```

This example shows how to limit the bandwidth on a port to 80 percent:

```
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# srr-queue bandwidth limit 80
```

When you configure this command to 80 percent, the port is idle 20 percent of the time. The line rate drops to 80 percent of the connected speed, which is 800 Mb/s. These values are not exact because the hardware adjusts the line rate in increments of six.

Related Topics

Allocating Buffer Space to and Setting WTD Thresholds for an Egress Queue-Set, on page 34
Queueing and Scheduling on Egress Queues
Mapping DSCP or CoS Values to an Egress Queue and to a Threshold ID, on page 35
Configuring SRR Shaped Weights on Egress Queues, on page 38
Configuring SRR Shared Weights on Egress Queues, on page 40
Configuring the Egress Expedite Queue, on page 41
Limiting the Bandwidth on an Egress Interface, on page 43

Where to Go Next

Review the auto-QoS documentation to see if you can use these automated capabilities for your QoS configuration.