QoS: Classification Configuration Guide, Cisco IOS Release 15M&T

First Published: January 16, 2013

Last Modified: January 16, 2013

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS

CHAPTER 1

Classification Overview

Classifying network traffic allows you to organize traffic (that is, packets) into traffic classes or categories on the basis of whether the traffic matches specific criteria. Classifying network traffic (used in conjunction with marking network traffic) is the foundation for enabling many quality of service (QoS) features on your network.

Packet classification is pivotal to policy techniques that select packets traversing a network element or a particular interface for different types of QoS service. For example, you can use classification to mark certain packets for IP Precedence, and you can identify other packets as belonging to a Resource Reservation Protocol (RSVP) flow.

Methods of classification were once limited to use of the contents of the packet header. Current methods of marking a packet with its classification allow you to set information in the Layer 2, 3, or 4 headers, or even by setting information within the payload of a packet. Criteria for classification of a group might be as broad as "traffic destined for subnetwork X" or as narrow as a single flow. For more information about classifying network traffic, see the "Classifying Network Traffic"chapter.

This chapter explains IP Precedence, and then it gives a brief description of the kinds of traffic classification provided by the Cisco IOS QoS features. It discusses features described in the following sections:

Finding Feature Information

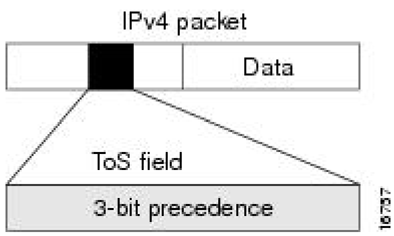
Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

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

About IP Precedence

Use of IP Precedence allows you to specify the class of service (CoS) for a packet. You use the three precedence bits in the type of service (ToS) field of the IP version 4 (IPv4) header for this purpose. The figure below shows the ToS field.

Figure 1: IPv4 Packet Type of Service Field



Using the ToS bits, you can define up to six classes of service. Other features configured throughout the network can then use these bits to determine how to treat the packet. These other QoS features can assign appropriate traffic-handling policies including congestion management strategy and bandwidth allocation. For example, although IP Precedence is not a queueing method, queueing methods such as weighted fair queueing (WFQ) and Weighted Random Early Detection (WRED) can use the IP Precedence setting of the packet to prioritize traffic.

By setting precedence levels on incoming traffic and using them in combination with the Cisco IOS QoS queueing features, you can create differentiated service. You can use features such as policy-based routing (PBR) and committed access rate (CAR) to set precedence based on extended access list classification. These features afford considerable flexibility for precedence assignment. For example, you can assign precedence based on application or user, or by destination and source subnetwork.

So that each subsequent network element can provide service based on the determined policy, IP Precedence is usually deployed as close to the edge of the network or the administrative domain as possible. You can think of IP Precedence as an edge function that allows core, or backbone, QoS features such as WRED to forward traffic based on CoS. IP Precedence can also be set in the host or network client, but this setting can be overridden by policy within the network.

The following QoS features can use the IP Precedence field to determine how traffic is treated:

- Distributed WRED (DWRED)
- WFQ
- CAR

How the IP Precedence Bits Are Used to Classify Packets

You use the three IP Precedence bits in the ToS field of the IP header to specify CoS assignment for each packet. You can partition traffic into a maximum of six classes and then use policy maps and extended access lists to define network policies for congestion handling and bandwidth allocation for each class.

For historical reasons, each precedence corresponds to a name. These names, which continue to evolve, are defined in RFC 791. The table below lists the numbers and their corresponding names, from least to most important.

Table 1: IP Precedence Values

Number	Name
0	routine
1	priority
2	immediate
3	flash
4	flash-override
5	critical
6	internet
7	network

However, the IP Precedence feature allows you considerable flexibility for precedence assignment. That is, you can define your own classification mechanism. For example, you might want to assign precedence based on application or access router.

Note IP Precedence bit settings 6 and 7 are reserved for network control information such as routing updates.

Setting or Changing the IP Precedence Value

By default, the Cisco IOS software leaves the IP Precedence value untouched, preserving the precedence value set in the header, allowing all internal network devices to provide service based on the IP Precedence setting. This policy follows the standard approach that stipulates that network traffic should be sorted into various types of service at the basic perimeter of the network and that those types of service should be implemented in the core of the network. Routers in the core of the network can then use the precedence bits, for example, to determine the order of transmission, the likelihood of packet drop, and so on.

Because traffic that enters your network can have precedence set by outside devices, we recommend that you reset the precedence for all traffic that enters your network. By controlling IP Precedence settings, you prohibit users that have already set the IP Precedence from acquiring better service for their traffic simply by setting a high precedence for all of their packets.

You can use CAR to set the IP Precedence in packets. As mentioned previously, after a packet has been classified, you can use other QoS features such as CAR and WRED to specify and enforce business policies to fit your business model.

Committed Access Rate

CAR is a multifaceted feature that implements both classification services and policing through rate limiting. This section describes the classification services of CAR. For information on CAR's rate limiting features, see the "Policing and Shaping Overview" chapter.

> **Note** In Cisco IOS Release 12.2 SR, the classification services of CAR are not supported on the Cisco 7600 series router.

You can use the classification services of CAR to set the IP Precedence for packets that enter the network. This capability of CAR allows you to partition your network into multiple priority levels or classes of service. Networking devices within your network can then use the adjusted IP Precedence to determine how to treat the traffic. For example, VIP-distributed WRED uses the IP Precedence to determine the probability a packet being dropped.

As discussed in the "About IP Precedence, on page 2" section, you can use the three precedence bits in the ToS field of the IP header to define up to six classes of service.

You can classify packets using policies based on physical port, source or destination IP or MAC address, application port, IP protocol type, or other criteria specifiable by access lists or extended access lists. You can classify packets by categories external to the network, for example, by a customer. After a packet has been classified, a network can either accept or override and reclassify the packet according to a specified policy. CAR includes commands that you can use to classify and reclassify packets.

CAR is supported on the majority of Cisco routers. Additionally, distributed CAR is supported on Cisco 7000 series routers with an RSP7000 interface processor or Cisco 7500 series routers with a VIP-based VIP2-40 or greater interface processor. A VIP2-50 interface processor is strongly recommended when the aggregate line rate of the port adapters on the VIP is greater than DS3. A VIP2-50 interface processor is required for OC-3 rates.

For information on how to configure CAR, see the "Configuring Committed Access Rate" chapter.

Classifying Network Traffic Using NBAR

Network-Based Application Recognition (NBAR) is a classification engine that recognizes and classifies a wide variety of protocols and applications. When NBAR recognizes and classifies a protocol or application, the network can be configured to apply the appropriate QoS for that application or traffic with that protocol.

For more information about NBAR, see the "Classifying Network Traffic Using NBAR" chapter.

Marking Network Traffic

Marking network traffic allows you to set or modify the attributes for traffic (that is, packets) belonging to a specific class or category. When used in conjunction with network traffic classification, network traffic marking is the foundation for enabling many QoS features on your network.

For more information about marking network traffic, see the "Marking Network Traffic" chapter.

CHAPTER 2

IPv6 Quality of Service

QoS features supported for IPv6 environments include packet classification, queueing, traffic shaping, weighted random early detection (WRED), class-based packet marking, and policing of IPv6 packets.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

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

Information About IPv6 Quality of Service

Implementation Strategy for QoS for IPv6

IPv6 packets are forwarded by paths that are different from those for IPv4. QoS features supported for IPv6 environments include packet classification, queuing, traffic shaping, weighted random early detection (WRED), class-based packet marking, and policing of IPv6 packets. These features are available at both the process switching and Cisco Express Forwarding switching paths of IPv6.

All of the QoS features available for IPv6 environments are managed from the modular QoS command-line interface (MQC). The MQC allows you to define traffic classes, create and configure traffic policies (policy maps), and then attach those traffic policies to interfaces.

To implement QoS in networks that are running IPv6, follow the same steps that you would follow to implement QoS in networks running only IPv4. At a very high level, the basic steps for implementing QoS are as follows:

- Know which applications in your network need QoS.
- Understand the characteristics of the applications so that you can make decisions about which QoS features would be appropriate.
- Know your network topology so that you know how link layer header sizes are affected by changes and forwarding.
- Create classes based on the criteria that you establish for your network. In particular, if the same network is also carrying IPv4 traffic along with IPv6 traffic, decide if you want to treat both of them the same way or treat them separately and specify match criteria accordingly. If you want to treat them the same, use match statements such as **match precedence**, **match dscp**, **set precedence**, and **set dscp**. If you want to treat them separately, add match criteria such as **match protocol ip** and **match protocol ipv6** in a match-all class map.
- Create a policy to mark each class.
- Work from the edge toward the core in applying QoS features.
- Build the policy to treat the traffic.
- Apply the policy.

Packet Classification in IPv6

Packet classification is available with both the process and Cisco Express Forwarding switching path. Classification can be based on IPv6 precedence, differentiated services control point (DSCP), and other IPv6 protocol-specific values that can be specified in IPv6 access lists in addition to other non-IPv6 values such as COS, packet length, and QoS group. Once you determine which applications need QoS, you can create classes based on the characteristics of the applications. You can use a variety of match criteria to classify traffic. You can combine various match criteria to segregate, isolate, and differentiate traffic.

The enhancements to the modular QoS CLI (MQC) allow you to create matches on precedence, DSCP, and IPv6 access group values in both IPv4 and IPv6 packets. The **match** command allows matches to be made on DSCP values and precedence for both IPv4 and IPv6 packets.

How to Configure IPv6 Quality of Service

Classifying Traffic in IPv6 Networks

The **set cos** and **match cos** commands for 802.1Q (dot1Q) interfaces are supported only for packets that are switched by Cisco Express Forwarding. Packets that are process-switched, such as device-generated packets, are not marked when these options are used.

Specifying Marking Criteria for IPv6 Packets

Perform this task to establish the match criteria to be used to match packets for classifying network traffic.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy map** *policy-map-name*
4. **class** {*class-name* | **class-default**}
5. Do one of the following:
   - **set precedence** {*precedence-value* | *from-field* [**table** *table-map-name*]}
   - **set** [**ip**] **dscp**{*dscp-value* | *from-field* [**table** *table-map-name*]}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**policy map** *policy-map-name* **Example:** `Device(config)# policy map policy1`	Creates a policy map using the specified name and enters QoS policy-map configuration mode. • Enter the name of the policy map that you want to create.
Step 4	**class** {*class-name* | **class-default**} **Example:** `Device(config-pmap)# class class-default`	Specifies the treatment for traffic of a specified class (or the default class) and enters QoS policy-map class configuration mode.
Step 5	Do one of the following: • **set precedence** {*precedence-value* | *from-field* [**table** *table-map-name*]}	Sets the precedence value and the DSCP value based on the CoS value (and action) defined in the specified table map. Both precedence and DSCP cannot be changed in the same packets.

	Command or Action	Purpose

	• **set** [**ip**] **dscp**{*dscp-value* | *from-field* [**table** *table-map-name*]}
	
	Example:
	`Device(config-pmap-c)#`
	`set precedence cos table table-map1`
	
	Example:
	`Device(config-pmap-c)#`
	`set dscp cos table table-map1`	

Using the Match Criteria to Manage IPv6 Traffic Flows

You can use multiple match statements. Depending on the type of class, you can specify whether to match all classes or any of the classes.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map** {*class-name*| **class-default**}
4. Do one of the following:
   - **match precedence** *precedence-value* [*precedence-value precedence-value*]
   - **match access-group name** *ipv6-access-group*
   - **match [ip] dscp** *dscp-value* [*dscp-value dscp-value dscp-value dscp-value dscp-value dscp-value dscp-value*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables such as privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**class-map** {*class-name*| **class-default**} **Example:** `Router(config-pmap-c)# class clsl`	Creates the specified class and enters QoS class-map configuration mode.

Step 4	Do one of the following:
	• **match precedence** *precedence-value* [*precedence-value precedence-value*]
	• **match access-group name** *ipv6-access-group*
	• **match [ip] dscp** *dscp-value* [*dscp-value dscp-value dscp-value dscp-value dscp-value dscp-value dscp-value*
	
	Example:
	
	`Router(config-pmap-c)#`
	
	`match precedence 5`
	
	Example:
	
	`Router(config-pmap-c)# match ip dscp 15`
		Matches the precedence value. The precedence applies to both IPv4 and IPv6 packets.
		or
		Specifies the name of an IPv6 access list against whose contents packets are checked to determine if they belong to the traffic class.
		or
		Identifies a specific IP DSCP value as a match criterion.

Configuration Examples for IPv6 Quality of Service

Example: Verifying Cisco Express Forwarding Switching

The following is sample output from the **show cef interface detail** command for Ethernet interface 1/0. Use this command to verify that Cisco Express Forwarding switching is enabled for policy decisions to occur. Notice that the display shows that Cisco Express Forwarding switching is enabled.

```
Router# show cef interface Ethernet 1/0 detail

Ethernet1/0 is up (if_number 9)
  Corresponding hwidb fast_if_number 9
  Corresponding hwidb firstsw->if_number 9
  Internet address is 10.2.61.8/24
```

```
  ICMP redirects are always sent
  Per packet load-sharing is disabled
  IP unicast RPF check is disabled
  Inbound access list is not set
  Outbound access list is not set
  IP policy routing is disabled
  Hardware idb is Ethernet1/0
  Fast switching type 1, interface type 5
  IP Distributed CEF switching enabled
  IP Feature Fast switching turbo vector
  IP Feature CEF switching turbo vector
  Input fast flags 0x0, Output fast flags 0x0
  ifindex 7(7)
  Slot 1 Slot unit 0 VC -1
  Transmit limit accumulator 0x48001A82 (0x48001A82)
  IP MTU 1500
```

Example: Verifying Packet Marking Criteria

The following example shows how to use the **match precedence** command to manage IPv6 traffic flows:

```
Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# class-m c1
Device(config-cmap)# match precedence 5
Device(config-cmap)# end
Device#
Device(config)# policy p1
Device(config-pmap)# class c1
Device(config-pmap-c)# police 10000 conform set-prec-trans 4
```

To verify that packet marking is working as expected, use the **show policy** command. The output of this command shows a difference between the number of total packets and the number of packets marked.

```
Device# show policy p1
  Policy Map p1
    Class c1
      police 10000 1500 1500 conform-action set-prec-transmit 4 exceed-action drop
Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# interface serial 4/1
Device(config-if)# service out p1
Device(config-if)# end
Device# show policy interface s4/1
 Serial4/1
  Service-policy output: p1
    Class-map: c1 (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: precedence 5
      police:
        10000 bps, 1500 limit, 1500 extended limit
        conformed 0 packets, 0 bytes; action: set-prec-transmit 4
        exceeded 0 packets, 0 bytes; action: drop
        conformed 0 bps, exceed 0 bps violate 0 bps
    Class-map: class-default (match-any)
      10 packets, 1486 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: any
```

During periods of transmit congestion at the outgoing interface, packets arrive faster than the interface can send them. It is helpful to know how to interpret the output of the **show policy-map interface** command, which is useful for monitoring the results of a service policy created with Cisco's MQC.

Congestion typically occurs when a fast ingress interface feeds a relatively slow egress interface. Functionally, congestion is defined as filling the transmit ring on the interface (a ring is a special buffer control structure). Every interface supports a pair of rings: a receive ring for receiving packets and a transmit ring for sending packets. The size of the rings varies with the interface controller and with the bandwidth of the interface or virtual circuit (VC). As in the following example, use the **show atm vc** *vcd* command to display the value of the transmit ring on a PA-A3 ATM port adapter.

```
Device# show atm vc 3

ATM5/0.2: VCD: 3, VPI: 2, VCI: 2
VBR-NRT, PeakRate: 30000, Average Rate: 20000, Burst Cells: 94
AAL5-LLC/SNAP, etype:0x0, Flags: 0x20, VCmode: 0x0
OAM frequency: 0 second(s)
PA TxRingLimit: 10
InARP frequency: 15 minutes(s)
Transmit priority 2
InPkts: 0, OutPkts: 0, InBytes: 0, OutBytes: 0
InPRoc: 0, OutPRoc: 0
InFast: 0, OutFast: 0, InAS: 0, OutAS: 0
InPktDrops: 0, OutPktDrops: 0
CrcErrors: 0, SarTimeOuts: 0, OverSizedSDUs: 0
OAM cells received: 0
OAM cells sent: 0
Status: UP
```

Cisco software (also referred to as the Layer 3 processor) and the interface driver use the transmit ring when moving packets to the physical media. The two processors collaborate in the following way:

- The interface sends packets according to the interface rate or a shaped rate.
- The interface maintains a hardware queue or transmit ring, where it stores the packets waiting for transmission onto the physical wire.
- When the hardware queue or transmit ring fills, the interface provides explicit back pressure to the Layer 3 processor system. It notifies the Layer 3 processor to stop dequeuing packets to the interface's transmit ring because the transmit ring is full. The Layer 3 processor now stores the excess packets in the Layer 3 queues.
- When the interface sends the packets on the transmit ring and empties the ring, it once again has sufficient buffers available to store the packets. It releases the back pressure, and the Layer 3 processor dequeues new packets to the interface.

The most important aspect of this communication system is that the interface recognizes that its transmit ring is full and throttles the receipt of new packets from the Layer 3 processor system. Thus, when the interface is congested, the drop decision is moved from a random, last-in, first-dropped decision in the first in, first out (FIFO) queue of the transmit ring to a differentiated decision based on IP-level service policies implemented by the Layer 3 processor.

Service policies apply only to packets stored in the Layer 3 queues. The table below illustrates which packets sit in the Layer 3 queue. Locally generated packets are always process-switched and are delivered first to the Layer 3 queue before being passed on to the interface driver. Fast-switched and CEF-switched packets are delivered directly to the transmit ring and sit in the L3 queue only when the transmit ring is full.

Table 2: Packet Types and the Layer 3 Queue

Packet Type	Congestion	Noncongestion
Locally generated packets, including Telnet packets and pings	Yes	Yes
Other packets that are process-switched	Yes	Yes

Packet Type	Congestion	Noncongestion
Packets that are CEF or fast-switched	Yes	No

The following example shows these guidelines applied to the **show policy-map interface** command output.

```
Device# show policy-map interface atm 1/0.1

ATM1/0.1: VC 0/100 -
 Service-policy output: cbwfq (1283)
   Class-map: A (match-all) (1285/2)
     28621 packets, 7098008 bytes

     5 minute offered rate 10000 bps, drop rate 0 bps
     Match: access-group 101 (1289)
     Weighted Fair Queueing
       Output Queue: Conversation 73
       Bandwidth 500 (kbps) Max Threshold 64 (packets)
       (pkts matched/bytes matched) 28621/7098008

       (depth/total drops/no-buffer drops) 0/0/0
   Class-map: B (match-all) (1301/4)

     2058 packets, 148176 bytes
     5 minute offered rate 0 bps, drop rate 0 bps
     Match: access-group 103 (1305)
     Weighted Fair Queueing
       Output Queue: Conversation 75
       Bandwidth 50 (kbps) Max Threshold 64 (packets)
       (pkts matched/bytes matched) 0/0
       (depth/total drops/no-buffer drops) 0/0/0
   Class-map: class-default (match-any) (1309/0)
     19 packets, 968 bytes
     5 minute offered rate 0 bps, drop rate 0 bps
     Match: any  (1313)
```

The table below defines counters that appear in the example.

Table 3: Packet Counters from show policy-map interface Output

Counter	Explanation
28621 packets, 7098008 bytes	The number of packets matching the criteria of the class. This counter increments whether or not the interface is congested.
(pkts matched/bytes matched) 28621/709800	The number of packets matching the criteria of the class when the interface was congested. In other words, the interface's transmit ring was full, and the driver and the L3 processor system worked together to queue the excess packets in the L3 queues, where the service policy applies. Packets that are process switched always go through the L3 queuing system and therefore increment the "packets matched" counter.
Class-map: B (match-all) (1301/4)	These numbers define an internal ID used with the CISCO-CLASS-BASED-QOS-MIB.

Counter	Explanation
5 minute offered rate 0 bps, drop rate 0 bps	Use the **load-interval** command to change this value and make it a more instantaneous value. The lowest value is 30 seconds; however, statistics displayed in the **show policy-map interface** command output are updated every 10 seconds. Because the command effectively provides a snapshot at a specific moment, the statistics may not reflect a temporary change in queue size.

Without congestion, there is no need to queue any excess packets. When congestion occurs, packets, including CEF and fast-switched packets, might go into the Layer 3 queue. If you use congestion management features, packets accumulating at an interface are queued until the interface is free to send them; they are then scheduled according to their assigned priority and the queueing mechanism configured for the interface.

Normally, the packets counter is much larger than the packets matched counter. If the values of the two counters are nearly equal, then the interface is receiving a large number of process-switched packets or is heavily congested. Both of these conditions should be investigated to ensure optimal packet forwarding.

Devices allocate conversation numbers for the queues that are created when the service policy is applied. The following example shows the queues and related information.

```
Device# show policy-map interface s1/0.1 dlci 100

      Serial1/0.1: DLCI 100 -
      output : mypolicy
       Class voice
        Weighted Fair Queueing
            Strict Priority
            Output Queue: Conversation 72

              Bandwidth 16 (kbps) Packets Matched 0
             (pkts discards/bytes discards) 0/0
       Class immediate-data
        Weighted Fair Queueing
            Output Queue: Conversation 73

              Bandwidth 60 (%) Packets Matched 0
              (pkts discards/bytes discards/tail drops) 0/0/0
              mean queue depth: 0
              drops: class  random   tail     min-th   max-th   mark-prob
                     0      0        0        64       128      1/10
                     1      0        0        71       128      1/10
                     2      0        0        78       128      1/10
                     3      0        0        85       128      1/10
                     4      0        0        92       128      1/10
                     5      0        0        99       128      1/10
                     6      0        0        106      128      1/10
                     7      0        0        113      128      1/10
                     rsvp   0        0        120      128      1/10
       Class priority-data
        Weighted Fair Queueing
            Output Queue: Conversation 74

              Bandwidth 40 (%) Packets Matched 0 Max Threshold 64 (packets)
              (pkts discards/bytes discards/tail drops) 0/0/0
       Class class-default
        Weighted Fair Queueing
            Flow Based Fair Queueing
            Maximum Number of Hashed Queues 64  Max Threshold 20 (packets)
```

Information reported for each class includes the following:

- Class definition
- Queueing method applied
- Output queue conversation number
- Bandwidth used
- Number of packets discarded
- Number of bytes discarded
- Number of packets dropped

The **class-default** class is the default class to which traffic is directed, if that traffic does not satisfy the match criteria of other classes whose policy is defined in the policy map. The **fair-queue** command allows you to specify the number of dynamic queues into which IP flows are sorted and classified. Alternately, devices allocate a default number of queues derived from the bandwidth on the interface or VC. Supported values in either case are a power of two, in a range from 16 to 4096.

The table below lists the default values for interfaces and for ATM permanent virtual circuits (PVCs).

Table 4: Default Number of Dynamic Queues as a Function of Interface Bandwidth

Bandwidth Range	Number of Dynamic Queues
Less than or equal to 64 kbps	16
More than 64 kbps and less than or equal to 128 kbps	32
More than 128 kbps and less than or equal to 256 kbps	64
More than 256 kbps and less than or equal to 512 kbps	128
More than 512 kbps	256

The table below lists the default number of dynamic queues in relation to ATM PVC bandwidth.

Table 5: Default Number of Dynamic Queues as a Function of ATM PVC Bandwidth

Bandwidth Range	Number of Dynamic Queues
Less than or equal to 128 kbps	16
More than 128 kbps and less than or equal to 512 kbps	32
More than 512 kbps and less than or equal to 2000 kbps	64

Bandwidth Range	Number of Dynamic Queues
More than 2000 kbps and less than or equal to 8000 kbps	128
More than 8000 kbps	256

Based on the number of reserved queues for WFQ, Cisco software assigns a conversation or queue number as shown in the table below.

Table 6: Conversation Numbers Assigned to Queues

Number	Type of Traffic
1 to 256	General flow-based traffic queues. Traffic that does not match to a user-created class will match to class-default and one of the flow-based queues.
257 to 263	Reserved for Cisco Discovery Protocol and for packets marked with an internal high-priority flag.
264	Reserved queue for the priority class (classes configured with the **priority** command). Look for the "Strict Priority" value for the class in the **show policy-map** interface output. The priority queue uses a conversation ID equal to the number of dynamic queues, plus 8.
265 and higher	Queues for user-created classes.

Example: Matching DSCP Value

The following example shows how to configure the service policy called priority50 and attach service policy priority50 to an interface. In this example, the **match dscp** command includes the optional **ip** keyword, meaning that the match is for IPv4 packets only. The class map called ipdscp15 will evaluate all packets entering interface Fast Ethernet 1/0. If the packet is an IPv4 packet and has a DSCP value of 15, the packet will be treated as priority traffic and will be allocated with bandwidth of 50 kbps.

```
Router(config)#
 class-map ipdscp15
Router(config-cmap)#
 match ip dscp 15
Router(config)#
 exit
Router(config)#
policy-map priority50
Router(config-pmap)#
 class ipdscp15
Router(config-pmap-c)#
```

```
priority 50
Router(config-pmap-c)#
exit
Router(config-pmap)#
 exit
Router(config)#
interface fa1/0
Router(config-if)#
service-policy input priority55
```

To match on IPv6 packets only, use the **match dscp** command without the **ip** keyword preceded by the **match protocol** command. Ensure that the class map has the **match-all** attribute (which is the default).

```
Router(config)#
 class-map ipdscp15
Router(config-cmap)#
 match protocol ipv6
Router(config-cmap)#
 match dscp 15
Router(config)#
 exit
```

To match packets on both IPv4 and IPv6 protocols, use the **match dscp** command:

```
Router(config)#
 class-map ipdscp15
Router(config-cmap)#
 match dscp 15
```

Additional References

Related Documents

Related Topic	Document Title
IPv6 addressing and connectivity	*IPv6 Configuration Guide*
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
IPv6 commands	Cisco IOS IPv6 Command Reference
Cisco IOS IPv6 features	Cisco IOS IPv6 Feature Mapping
Classifying Network Traffic	“Classifying Network Traffic” module
Marking Network Traffic	“Marking Network Traffic” module

Standards and RFCs

Standard/RFC	Title
RFC 2474	*Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers*
RFC 2475	*An Architecture for Differentiated Services Framework*
RFC 2597	*Assured Forwarding PHB*
RFC 2598	*An Expedited Forwarding PHB*
RFC 2697	*A Single Rate Three Color Marker*
RFC 2698	*A Two Rate Three Color Marker*
RFCs for IPv6	IPv6 RFCs

MIBs

MIB	MIBs Link
No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for IPv6 Quality of Service

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

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

Table 7: Feature Information for IPv6 Quality of Service

Feature Name	Releases	Feature Information
IPv6 Quality of Service	12.2(13)T 12.3 12.2(50)SG 3.2.0SG 15.0(2)SG 12.2(33)SRA 12.2(18)SXE Cisco IOS XE Release 2.1	QoS features supported for IPv6 environments include packet classification, queueing, traffic shaping, WRED, class-based packet marking, and policing of IPv6 packets. The following commands were introduced or modified: **match dscp**, **match precedence**, **set dscp**, **set precedence**. The following commands were introduced or modified: **match access-group name**, **match dscp**, **match precedence**, **set dscp**, **set precedence**.

CHAPTER 3

Configuring Committed Access Rate

This module describes the tasks for configuring committed access rate (CAR) and distributed CAR (DCAR).

Note In Cisco IOS Release 12.2 SR, CAR is not supported on the Cisco 7600 series router.

For complete conceptual information about these features, see the "Classification Overview"module and the "Policing and Shaping Overview" module.

For a complete description of the CAR commands in this module, see the Cisco IOS Quality of Service Solutions Command Reference. To locate documentation of other commands that appear in this module, use the command reference master index or search online.

Note CAR and DCAR can only be used with IP traffic. Non-IP traffic is not rate limited. CAR and DCAR can be configured on an interface or subinterface. However, CAR and DCAR are not supported on the Fast EtherChannel, tunnel, or PRI interfaces, nor on any interface that does not support Cisco Express Forwarding (CEF). CEF must be enabled on the interface before you configure CAR or DCAR. CAR is not supported for Internetwork Packet Exchange (IPX) packets.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

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

Committed Access Rate Configuration Task List

The CAR and DCAR services limit the input or output transmission rate on an interface or subinterface based on a flexible set of criteria. CAR is often configured on interfaces at the edge of a network to limit traffic into or out of the network.

CAR can rate limit traffic based on certain matching criteria, such as incoming interface, IP precedence, or IP access list. You configure the actions that CAR will take when traffic conforms to or exceeds the rate limit.

You can set CAR rate policies that are associated with one of the following:

- All IP traffic
- IP precedence
- MAC address
- IP access list, both standard and extended. Matching to IP access lists is more processor-intensive than matching based on other criteria.

Each interface can have multiple CAR policies, corresponding to different types of traffic. For example, low priority traffic may be limited to a lower rate than high-priority traffic. With multiple rate policies, the router examines each policy in the order entered until the packet matches. If a match is not found, the default action is to send.

The rate policies can be independent; each rate policy deals with a different type of traffic. Alternatively, rate policies can be cascading; a packet can be compared to multiple different rate policies in succession. You can configure up to 100 rate policies on a subinterface.

Note Because of the linear search for the matching rate-limit statement, the CPU load increases with the number of rate policies.

Basic CAR and DCAR functionality requires that the following criteria be defined:

- Packet direction, incoming or outgoing.
- An average rate, determined by a long-term average of the transmission rate. Traffic that falls under this rate will always conform.
- A normal burst size, which determines how large traffic bursts can be before some traffic is considered to exceed the rate limit.

- An excess burst size (Be). Traffic that falls between the normal burst size and the Excess Burst size exceeds the rate limit with a probability that increases as the burst size increases. CAR propagates bursts. It does no smoothing or shaping of traffic.

Table 8: Rate-Limit Command Action Keywords

Keyword	Description
continue	Evaluates the next **rate-limit** command.
drop	Drops the packet.
set-prec-continue *new-prec*	Sets the IP Precedence and evaluates the next **rate-limit** command.
set-prec-transmit *new-prec*	Sets the IP Precedence and sends the packet.
transmit	Sends the packet.

IP Precedence or MAC Address

Use the **access-list rate-limit** command to classify packets using either IP Precedence or MAC addresses. You can then apply CAR policies using the **rate-limit** command to individual rate-limited access lists. Packets with different IP precedences or MAC addresses are treated differently by the CAR service. See the section Example Rate Limiting in an IXP, on page 25 for an example of how to configure a CAR policy using MAC addresses.

IP Access List

Use the **access-list** command to define CAR policy based on an access list. The *acl-index* argument is an access list number. Use a number from 1 to 99 to classify packets by precedence or precedence mask. Use a number from 100 to 199 to classify by MAC address.

Note If an access list is not present, the **rate-limit** command will act as if no access list is defined and all traffic will be rate limited accordingly.

When you configure DCAR on Cisco 7000 series routers with RSP7000 or Cisco 7500 series routers with a VIP2-40 or greater interface processor, you can classify packets by group, to allow you to partition your network into multiple priority levels or classes of service. This classification is achieved by setting IP precedences based on different criteria for use by other QoS features such as Weighted Random Early Detection (WRED) or weighted fair queueing (WFQ).

Configuring CAR and DCAR for All IP Traffic

SUMMARY STEPS

1. Router(config)# **interface***interface-type interface-number*
2. Router(config-if)# **rate-limit** {**input** | **output**} *bps burst-normal burst-max* **conform-action** *action* **exceed-action** *action*

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router(config)# **interface***interface-type interface-number*	Specifies the interface or subinterface. This command puts the router in interface configuration mode.
Step 2	Router(config-if)# **rate-limit** {**input** | **output**} *bps burst-normal burst-max* **conform-action** *action* **exceed-action** *action*	Specifies a basic CAR policy for allConfiguring CAR and DCAR for All IP Traffic, on page 22ef"> Table 1 for a description of conform and exceed *action* keywords.

Configuring CAR and DCAR Policies

SUMMARY STEPS

1. Router(config-if)# **interface** *interface-type interface-number*
2. Router(config-if)# **rate-limit** {**input** | **output**} [**access-group** [**rate-limit**] *acl-index*] *bps burst-normal burst-max* **conform-action** *action* **exceed-action** *action*
3. Router(config-if) **exit**
4. Router(config)# **access-list rate-limit** *acl-index* {*precedence* | *mac-address*| **mask** *prec-mask*}
5. Do one of the following:
    - Router(config)# **access-list** *acl-index* {**deny** | **permit**} *source*[*source-wildcard*]
    - Router(config)# **access-list** *acl-index* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard*[**precedence** *precedence*][**tos** *tos*] [**log**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router(config-if)# **interface** *interface-type interface-number*	Specifies the interface or subinterface. This command puts the router in interface configuration mode.

	Command or Action	Purpose
Step 2	Router(config-if)# **rate-limit** {**input** | **output**} [**access-group** [**rate-limit**] *acl-index*] *bps burst-normal burst-max* **conform-action** *action* **exceed-action** *action*	Specifies the rate policy for each particular class of traffic. See Configuring CAR and DCAR Policies, on page 22 for a description of the **rate-limit** command action keywords. Repeat this command for each different class of traffic.
Step 3	Router(config-if) **exit**	(Optional) Returns to global configuration mode. **Note** This change in configuration mode is needed only if you complete optional Configuring CAR and DCAR Policies or Configuring CAR and DCAR Policies.
Step 4	Router(config)# **access-list rate-limit** *acl-index* {*precedence* | *mac-address*| **mask** *prec-mask*}	(Optional) Specifies a rate-limited access list. Repeat this command if you wish to specify a new access list.
Step 5	Do one of the following: • Router(config)# **access-list** *acl-index* {**deny** | **permit**} *source*[*source-wildcard*] • Router(config)# **access-list** *acl-index* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard*[**precedence** *precedence*][**tos** *tos*] [**log**]	(Optional) Specifies a standard or extended access list. Repeat this command to further configure the access list or specify a new access list.

Configuring a Class-Based DCAR Policy

SUMMARY STEPS

1. Router(config-if)# **interface** *interface-type interface-number*
2. Router(config-if)# **rate-limit** {**input** | **output**} [**access-group** [**rate-limit**] *acl-index*] *bps burst-normal burst-max* **conform-action** *action* **exceed-action** *action*
3. Router(config-if)# **random-detect precedence** *precedence min-threshold max-threshold mark-prob-denominator*
4. Do one of the following:
    - Router(config-if)# **access-list** *acl-index* {**deny** | **permit**} *source*[*source-wildcard*]
    - Router(config-if)# **access-list** *acl-index* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard*[**precedence** *precedence*] [**tos** *tos*] [**log**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	Router(config-if)# **interface** *interface-type interface-number*	Specifies the interface or subinterface. This command puts the router in interface configuration mode.
Step 2	Router(config-if)# **rate-limit** {**input** | **output**} [**access-group** [**rate-limit**] *acl-index*] *bps burst-normal burst-max* **conform-action** *action* **exceed-action** *action*	Specifies the rate policy for each particular class of traffic. See Configuring a Class-Based DCAR Policy, on page 23 for a description of the **rate-limit** command action keywords. Repeat this command for each different class of traffic.
Step 3	Router(config-if)# **random-detect precedence** *precedence min-threshold max-threshold mark-prob-denominator*	Configures WRED and specifies parameters for packets with specific IP Precedence.
Step 4	Do one of the following: • Router(config-if)# **access-list** *acl-index* {**deny** | **permit**} *source*[*source-wildcard*] • Router(config-if)# **access-list** *acl-index* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard*[**precedence** *precedence*] [**tos** *tos*] [**log**]	(Optional) Specifies a standard or extended access list. Repeat this command to further configure the access list or specify a new access list.

Monitoring CAR and DCAR

Command	Purpose
`Router#` **show access-lists**	Displays the contents of current IP and rate-limited access lists.
`Router#` **show access-lists rate-limit** [*access-list-number*]	Displays information about rate-limited access lists.
`Router#` **show interfaces** [*interface-type interface-number*] **rate-limit**	Displays information about an interface configured for CAR.

CAR and DCAR Configuration Examples

Example Subrate IP Services

The following example illustrates how to configure a basic CAR policy that allows all IP traffic. In the example, the network operator delivers a physical T3 link to the customer, but offers a less expensive 15 Mbps subrate service. The customer pays only for the subrate bandwidth, which can be upgraded with additional access bandwidth based on demand. The CAR policy limits the traffic rate available to the customer and delivered to the network to the agreed upon rate limit, plus the ability to temporarily burst over the limit.

```
interface hssi 0/0/0
rate-limit output 15000000 2812500 5625000 conform-action transmit exceed-action drop
ip address 10.1.0.9 255.255.255.0
```

Example Input and Output Rate Limiting on an Interface

In this example, a customer is connected to an Internet service provider (ISP) by a T3 link. The ISP wants to rate limit transmissions from the customer to 15 Mbps of the 45 Mbps. In addition, the customer is allowed to send bursts of 2,812,500 bytes. All packets exceeding this limit are dropped. The following commands are configured on the High-Speed Serial Interface (HSSI) of the ISP connected to the customer:

```
interface Hssi0/0/0
 description 45Mbps to R1
 rate-limit input 15000000 2812500 2812500 conform-action transmit exceed-action drop
 ip address 200.200.14.250 255.255.255.252
 rate-limit output 15000000 2812500 2812500 conform-action transmit exceed-action drop
```

The following sample output shows how to verify the configuration and monitor CAR statistics using the **show interfaces rate-limit** command:

```
Router# show interfaces hssi 0/0/0 rate-limit
Hssi0/0/0 45Mbps to R1
 Input
  matches: all traffic
   params: 15000000 bps, 2812500 limit, 2812500 extended limit
   conformed 8 packets, 428 bytes; action: transmit
   exceeded 0 packets, 0 bytes; action: drop
   last packet: 8680ms ago, current burst: 0 bytes
   last cleared 00:03:59 ago, conformed 0 bps, exceeded 0 bps
 Output
  matches: all traffic
   params: 15000000 bps, 2812500 limit, 2812500 extended limit
   conformed 0 packets, 0 bytes; action: transmit
   exceeded 0 packets, 0 bytes; action: drop
   last packet: 8680ms ago, current burst: 0 bytes
   last cleared 00:03:59 ago, conformed 0 bps, exceeded 0 bps
```

Example Rate Limiting in an IXP

The following example uses rate limiting to control traffic in an Internet Exchange Point (IXP). Because an IXP comprises many neighbors around an FDDI ring, MAC address rate-limited access lists are used to control traffic from a particular ISP. Traffic from one ISP (at MAC address 00e0.34b0.7777) is compared to a rate limit of 80 Mbps of the 100 Mbps available on the FDDI connection. Traffic that conforms to this rate is sent. Nonconforming traffic is dropped.

```
interface Fddi2/1/0
 rate-limit input access-group rate-limit 100 80000000 15000000 30000000 conform-action
transmit exceed-action drop
 ip address 200.200.6.1 255.255.255.0
!
access-list rate-limit 100 00e0.34b0.7777
```

The following sample output shows how to verify the configuration and monitor the CAR statistics using the **show interfaces rate-limit** command:

```
Router# show interfaces fddi2/1/0 rate-limit
Fddi2/1/0
 Input
  matches: access-group rate-limit 100
   params: 800000000 bps, 15000000 limit, 30000000 extended limit
   conformed 0 packets, 0 bytes; action: transmit
   exceeded 0 packets, 0 bytes; action: drop
   last packet: 4737508ms ago, current burst: 0 bytes
   last cleared 01:05:47 ago, conformed 0 bps, exceeded 0 bps
```

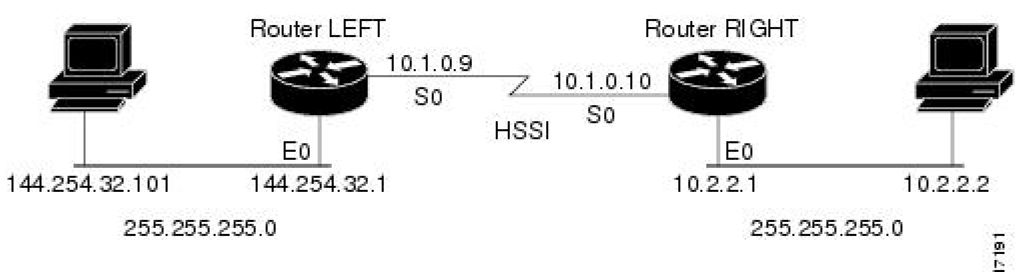
Example Rate Limiting by Access List

The following example shows how CAR can be used to limit the rate by application to ensure capacity for other traffic including mission-critical applications:

- All World Wide Web traffic is sent. However, the IP precedence for Web traffic that conforms to the first rate policy is set to 5. For nonconforming Web traffic, the IP precedence is set to 0 (best effort).
- File Transfer Protocol (FTP) traffic is sent with an IP precedence of 5 if it conforms to the second rate policy. If the FTP traffic exceeds the rate policy, it is dropped.
- Any remaining traffic is limited to 8 Mbps, with a normal burst size of 15,000 bytes and an Excess Burst size of 30,000 bytes. Traffic that conforms is sent with an IP precedence of 5. Traffic that does not conform is dropped.

The figure below illustrates the configuration. Notice that two access lists are created to classify the Web and FTP traffic so that they can be handled separately by CAR.

Figure 2: Rate Limiting by Access List



Router LEFT Configuration

```
interface Hssi0/0/0
description 45Mbps to R2
rate-limit output access-group 101 20000000 3750000 7500000 conform-action set-prec-
```

```
transmit 5 exceed-action set-prec-transmit 0
rate-limit output access-group 102 10000000 1875000 3750000 conform-action
set-prec-transmit 5 exceed-action drop
rate-limit output 8000000 1500000 3000000 conform-action set-prec-transmit 5
exceed-action drop
ip address 10.1.0.9 255.255.255.0
!
access-list 101 permit tcp any any eq www
access-list 102 permit tcp any any eq ftp
```

The following sample output shows how to verify the configuration and monitor CAR statistics using the **show interfaces rate-limit** command:

```
Router# show interfaces hssi 0/0/0 rate-limit
Hssi0/0/0 45Mbps to R2
 Input
  matches: access-group 101
   params: 20000000 bps, 3750000 limit, 7500000 extended limit
   conformed 3 packets, 189 bytes; action: set-prec-transmit 5
   exceeded 0 packets, 0 bytes; action: set-prec-transmit 0
   last packet: 309100ms ago, current burst: 0 bytes
   last cleared 00:08:00 ago, conformed 0 bps, exceeded 0 bps
  matches: access-group 102
   params: 10000000 bps, 1875000 limit, 3750000 extended limit
   conformed 0 packets, 0 bytes; action: set-prec-transmit 5
   exceeded 0 packets, 0 bytes; action: drop
   last packet: 19522612ms ago, current burst: 0 bytes
   last cleared 00:07:18 ago, conformed 0 bps, exceeded 0 bps
  matches: all traffic
   params: 8000000 bps, 1500000 limit, 3000000 extended limit
   conformed 5 packets, 315 bytes; action: set-prec-transmit 5
   exceeded 0 packets, 0 bytes; action: drop
   last packet: 9632ms ago, current burst: 0 bytes
   last cleared 00:05:43 ago, conformed 0 bps, exceeded 0 bps
```

CHAPTER 4

IPv6 QoS: MQC Packet Marking/Remarking

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

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

Information About IPv6 QoS: MQC Packet Marking/Remarking

Implementation Strategy for QoS for IPv6

IPv6 packets are forwarded by paths that are different from those for IPv4. QoS features supported for IPv6 environments include packet classification, queuing, traffic shaping, weighted random early detection (WRED), class-based packet marking, and policing of IPv6 packets. These features are available at both the process switching and Cisco Express Forwarding switching paths of IPv6.

All of the QoS features available for IPv6 environments are managed from the modular QoS command-line interface (MQC). The MQC allows you to define traffic classes, create and configure traffic policies (policy maps), and then attach those traffic policies to interfaces.

To implement QoS in networks that are running IPv6, follow the same steps that you would follow to implement QoS in networks running only IPv4. At a very high level, the basic steps for implementing QoS are as follows:

- Know which applications in your network need QoS.
- Understand the characteristics of the applications so that you can make decisions about which QoS features would be appropriate.
- Know your network topology so that you know how link layer header sizes are affected by changes and forwarding.
- Create classes based on the criteria that you establish for your network. In particular, if the same network is also carrying IPv4 traffic along with IPv6 traffic, decide if you want to treat both of them the same way or treat them separately and specify match criteria accordingly. If you want to treat them the same, use match statements such as **match precedence**, **match dscp**, **set precedence**, and **set dscp**. If you want to treat them separately, add match criteria such as **match protocol ip** and **match protocol ipv6** in a match-all class map.
- Create a policy to mark each class.
- Work from the edge toward the core in applying QoS features.
- Build the policy to treat the traffic.
- Apply the policy.

Policies and Class-Based Packet Marking in IPv6 Networks

You can create a policy to mark each class of traffic with appropriate priority values, using either DSCP or precedence. Class-based marking allows you to set the IPv6 precedence and DSCP values for traffic management. The traffic is marked as it enters the router on the ingress interface. The markings are used to treat the traffic (forward, queue) as it leaves the router on the egress interface. Always mark and treat the traffic as close as possible to its source.

Traffic Policing in IPv6 Environments

Congestion management for IPv6 is similar to IPv4, and the commands used to configure queueing and traffic shaping features for IPv6 environments are the same commands as those used for IPv4. Traffic shaping allows you to limit the packet dequeue rate by holding additional packets in the queues and forwarding them as specified by parameters configured for traffic shaping features. Traffic shaping uses flow-based queueing by default. CBWFQ can be used to classify and prioritize the packets. Class-based policer and generic traffic shaping (GTS) or Frame Relay traffic shaping (FRTS) can be used for conditioning and policing traffic.

How to Specify IPv6 QoS: MQC Packet Marking/Remarking

Specifying Marking Criteria for IPv6 Packets

Perform this task to establish the match criteria to be used to match packets for classifying network traffic.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy map** *policy-map-name*
4. **class** {*class-name* | **class-default**}
5. Do one of the following:
   - **set precedence** {*precedence-value* | *from-field* [**table** *table-map-name*]}
   - **set** [**ip**] **dscp**{*dscp-value* | *from-field* [**table** *table-map-name*]}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**policy map** *policy-map-name* **Example:** `Device(config)# policy map policy1`	Creates a policy map using the specified name and enters QoS policy-map configuration mode. • Enter the name of the policy map that you want to create.
Step 4	**class** {*class-name* | **class-default**} **Example:** `Device(config-pmap)# class class-default`	Specifies the treatment for traffic of a specified class (or the default class) and enters QoS policy-map class configuration mode.
Step 5	Do one of the following: • **set precedence** {*precedence-value* | *from-field* [**table** *table-map-name*]} • **set** [**ip**] **dscp**{*dscp-value* | *from-field* [**table** *table-map-name*]}	Sets the precedence value and the DSCP value based on the CoS value (and action) defined in the specified table map. Both precedence and DSCP cannot be changed in the same packets.

	Command or Action	Purpose

	Example:
	`Device(config-pmap-c)#`
	`set precedence cos table table-map1`
	Example:
	`Device(config-pmap-c)#`
	`set dscp cos table table-map1`	

Configuration Examples for IPv6 QoS: MQC Packet Marking/Remarking

Example: Verifying Packet Marking Criteria

The following example shows how to use the **match precedence** command to manage IPv6 traffic flows:

```
Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# class-m c1
Device(config-cmap)# match precedence 5
Device(config-cmap)# end
Device#
Device(config)# policy p1
Device(config-pmap)# class c1
Device(config-pmap-c)# police 10000 conform set-prec-trans 4
```

To verify that packet marking is working as expected, use the **show policy** command. The output of this command shows a difference between the number of total packets and the number of packets marked.

```
Device# show policy p1
  Policy Map p1
    Class c1
      police 10000 1500 1500 conform-action set-prec-transmit 4 exceed-action drop
Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# interface serial 4/1
Device(config-if)# service out p1
Device(config-if)# end
Device# show policy interface s4/1
 Serial4/1
  Service-policy output: p1
    Class-map: c1 (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: precedence 5
      police:
        10000 bps, 1500 limit, 1500 extended limit
        conformed 0 packets, 0 bytes; action: set-prec-transmit 4
        exceeded 0 packets, 0 bytes; action: drop
        conformed 0 bps, exceed 0 bps violate 0 bps
```

```
    Class-map: class-default (match-any)
      10 packets, 1486 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: any
```

During periods of transmit congestion at the outgoing interface, packets arrive faster than the interface can send them. It is helpful to know how to interpret the output of the **show policy-map interface** command, which is useful for monitoring the results of a service policy created with Cisco's MQC.

Congestion typically occurs when a fast ingress interface feeds a relatively slow egress interface. Functionally, congestion is defined as filling the transmit ring on the interface (a ring is a special buffer control structure). Every interface supports a pair of rings: a receive ring for receiving packets and a transmit ring for sending packets. The size of the rings varies with the interface controller and with the bandwidth of the interface or virtual circuit (VC). As in the following example, use the **show atm vc** *vcd* command to display the value of the transmit ring on a PA-A3 ATM port adapter.

```
Device# show atm vc 3

ATM5/0.2: VCD: 3, VPI: 2, VCI: 2
VBR-NRT, PeakRate: 30000, Average Rate: 20000, Burst Cells: 94
AAL5-LLC/SNAP, etype:0x0, Flags: 0x20, VCmode: 0x0
OAM frequency: 0 second(s)
PA TxRingLimit: 10
InARP frequency: 15 minutes(s)
Transmit priority 2
InPkts: 0, OutPkts: 0, InBytes: 0, OutBytes: 0
InPRoc: 0, OutPRoc: 0
InFast: 0, OutFast: 0, InAS: 0, OutAS: 0
InPktDrops: 0, OutPktDrops: 0
CrcErrors: 0, SarTimeOuts: 0, OverSizedSDUs: 0
OAM cells received: 0
OAM cells sent: 0
Status: UP
```

Cisco software (also referred to as the Layer 3 processor) and the interface driver use the transmit ring when moving packets to the physical media. The two processors collaborate in the following way:

- The interface sends packets according to the interface rate or a shaped rate.
- The interface maintains a hardware queue or transmit ring, where it stores the packets waiting for transmission onto the physical wire.
- When the hardware queue or transmit ring fills, the interface provides explicit back pressure to the Layer 3 processor system. It notifies the Layer 3 processor to stop dequeuing packets to the interface's transmit ring because the transmit ring is full. The Layer 3 processor now stores the excess packets in the Layer 3 queues.
- When the interface sends the packets on the transmit ring and empties the ring, it once again has sufficient buffers available to store the packets. It releases the back pressure, and the Layer 3 processor dequeues new packets to the interface.

The most important aspect of this communication system is that the interface recognizes that its transmit ring is full and throttles the receipt of new packets from the Layer 3 processor system. Thus, when the interface is congested, the drop decision is moved from a random, last-in, first-dropped decision in the first in, first out (FIFO) queue of the transmit ring to a differentiated decision based on IP-level service policies implemented by the Layer 3 processor.

Service policies apply only to packets stored in the Layer 3 queues. The table below illustrates which packets sit in the Layer 3 queue. Locally generated packets are always process-switched and are delivered first to the Layer 3 queue before being passed on to the interface driver. Fast-switched and CEF-switched packets are delivered directly to the transmit ring and sit in the L3 queue only when the transmit ring is full.

Table 9: Packet Types and the Layer 3 Queue

Packet Type	Congestion	Noncongestion
Locally generated packets, including Telnet packets and pings	Yes	Yes
Other packets that are process-switched	Yes	Yes
Packets that are CEF or fast-switched	Yes	No

The following example shows these guidelines applied to the **show policy-map interface** command output.

```
Device# show policy-map interface atm 1/0.1

ATM1/0.1: VC 0/100 -
 Service-policy output: cbwfq (1283)
   Class-map: A (match-all) (1285/2)
     28621 packets, 7098008 bytes

     5 minute offered rate 10000 bps, drop rate 0 bps
     Match: access-group 101 (1289)
     Weighted Fair Queueing
       Output Queue: Conversation 73
       Bandwidth 500 (kbps) Max Threshold 64 (packets)
       (pkts matched/bytes matched) 28621/7098008

       (depth/total drops/no-buffer drops) 0/0/0
   Class-map: B (match-all) (1301/4)

     2058 packets, 148176 bytes
     5 minute offered rate 0 bps, drop rate 0 bps
     Match: access-group 103 (1305)
     Weighted Fair Queueing
       Output Queue: Conversation 75
       Bandwidth 50 (kbps) Max Threshold 64 (packets)
       (pkts matched/bytes matched) 0/0
       (depth/total drops/no-buffer drops) 0/0/0
   Class-map: class-default (match-any) (1309/0)
     19 packets, 968 bytes
     5 minute offered rate 0 bps, drop rate 0 bps
     Match: any  (1313)
```

The table below defines counters that appear in the example.

Table 10: Packet Counters from show policy-map interface Output

Counter	Explanation
28621 packets, 7098008 bytes	The number of packets matching the criteria of the class. This counter increments whether or not the interface is congested.

Counter	Explanation
(pkts matched/bytes matched) 28621/709800	The number of packets matching the criteria of the class when the interface was congested. In other words, the interface's transmit ring was full, and the driver and the L3 processor system worked together to queue the excess packets in the L3 queues, where the service policy applies. Packets that are process switched always go through the L3 queuing system and therefore increment the "packets matched" counter.
Class-map: B (match-all) (1301/4)	These numbers define an internal ID used with the CISCO-CLASS-BASED-QOS-MIB.
5 minute offered rate 0 bps, drop rate 0 bps	Use the **load-interval** command to change this value and make it a more instantaneous value. The lowest value is 30 seconds; however, statistics displayed in the **show policy-map interface** command output are updated every 10 seconds. Because the command effectively provides a snapshot at a specific moment, the statistics may not reflect a temporary change in queue size.

Without congestion, there is no need to queue any excess packets. When congestion occurs, packets, including CEF and fast-switched packets, might go into the Layer 3 queue. If you use congestion management features, packets accumulating at an interface are queued until the interface is free to send them; they are then scheduled according to their assigned priority and the queueing mechanism configured for the interface.

Normally, the packets counter is much larger than the packets matched counter. If the values of the two counters are nearly equal, then the interface is receiving a large number of process-switched packets or is heavily congested. Both of these conditions should be investigated to ensure optimal packet forwarding.

Devices allocate conversation numbers for the queues that are created when the service policy is applied. The following example shows the queues and related information.

```
Device# show policy-map interface s1/0.1 dlci 100

      Serial1/0.1: DLCI 100 -
      output : mypolicy
       Class voice
        Weighted Fair Queueing
            Strict Priority
            Output Queue: Conversation 72

              Bandwidth 16 (kbps) Packets Matched 0
             (pkts discards/bytes discards) 0/0
       Class immediate-data
        Weighted Fair Queueing
            Output Queue: Conversation 73

              Bandwidth 60 (%) Packets Matched 0
              (pkts discards/bytes discards/tail drops) 0/0/0
              mean queue depth: 0
              drops: class  random   tail     min-th   max-th   mark-prob
                     0      0        0        64       128      1/10
                     1      0        0        71       128      1/10
```

```
                  2      0        0        78       128      1/10
                  3      0        0        85       128      1/10
                  4      0        0        92       128      1/10
                  5      0        0        99       128      1/10
                  6      0        0        106      128      1/10
                  7      0        0        113      128      1/10
                  rsvp   0        0        120      128      1/10
    Class priority-data
     Weighted Fair Queueing
         Output Queue: Conversation 74

           Bandwidth 40 (%) Packets Matched 0 Max Threshold 64 (packets)
           (pkts discards/bytes discards/tail drops) 0/0/0
    Class class-default
     Weighted Fair Queueing
         Flow Based Fair Queueing
         Maximum Number of Hashed Queues 64  Max Threshold 20 (packets)
```

Information reported for each class includes the following:

- Class definition
- Queueing method applied
- Output queue conversation number
- Bandwidth used
- Number of packets discarded
- Number of bytes discarded
- Number of packets dropped

The **class-default** class is the default class to which traffic is directed, if that traffic does not satisfy the match criteria of other classes whose policy is defined in the policy map. The **fair-queue** command allows you to specify the number of dynamic queues into which IP flows are sorted and classified. Alternately, devices allocate a default number of queues derived from the bandwidth on the interface or VC. Supported values in either case are a power of two, in a range from 16 to 4096.

The table below lists the default values for interfaces and for ATM permanent virtual circuits (PVCs).

Table 11: Default Number of Dynamic Queues as a Function of Interface Bandwidth

Bandwidth Range	Number of Dynamic Queues
Less than or equal to 64 kbps	16
More than 64 kbps and less than or equal to 128 kbps	32
More than 128 kbps and less than or equal to 256 kbps	64
More than 256 kbps and less than or equal to 512 kbps	128
More than 512 kbps	256

The table below lists the default number of dynamic queues in relation to ATM PVC bandwidth.

Table 12: Default Number of Dynamic Queues as a Function of ATM PVC Bandwidth

Bandwidth Range	Number of Dynamic Queues
Less than or equal to 128 kbps	16
More than 128 kbps and less than or equal to 512 kbps	32
More than 512 kbps and less than or equal to 2000 kbps	64
More than 2000 kbps and less than or equal to 8000 kbps	128
More than 8000 kbps	256

Based on the number of reserved queues for WFQ, Cisco software assigns a conversation or queue number as shown in the table below.

Table 13: Conversation Numbers Assigned to Queues

Number	Type of Traffic
1 to 256	General flow-based traffic queues. Traffic that does not match to a user-created class will match to class-default and one of the flow-based queues.
257 to 263	Reserved for Cisco Discovery Protocol and for packets marked with an internal high-priority flag.
264	Reserved queue for the priority class (classes configured with the **priority** command). Look for the "Strict Priority" value for the class in the **show policy-map** interface output. The priority queue uses a conversation ID equal to the number of dynamic queues, plus 8.
265 and higher	Queues for user-created classes.

Additional References

Related Documents

Related Topic	Document Title
IPv6 addressing and connectivity	*IPv6 Configuration Guide*
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
IPv6 commands	Cisco IOS IPv6 Command Reference
Cisco IOS IPv6 features	Cisco IOS IPv6 Feature Mapping
Marking Network Traffic	"Marking Network Traffic" module

Standards and RFCs

Standard/RFC	Title
RFCs for IPv6	IPv6 RFCs

MIBs

MIB	MIBs Link
	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for IPv6 QoS: MQC Packet Marking/Remarking

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

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

Table 14: Feature Information for IPv6 QoS: MQC Packet Marking/Remarking

Feature Name	Releases	Feature Information
IPv6 QoS: MQC Packet Marking/Remarking	12.0(28)S 12.2(33)SRA 12.2(18)SXE2 12.2(13)T 12.3 12.3(2)T 12.4 12.4(2)T	Class-based marking allows you to set the IPv6 precedence and DSCP values for traffic management.

CHAPTER 5

Marking Network Traffic

Marking network traffic allows you to set or modify the attributes for traffic (that is, packets) belonging to a specific class or category. When used in conjunction with network traffic classification, marking network traffic is the foundation for enabling many quality of service (QoS) features on your network. This module contains conceptual information and the configuration tasks for marking network traffic.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

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

Prerequisites for Marking Network Traffic

In order to mark network traffic, Cisco Express Forwarding must be configured on both the interface receiving the traffic and the interface sending the traffic.

Restrictions for Marking Network Traffic

•
•
•
•

Information About Marking Network Traffic

Purpose of Marking Network Traffic

Traffic marking is a method used to identify certain traffic types for unique handling, effectively partitioning network traffic into different categories.

After the network traffic is organized into classes by traffic classification, traffic marking allows you to mark (that is, set or change) a value (attribute) for the traffic belonging to a specific class. For instance, you may want to change the class of service (CoS) value from 2 to 1 in one class, or you may want to change the differentiated services code point (DSCP) value from 3 to 2 in another class. In this module, these values are referred to as attributes.

Attributes that can be set and modified include the following:

- Cell loss priority (CLP) bit
- CoS value of an outgoing packet
- Discard-class value
- Discard eligible (DE) bit setting in the address field of a Frame Relay frame
- DSCP value in the type of service (ToS) byte
- MPLS EXP field value in the topmost label on an input or output interface
- Multiprotocol Label Switching (MPLS) experimental (EXP) field on all imposed label entries
- Precedence value in the packet header
- QoS group identifier (ID)
- ToS bits in the header of an IP packet

Benefits of Marking Network Traffic

Improved Network Performance

Traffic marking allows you to fine-tune the attributes for traffic on your network. This increased granularity helps single out traffic that requires special handling and, thus, helps to achieve optimal application performance.

Traffic marking allows you to determine how traffic will be treated, based on how the attributes for the network traffic are set. It allows you to segment network traffic into multiple priority levels or classes of service based on those attributes, as follows:

- Traffic marking is often used to set the IP precedence or IP DSCP values for traffic entering a network. Networking devices within your network can then use the newly marked IP precedence values to determine how traffic should be treated. For example, voice traffic can be marked with a particular IP precedence or DSCP, and a queueing mechanism can then be configured to put all packets of that mark into a priority queue.
- Traffic marking can be used to identify traffic for any class-based QoS feature (any feature available in policy-map class configuration mode, although some restrictions exist).
- Traffic marking can be used to assign traffic to a QoS group within a device. The device can use the QoS groups to determine how to prioritize traffic for transmission. The QoS group value is usually used for one of the two following reasons:
    - To leverage a large range of traffic classes. The QoS group value has 100 different individual markings, as opposed to DSCP and IP precedence, which have 64 and 8, respectively.
    - If changing the IP precedence or DSCP value is undesirable.
- If a packet (for instance, in a traffic flow) that needs to be marked to differentiate user-defined QoS services is leaving a device and entering a switch, the device can set the CoS value of the traffic, because the switch can process the Layer 2 CoS header marking. Alternatively, the Layer 2 CoS value of the traffic leaving a switch can be mapped to the Layer 3 IP or MPLS value.

Two Methods for Marking Traffic Attributes

There are two methods for specifying and marking traffic attributes:

- You can specify and mark the traffic attribute by using a **set** command.

With this method, you configure individual **set** commands for the traffic attribute that you want to mark.

With this method, you configure the traffic attributes that you want to mark once in a table map and then the markings can be propagated throughout the network.

These methods are further described in the sections that follow.

Method One Using a set Command

You specify the traffic attribute that you want to change with a **set**command configured in a policy map. The table below lists the available **set**commands and the corresponding attribute. The table also includes the network layer and the network protocol typically associated with the traffic attribute.

Table 15: set Commands and Corresponding Traffic Attribute, Network Layer, and Protocol

set Commands[1]	Traffic Attribute	Network Layer	Protocol
set cos	Layer 2 CoS value of the outgoing traffic	Layer 2	

set Commands[1]	Traffic Attribute	Network Layer	Protocol
set discard-class	discard-class value	Layer 2	
set dscp	DSCP value in the ToS byte	Layer 3	IP
set fr-de	DE bit setting in the address field of a Frame Relay frame	Layer 2	
set ip tos (route-map)	ToS bits in the header of an IP packet	Layer 3	IP
set mpls experimental imposition	MPLS EXP field on all imposed label entries	Layer 3	MPLS
set mpls experimental topmost	MPLS EXP field value in the topmost label on either an input or an output interface	Layer 3	MPLS
set precedence	Precedence value in the packet header	Layer 3	IP
set qos-group	QoS group ID	Layer 3	IP, MPLS

1 Cisco set commands can vary by release. For more information, see the command documentation for the Cisco release that you are using

Method Two Using a Table Map

You can create a table map that can be used to mark traffic attributes. A table map is a kind of two-way conversion chart that lists and maps one traffic attribute to another. A table map supports a many-to-one type of conversion and mapping scheme. The table map establishes a to-from relationship for the traffic attributes and defines the change to be made to the attribute. That is, an attribute is set *to* one value that is taken *from* another value. The values are based on the specific attribute being changed. For instance, the Precedence attribute can be a number from 0 to 7, while the DSCP attribute can be a number from 0 to 63.

The following is a sample table map configuration:

```
table-map table-map1
map from 0 to 1
map from 2 to 3
exit
```

The table below lists the traffic attributes for which a to-from relationship can be established using the table map.

Table 16: Traffic Attributes for Which a To-From Relationship Can Be Established

The "To" Attribute	The "From" Attribute
Precedence	CoS
	QoS group
DSCP	CoS
	QoS group
CoS	Precedence
	DSCP
QoS group	Precedence
	DSCP
	MPLS EXP topmost
MPLS EXP topmost	QoS group
MPLS EXP imposition	Precedence
	DSCP

Once the table map is created, you configure a policy map to use the table map. In the policy map, you specify the table map name and the attributes to be mapped by using the **table** keyword and the *table-map-name* argument with one of the commands listed in the table below.

Table 17: Commands Used in Policy Maps to Map Attributes

Command Used in Policy Maps	Maps These Attributes
set cos dscp table *table-map-name*	CoS to DSCP
set cos precedence table *table-map-name*	CoS to Precedence
set dscp cos table *table-map-name*	DSCP to CoS
set dscp qos-group table *table-map-name*	DSCP to qos-group
set mpls experimental imposition dscp table *table-map-name*	MPLS EXP imposition to DSCP
set mpls experimental imposition precedence table *table-map-name*	MPLS EXP imposition to precedence

Command Used in Policy Maps	Maps These Attributes
set mpls experimental topmost qos-group table *table-map-name*	MPLS EXP topmost to QoS-group
set precedence cos table *table-map-name*	Precedence to CoS
set precedence qos-group table *table-map-name*	Precedence to QoS-group
set qos-group dscp table *table-map-name*	QoS-group to DSCP
set qos-group mpls exp topmost table *table-map-name*	QoS-group to MPLS EXP topmost
set qos-group precedence table *table-map-name*	QoS-group to Precedence

The following is an example of a policy map (policy2) configured to use the table map (table-map1) created earlier:

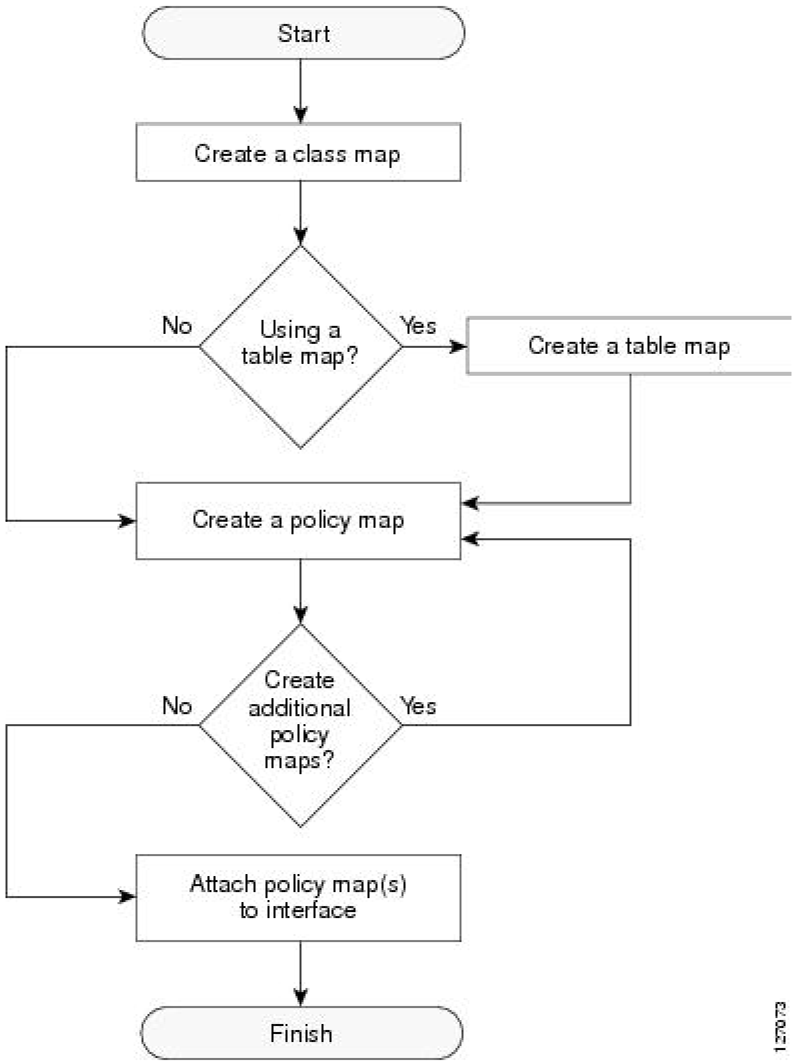
```
policy map policy2
class class-default
set cos dscp table table-map1
exit
```

In this example, a mapping relationship was created between the CoS attribute and the DSCP attribute as defined in the table map.

Traffic Marking Procedure Flowchart

The figure below illustrates the order of the procedures for configuring traffic marking.

Figure 3: Traffic Marking Procedure Flowchart



Method for Marking Traffic Attributes

You specify and mark the traffic attribute that you want to change by using a **set** command configured in a policy map.

With this method, you configure individual **set** commands for the traffic attribute that you want to mark.

Using a set Command

The table below lists the available **set** commands and the corresponding attribute. The table below also includes the network layer and the network protocol typically associated with the traffic attribute.

Table 18: set Commands and Corresponding Traffic Attribute, Network Layer, and Protocol

set Commands[2]	Traffic Attribute	Network Layer	Protocol
set cos	Layer 2 CoS value of the outgoing traffic	Layer 2	ATM, Frame Relay
set discard-class	discard-class value	Layer 2	ATM, Frame Relay
set dscp	DSCP value in the ToS byte	Layer 3	IP
set fr-de	DE bit setting in the address field of a Frame Relay frame	Layer 2	Frame Relay
set ip tos (route-map)	ToS bits in the header of an IP packet	Layer 3	IP
set mpls experimental imposition	MPLS EXP field on all imposed label entries	Layer 3	MPLS
set mpls experimental topmost	MPLS EXP field value in the topmost label on an input or output interface	Layer 3	MPLS
set precedence	Precedence value in the packet header	Layer 3	IP
set qos-group	QoS group ID	Layer 3	IP, MPLS

2 Cisco set commands can vary by release. For more information, see the command documentation.

If you are using individual **set** commands, those **set** commands are specified in a policy map. The following is a sample policy map configured with one of the **set** commands listed in the table above. In this sample configuration, the **set cos** command has been configured in the policy map (policy1) to mark the CoS value.

```
policy-map policy1
 class class1
  set cos 1
  end
```

For information on configuring a policy map, see the “Creating a Policy Map for Applying a QoS Feature to Network Traffic” section.

The final task is to attach the policy map to the interface. For information on attaching the policy map to the interface, see the “Attaching the Policy Map to an Interface” section.

MQC and Network Traffic Marking

To configure network traffic marking, you use the Modular QoS CLI (MQC).

The MQC is a CLI structure that allows you to complete the following tasks:

- Specify the matching criteria used to define a traffic class.
- Create a traffic policy (policy map). The traffic policy defines the QoS policy actions to be taken for each traffic class.
- Apply the policy actions specified in the policy map to an interface, subinterface, or ATM PVC by using the **service-policy** command.

Traffic Classification Compared with Traffic Marking

Traffic classification and traffic marking are closely related and can be used together. Traffic marking can be viewed as an additional action, specified in a policy map, to be taken on a traffic class.

Traffic classification allows you to organize into traffic classes on the basis of whether the traffic matches specific criteria. For example, all traffic with a CoS value of 2 is grouped into one class, and traffic with a DSCP value of 3 is grouped into another class. The match criteria are user-defined.

After the traffic is organized into traffic classes, traffic marking allows you to mark (that is, set or change) an attribute for the traffic belonging to that specific class. For instance, you may want to change the CoS value from 2 to 1, or you may want to change the DSCP value from 3 to 2.

The match criteria used by traffic classification are specified by configuring a **match** command in a class map. The marking action taken by traffic marking is specified by configuring a **set** command in a policy map. These class maps and policy maps are configured using the MQC.

The table below compares the features of traffic classification and traffic marking.

Table 19: Traffic Classification Compared with Traffic Marking

Feature	Traffic Classification	Traffic Marking
Goal	Groups network traffic into specific traffic classes on the basis of whether the traffic matches the user-defined criterion.	After the network traffic is grouped into traffic classes, modifies the attributes for the traffic in a particular traffic class.
Configuration Mechanism	Uses class maps and policy maps in the MQC.	Uses class maps and policy maps in the MQC.
CLI	In a class map, uses **match** commands (for example, **match cos**) to define the traffic matching criteria.	Uses the traffic classes and matching criteria specified by traffic classification. In addition, uses **set** commands (for example, **set cos**) in a policy map to modify the attributes for the network traffic.

How to Mark Network Traffic

Creating a Class Map for Marking Network Traffic

Note: The **match protocol** command is included in the steps below. The **match protocol** command is just an example of one of the **match** commands that can be used. See the command documentation for a complete list of **match** commands.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map** *class-map-name* [**match-all** | **match-any**]
4. **match protocol** *protocol-name*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**class-map** *class-map-name* [**match-all** | **match-any**] **Example:** `Device(config)# class-map class1`	Creates a class map to be used for matching traffic to a specified class and enters class-map configuration mode.
Step 4	**match protocol** *protocol-name* **Example:** `Device(config-cmap)# match protocol ftp`	(Optional) Configures the match criterion for a class map on the basis of the specified protocol. **Note** The **match protocol** command is just an example of one of the **match** commands that can be used. The **match** commands vary by Cisco release. See the command documentation for a complete list of **match** commands.

	Command or Action	Purpose
Step 5	**end** **Example:** `Device(config-cmap)# end`	(Optional) Returns to privileged EXEC mode.

Creating a Table Map for Marking Network Traffic

Note If you are not using a table map, skip this procedure and advance to the "Creating a Policy Map for Applying a QoS Feature to Network Traffic".

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **table-map** *table-map-name* **map from** *from-value* **to** *to-value* [**default** *default-action-or-value*]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**table-map** *table-map-name* **map from** *from-value* **to** *to-value* [**default** *default-action-or-value*] **Example:**	Creates a table map using the specified name and enters tablemap configuration mode. • Enter the name of the table map you want to create. • Enter each value mapping on a separate line. Enter as many separate lines as needed for the values you want to map.

	Command or Action	Purpose
	Example: `Device(config)# table-map table-map1 map from 2 to 1`	• The **default** keyword and *default-action-or-value* argument set the default value (or action) to be used if a value is not explicitly designated.

Step 4	**end**
	Example:
	`Device(config-tablemap)#`
	`end`
		(Optional) Exits tablemap configuration mode and returns to privileged EXEC mode.

Creating a Policy Map for Applying a QoS Feature to Network Traffic

Before You Begin

The following restrictions apply to creating a QoS policy map:

- A policy map containing the **set qos-group** command can only be attached as an input traffic policy. QoS group values are not usable for traffic leaving a device.
- A policy map containing the **set cos** command can only be attached as an output traffic policy.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map** *policy-map-name*
4. **class** {*class-name* | **class-default**}
5. **set cos** *cos-value*
6. **end**
7. **show policy-map**
8. **show policy-map** *policy-map* **class** *class-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: `Device> enable`	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**policy-map** *policy-map-name* **Example:** `Device(config)# policy-map policy1`	Specifies the name of the policy map and enters policy-map configuration mode.
Step 4	**class** {*class-name* | **class-default**} **Example:** `Device(config-pmap)# class class1`	Specifies the name of the class whose policy you want to create and enters policy-map class configuration mode. This class is associated with the class map created earlier.
Step 5	**set cos** *cos-value* **Example:** `Device(config-pmap-c)# set cos 2`	(Optional) Sets the CoS value in the type of service (ToS) byte. **Note** The **set cos** command is an example of one of the **set** commands that can be used when marking traffic. Other **set** commands can be used. For a list of other **set** commands, see "Information About Marking Network Traffic".
Step 6	**end** **Example:** `Device(config-pmap-c)# end`	Returns to privileged EXEC mode.
Step 7	**show policy-map** **Example:** `Device# show policy-map`	(Optional) Displays all configured policy maps.
Step 8	**show policy-map** *policy-map* **class** *class-name* **Example:** `Device# show policy-map policy1 class class1`	(Optional) Displays the configuration for the specified class of the specified policy map.

What to Do Next

Create and configure as many policy maps as you need for your network. To create and configure additional policy maps, repeat the steps in the "Creating a Policy Map for Applying a QoS Feature to Network Traffic" section. Then attach the policy maps to the appropriate interface, following the instructions in the "Attaching the Policy Map to an Interface" section.

Attaching the Policy Map to an Interface

Note

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number* [**name-tag**]
4. **pvc** [*name*] *vpi/vci* [**ilmi** | **qsaal** | **smds** | **l2transport**]
5. **exit**
6. **service-policy** {**input** | **output**} *policy-map-name*
7. **end**
8. **show policy-map interface** *type number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* [**name-tag**] **Example:** `Device(config)# interface serial4/0/0`	Configures an interface type and enters interface configuration mode.

	Command or Action	Purpose
Step 4	**pvc** [*name*] *vpi*/*vci* [**ilmi** | **qsaal** | **smds** | **l2transport**] **Example:** `Device(config-if)# pvc cisco 0/16`	(Optional) Creates or assigns a name to an ATM permanent virtual circuit (PVC), specifies the encapsulation type on an ATM PVC, and enters ATM virtual circuit configuration mode. **Note** This step is required only if you are attaching the policy map to an ATM PVC. If you are not attaching the policy map to an ATM PVC, advance to Step 6 below.
Step 5	**exit** **Example:** `Device(config-atm-vc)# exit`	(Optional) Returns to interface configuration mode. **Note** This step is required only if you are attaching the policy map to an ATM PVC and you completed Step 4 above. If you are not attaching the policy map to an ATM PVC, advance to Step 6 below.
Step 6	**service-policy** {**input** | **output**} *policy-map-name* **Example:** `Device(config-if)# service-policy input policy1`	Attaches a policy map to an input or output interface. **Note** Policy maps can be configured on ingress or egress devices. They can also be attached in the input or output direction of an interface. The direction (input or output) and the device (ingress or egress) to which the policy map should be attached vary according your network configuration. When using the **service-policy** command to attach the policy map to an interface, be sure to choose the device and the interface direction that are appropriate for your network configuration.
Step 7	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 8	**show policy-map interface** *type number* **Example:** `Device# show policy-map interface serial4/0/0`	(Optional) Displays the traffic statistics of all classes that are configured for all service policies either on the specified interface or subinterface or on a specific PVC on the interface.

Configuration Examples for Marking Network Traffic

Example: Creating a Class Map for Marking Network Traffic

The following is an example of creating a class map to be used for marking network traffic. In this example, a class called class1 has been created. Ttraffic with a protocol type of FTP will be put in this class.

```
Device> enable
Device# configure terminal
```

```
Device(config)# class-map class1
Device(config-cmap)# match protocol ftp
Device(config-cmap)# end
```

Example Creating a Policy Map for Applying a QoS Feature to Network Traffic

The following is an example of creating a policy map to be used for traffic classification. In this example, a policy map called policy1 has been created, and the **bandwidth** command has been configured for class1. The **bandwidth** command configures the QoS feature CBWFQ.

```
Router> enable
Router# configure terminal
Router(config)# policy-map policy1
Router(config-pmap)# class class1
Router(config-pmap-c)# bandwidth percent 50
Router(config-pmap-c)# end
Router#
show policy-map policy1 class class1
Router# exit
```

Note This example uses the **bandwidth** command. The **bandwidth** command configures the QoS feature class-based weighted fair queuing (CBWFQ). CBWFQ is just an example of a QoS feature that can be configured. Use the appropriate command for the QoS feature that you want to use.

Example: Attaching the Policy Map to an Interface

The following is an example of attaching the policy map to the interface. In this example, the policy map called policy1 has been attached in the input direction to the serial interface 4/0/0.

```
Device> enable
Device# configure terminal
Device(config)# interface serial4/0/0
Device(config-if)# service-policy input policy1
Device(config-if)# end
```

Additional References for Marking Network Traffic

Related Documents

Related Topic	Document Title
Cisco commands	Cisco IOS Master Commands List, All Releases
QoS commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco IOS Quality of Service Solutions Command Reference*
MQC	“Applying QoS Features Using the MQC” module

Related Topic	Document Title
Classifying network traffic	"Classifying Network Traffic" module

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Marking Network Traffic

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

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

Table 20: Feature Information for Marking Network Traffic

Feature Name	Software Releases	Feature Configuration Information
Class-Based Marking	12.2(2)T Cisco IOS XE Release 2.1 Cisco IOS XE Release 2.2 Cisco IOS XE Release 3.2SE	The Class-Based Packet Marking feature provides a user-friendly command-line interface (CLI) for efficient packet marking by which users can differentiate packets. This feature was implemented on Cisco ASR 1000 Series Routers. This feature was integrated into Cisco IOS XE Release 2.2.

Feature Name	Software Releases	Feature Configuration Information
Enhanced Packet Marking	12.2(13)T	The Enhanced Packet Marking feature allows you to map and convert the marking of a packet from one value to another by using a kind of conversion chart called a table map. The table map establishes an equivalency from one value to another. For example, the table map can map and convert the class of service (CoS) value of a packet to the precedence value of the packet. This value mapping can be propagated for use on the network, as needed.
QoS Packet Marking	12.2(8)T Cisco IOS XE Release 2.1 Cisco IOS XE Release 2.2 Cisco IOS XE Release 3.5S	The QoS Packet Marking feature allows you to mark packets by setting the IP precedence bit or the IP differentiated services code point (DSCP) in the Type of Service (ToS) byte, and to associate a local QoS group value with a packet. This feature was implemented on Cisco ASR 1000 Series Routers. This feature was integrated into Cisco IOS XE Software Release 2.2. In Cisco IOS XE Release 3.5S, support was added for the Cisco ASR 903 Router.
IP DSCP marking for Frame-Relay PVC	12.2(15)T Cisco IOS XE Release 2.1	This feature was implemented on Cisco ASR 1000 Series Routers.
PXF Based Frame Relay DE Bit Marking	12.2(31)SB2 15.0(1)S	PXF Based Frame Relay DE Bit Marking was integrated into the Cisco IOS Release 15.0(1)S release.

CHAPTER 6

QoS Tunnel Marking for GRE Tunnels

The QoS Tunnel Marking for GRE Tunnels feature introduces the capability to define and control the quality of service (QoS) for both incoming and outgoing customer traffic on the provider edge (PE) router in a service provider network.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

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

Prerequisites for QoS Tunnel Marking for GRE Tunnels

- You must determine the topology and interfaces that need to be configured to mark incoming and outgoing traffic.

Restrictions for QoS Tunnel Marking for GRE Tunnels

- GRE tunnel marking is not supported on the following paths:
    - IPsec tunnels
    - Multiprotocol Label Switching over generic routing encapsulation (MPLSoGRE)
    - Layer 2 Tunneling Protocol (L2TP)

Information About QoS Tunnel Marking for GRE Tunnels

GRE Definition

Generic routing encapsulation (GRE) is a tunneling protocol developed by Cisco that can encapsulate a wide variety of protocol packet types inside IP tunnels, creating a virtual point-to-point link to Cisco routers at remote points over an IP internetwork.

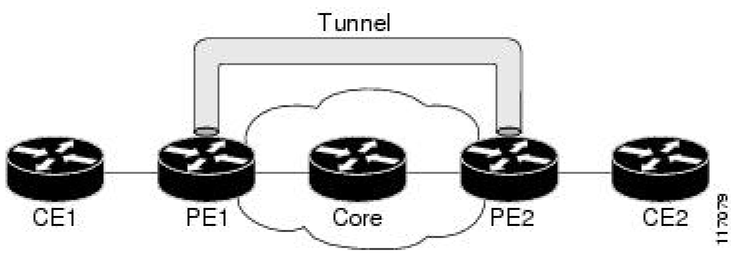
GRE Tunnel Marking Overview

The QoS Tunnel Marking for GRE Tunnels feature allows you to define and control QoS for incoming and outgoing customer traffic on the PE router in a service provider (SP) network. This feature lets you set (mark) either the IP precedence value or the differentiated services code point (DSCP) value in the header of an GRE tunneled packet. GRE tunnel marking can be implemented by a QoS marking command, such as **set ip** {**dscp** | **precedence**} [**tunnel**], and it can also be implemented in QoS traffic policing. This feature reduces administrative overhead previously required to control customer bandwidth by allowing you to mark the GRE tunnel header on the tunnel interface on the PE routers.

Note The **set ip** {**dscp** | **precedence**} [**tunnel**] command is equivalent to the **set** {**dscp** | **precedence**} [**tunnel**] command.

The figure below shows traffic being received from the CE1 router through the incoming interface on the PE1 router on which tunnel marking occurs. The traffic is encapsulated (tunneled), and the tunnel header is marked on the PE1 router. The marked packets travel (tunnel) through the core and are decapsulated automatically on the exit interface of the PE2 router. This feature is designed to simplify classifying customer edge (CE) traffic and is configured only in the service provider network. This process is transparent to the customer sites. The CE1 and CE2 routers exist as a single network.

Figure 4: Tunnel Marking



GRE Tunnel Marking and the MQC

Before you can configure tunnel marking for GRE tunnels, you must first configure a class map and a policy map and then attach that policy map to the appropriate interface. These three tasks can be accomplished by using the MQC.

For information on using the MQC, see the "Applying QoS Features Using the MQC" module.

GRE Tunnel Marking and DSCP or IP Precedence Values

GRE tunnel marking is configured with the **set ip precedence tunnel** or **set ip dscp tunnel** command on PE routers that carry incoming traffic from customer sites. GRE tunnel marking allows you to mark the header of a GRE tunnel by setting a DSCP value from 0 to 63 or an IP precedence value from 0 to 7 to control GRE tunnel traffic bandwidth and priority.

GRE traffic can also be marked under traffic policing with the **set-dscp-tunnel-transmit** and the **set-prec-tunnel-transmit** actions (or keywords) of the **police** command. The tunnel marking value is from 0 to 63 for the **set-dscp-tunnel-transmit** actions and from 0 to 7 for the **set-prec-tunnel-transmit** command. Under traffic policing, tunnel marking can be applied with conform, exceed, and violate action statements, allowing you to automatically apply a different value for traffic that does not conform to the expected traffic rate.

After the tunnel header is marked, GRE traffic is carried through the tunnel and across the service provider network. This traffic is decapsulated on the interface of the PE router that carries the outgoing traffic to the other customer site. The configuration of GRE tunnel marking is transparent to customer sites. All internal configuration is preserved.

There is a different between the **set ip precedence** and **set ip dscp** commands and the **set ip precedence tunnel** and **set ip dscp tunnel** commands:

- The **set ip precedence** and **set ip dscp** commands are used to set the IP precedence value or DSCP value in the header of an IP packet.
- The **set ip precedence tunnel** and **set ip dscp tunnel** commands are used to set (mark) the IP precedence value or DSCP value in the tunnel header that encapsulates the GRE traffic.
- The **set ip precedence tunnel** and **set ip dscp tunnel** commands have no effect on egress traffic that is not encapsulated in a GRE tunnel.

Benefits of GRE Tunnel Marking

GRE tunnel marking provides a simple mechanism to control the bandwidth of customer GRE traffic. The QoS Tunnel Marking for GRE Tunnels feature is configured entirely within the service provider network and on interfaces that carry incoming and outgoing traffic on the PE routers.

GRE Tunnel Marking and Traffic Policing

Traffic policing allows you to control the maximum rate of traffic sent or received on an interface and to partition a network into multiple priority levels or class of service (CoS). If you use traffic policing in your network, you can also implement the GRE tunnel marking feature with the **set-dscp-tunnel-transmit** or **set-prec-tunnel-transmit** action (or keyword) of the **police** command in policy-map class configuration mode. Under traffic policing, tunnel marking can be applied with conform, exceed, and violate action statements, allowing you to apply a different value automatically for traffic that does not conform to the expected traffic rate.

GRE Tunnel Marking Values

The range of the tunnel marking values for the **set ip dscp tunnel** and **set-dscp-tunnel-transmit** commands is from 0 to 63, and the range of values for the **set ip precedence tunnel** and **set-prec-tunnel-transmit** commands is from 0 to 7.

How to Configure Tunnel Marking for GRE Tunnels

Configuring a Class Map

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map** [**match-all** | **match-any**] *class-map-name*
4. **match ip precedence** *precedence-value*
5. **exit**
6. **class-map** [**match-all** | **match-any**] *class-map-name*
7. **match ip dscp** *dscp-value*
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: `Router> enable`	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**class-map** [**match-all** | **match-any**] *class-map-name* **Example:** `Router(config)# class-map match-any MATCH_PREC`	Specifies the name of the class map to be created and enters QoS class map configuration mode. • The class map defines the criteria to use to differentiate the traffic. For example, you can use the class map to differentiate voice traffic from data traffic, based on a series of match criteria defined using the **match** command. **Note** If the **match-all** or **match-any** keyword is not specified, traffic must match all the match criteria to be classified as part of the traffic class.
Step 4	**match ip precedence** *precedence-value* **Example:** `Router(config-cmap)# match ip precedence 0`	Enables packet matching on the basis of the IP precedence values you specify. **Note** You can enter up to four matching criteria, as number abbreviation (0 to 7) or criteria names (critical, flash, and so on), in a single match statement.
Step 5	**exit** **Example:** `Router(config-cmap)# exit`	Returns to global configuration mode.
Step 6	**class-map** [**match-all** | **match-any**] *class-map-name* **Example:** `Router(config)# class-map match-any MATCH_DSCP`	Specifies the name of the class map to be created and enters QoS class map configuration mode.
Step 7	**match ip dscp** *dscp-value* **Example:** `Router(config-cmap)# match ip dscp 0`	Enables packet matching on the basis of the DSCP values you specify. • This command is used by the class map to identify a specific DSCP value marking on a packet. • The treatment of these marked packets is defined by the user through the setting of QoS policies in policy-map class configuration mode.

	Command or Action	Purpose
Step 8	**end** **Example:** `Router(config-cmap)# end`	(Optional) Returns to privileged EXEC mode.

Creating a Policy Map

Perform this task to create a tunnel marking policy marp and apply the map to a specific interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map** *policy-map-name*
4. **class** {*class-name* | **class-default**}
5. **set ip precedence tunnel** *precedence-value*
6. **exit**
7. **class** {*class-name* | **class-default**}
8. **set ip dscp tunnel** *dscp-value*
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**policy-map** *policy-map-name* **Example:** `Router(config)# policy-map TUNNEL_MARKING`	Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy, and enters QoS policy-map configuration mode.

	Command or Action	Purpose
Step 4	**class** {*class-name* | **class-default**} **Example:** `Router(config-pmap)# class MATCH_PREC`	Specifies the name of the class whose policy you want to create or change or specifies the default class (commonly known as the class-default class) before you configure its policy. • Enters policy-map class configuration mode.
Step 5	**set ip precedence tunnel** *precedence-value* **Example:** `Router(config-pmap-c)# set ip precedence tunnel 3`	Sets the IP precedence value in the tunnel header of a GRE-tunneled packet on the ingress interface. The tunnel marking value is a number from 0 to 7 when IP precedence is configured.
Step 6	**exit** **Example:** `Router(config-pmap-c)# exit`	Returns to QoS policy-map configuration mode.
Step 7	**class** {*class-name* | **class-default**} **Example:** `Router(config-pmap)# class MATCH_DSCP`	Specifies the name of the class whose policy you want to create or change or specifies the default class (commonly known as the class-default class) before you configure its policy. • Enters policy-map class configuration mode.
Step 8	**set ip dscp tunnel** *dscp-value* **Example:** `Router(config-pmap-c)# set ip dscp tunnel 3`	Sets the differentiated services code point (DSCP) value in the tunnel header of a GRE-tunneled packet on the ingress interface. The tunnel marking value is a number from 0 to 63 when DSCP is configured.
Step 9	**end** **Example:** `Router(config-pmap-c)# end`	(Optional) Returns to privileged EXEC mode.

Attaching the Policy Map to an Interface or a VC

Policy maps can be attached to main interfaces, subinterfaces, or ATM permanent virtual circuits (PVCs). Policy maps are attached to interfaces by using the **service-policy** command and specifying either the **input** or **output** keyword to indicate the direction of the interface.

Note Tunnel marking policy can be applied on Ingress or Egress direction. A tunnel marking policy can be applied as an ingress policy on the ingress physical interface of a Service Provider Edge (SPE) router or as an egress policy on a tunnel interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **service-policy** {**input** | **output**} *policy-map-name*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface GigabitEthernet 0/0/1`	Configures an interface type and enters interface configuration mode.
Step 4	**service-policy** {**input** | **output**} *policy-map-name* **Example:** `Router(config-if)# service-policy input TUNNEL_MARKING`	Specifies the name of the policy map to be attached to the input or output direction of the interface. • Policy maps can be configured on ingress or egress routers. They can also be attached in the input or output direction of an interface. The direction (input or output) and the router (ingress or egress) to which the policy map should be attached vary according your network configuration.
Step 5	**end** **Example:** `Router(config-if)# end`	(Optional) Returns to privileged EXEC mode.

Verifying the Configuration of Tunnel Marking for GRE Tunnels

Use the **show** commands in this procedure to view the GRE tunnel marking configuration settings. The **show** commands are optional and can be entered in any order.

SUMMARY STEPS

1. **enable**
2. **show policy-map interface** *interface-name*
3. **show policy-map** *policy-map*
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**show policy-map interface** *interface-name* **Example:** `Router# show policy-map interface GigabitEthernet0/0/1`	(Optional) Displays the packet statistics of all classes that are configured for all service policies either on the specified interface or subinterface.
Step 3	**show policy-map** *policy-map* **Example:** `Router# show policy-map TUNNEL_MARKING`	(Optional) Displays the configuration of all classes for a specified service policy map or all classes for all existing policy maps.
Step 4	**exit** **Example:** `Router# exit`	(Optional) Returns to user EXEC mode.

Troubleshooting Tips

If you find that the configuration is not functioning as expected, perform these operations to troubleshoot the configuration:

- Use the **show running-config** command and analyze the output of the command.
- If the policy map does not appear in the output of the **show running-config** command, enable the **logging console** command.
- Attach the policy map to the interface again.

Configuration Examples for QoS Tunnel Marking for GRE Tunnels

Example: Configuring Tunnel Marking for GRE Tunnels

The following is an example of a GRE tunnel marking configuration. In this example, a class map called "MATCH_PREC" has been configured to match traffic based on the DSCP value.

```
Router> enable
Router# configure terminal
Router(config)# class-map MATCH_DSCP
Router(config-cmap)# match ip dscp 0
Router(config-cmap)# end
```

In the following part of the example configuration, a policy map called "TUNNEL_MARKING" has been created and the **set ip dscp tunnel** command has been configured in the policy map. You could use the **set ip precedence tunnel** command instead of the **set ip dscp tunnel** command if you do not use DSCP in your network.

```
Router(config)# policy-map TUNNEL_MARKING
Router(config-pmap)# class MATCH_DSCP
Router(config-pmap-c)# set ip dscp tunnel 3
Router(config-pmap-c)# end
```

Note The following part of the example configuration is not required to configure this feature if you use the **set ip dscp tunnel** or **set ip precedence tunnel** commands to enable GRE tunnel marking. This example shows how GRE tunnel marking can be enabled under traffic policing.

In the following part of the example configuration, the policy map called "TUNNEL_MARKING" has been created and traffic policing has also been configured by using the **police** command and specifying the appropriate policing actions. The **set-dscp-tunnel-transmit** command can be used instead of the **set-prec-tunnel-transmit** command if you use DSCP in your network.

```
Router(config)# policy-map TUNNEL_MARKING
Router(config-pmap)# class class-default
Router(config-pmap-c)# police 8000 conform-action set-prec-tunnel-transmit 4 exceed-action
 set-prec-tunnel-transmit 0
Router(config-pmap-c)# end
```

In the following part of the example configuration, the policy map is attached to GigabitEthernet interface 0/0/1 in the inbound (input) direction by specifying the **input** keyword of the **service-policy** command:

```
Router(config)# interface GigabitEthernet 0/0/1
Router(config-if)# service-policy input TUNNEL_MARKING
Router(config-if)# end
```

In the final part of the example configuration, the policy map is attached to tunnel interface 0 in the outbound (output) direction using the **output** keyword of the **service-policy** command:

```
Router(config)# interface Tunnel 0
Router(config-if)# service-policy output TUNNEL_MARKING
Router(config-if)# end
```

Example: Verifying the Tunnel Marking for GRE Tunnels Configuration

This section contains sample output from the **show policy-map interface** and the **show policy-map** commands. The output from these commands can be used to verify and monitor the feature configuration in your network.

The following is sample output from the **show policy-map interface** command. In this sample output:

- The character string "ip dscp tunnel 3" indicates that GRE tunnel marking has been configured to set the DSCP value in the header of a GRE-tunneled packet.
- The character string "ip precedence tunnel 3" indicates that GRE tunnel marking has been configured to set the precedence value in the header of a GRE-tunneled packet.

```
Router# show policy-map interface GigabitEthernet0/0/1
 Service-policy input: TUNNEL_MARKING

    Class-map: MATCH_PREC (match-any)
      22 packets, 7722 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: ip precedence 0
      QoS Set
        ip precedence tunnel 3
          Marker statistics: Disabled

    Class-map: MATCH_DSCP (match-any)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: ip dscp default (0)
      QoS Set
        ip dscp tunnel 3
          Marker statistics: Disabled

    Class-map: class-default (match-any)
      107 packets, 8658 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
```

The following is sample output from the **show policy-map** command. In this sample output, the character string "ip precedence tunnel 3" indicates that the GRE tunnel marking feature has been configured to set the IP precedence value in the header of an GRE-tunneled packet.

```
Router# show policy-map

Policy Map TUNNEL_MARKING
    Class MATCH_PREC
      set ip precedence tunnel 3
    Class MATCH_DSCP
      set ip dscp tunnel 3
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
QoS commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco IOS Quality of Service Solutions Command Reference*
MQC	"Applying QoS Features Using the MQC" module
Tunnel marking for Layer 2 Tunnel Protocol Version 3 (L2TPv3) tunnels	"QoS: Tunnel Marking for L2TPv3 Tunnels" module
DSCP	"Overview of DiffServ for Quality of Service" module

Standards

Standard	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	--

MIBs

MIB	MIBs Link
No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	--

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for QoS Tunnel Marking for GRE Tunnels

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

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

Table 21: Feature Information for QoS Tunnel Marking for GRE Tunnels

Feature Name	Releases	Feature Information
QoS Tunnel Marking for GRE Tunnels	Cisco IOS XE Release 3.5S	The QoS Tunnel Marking for GRE Tunnels feature introduces the capability to define and control the QoS for incoming customer traffic on the PE router in a service provider network. The following commands were introduced or modified: **match atm-clp**, **match cos**, **match fr-de**, **police**, **police (two rates)**, **set ip dscp tunnel**, **set ip precedence tunnel**, **show policy-map**, **show policy-map interface**.

CHAPTER 7

IPv6 QoS: MQC Packet Classification

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

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

Information About IPv6 QoS: MQC Packet Classification

Implementation Strategy for QoS for IPv6

IPv6 packets are forwarded by paths that are different from those for IPv4. QoS features supported for IPv6 environments include packet classification, queuing, traffic shaping, weighted random early detection (WRED), class-based packet marking, and policing of IPv6 packets. These features are available at both the process switching and Cisco Express Forwarding switching paths of IPv6.

All of the QoS features available for IPv6 environments are managed from the modular QoS command-line interface (MQC). The MQC allows you to define traffic classes, create and configure traffic policies (policy maps), and then attach those traffic policies to interfaces.

To implement QoS in networks that are running IPv6, follow the same steps that you would follow to implement QoS in networks running only IPv4. At a very high level, the basic steps for implementing QoS are as follows:

- Know which applications in your network need QoS.
- Understand the characteristics of the applications so that you can make decisions about which QoS features would be appropriate.
- Know your network topology so that you know how link layer header sizes are affected by changes and forwarding.
- Create classes based on the criteria that you establish for your network. In particular, if the same network is also carrying IPv4 traffic along with IPv6 traffic, decide if you want to treat both of them the same way or treat them separately and specify match criteria accordingly. If you want to treat them the same, use match statements such as **match precedence**, **match dscp**, **set precedence**, and **set dscp**. If you want to treat them separately, add match criteria such as **match protocol ip** and **match protocol ipv6** in a match-all class map.
- Create a policy to mark each class.
- Work from the edge toward the core in applying QoS features.
- Build the policy to treat the traffic.
- Apply the policy.

Packet Classification in IPv6

Packet classification is available with both the process and Cisco Express Forwarding switching path. Classification can be based on IPv6 precedence, differentiated services control point (DSCP), and other IPv6 protocol-specific values that can be specified in IPv6 access lists in addition to other non-IPv6 values such as COS, packet length, and QoS group. Once you determine which applications need QoS, you can create classes based on the characteristics of the applications. You can use a variety of match criteria to classify traffic. You can combine various match criteria to segregate, isolate, and differentiate traffic.

The enhancements to the modular QoS CLI (MQC) allow you to create matches on precedence, DSCP, and IPv6 access group values in both IPv4 and IPv6 packets. The **match** command allows matches to be made on DSCP values and precedence for both IPv4 and IPv6 packets.

How to Configure IPv6 QoS: MQC Packet Classification

Classifying Traffic in IPv6 Networks

The **set cos** and **match cos** commands for 802.1Q (dot1Q) interfaces are supported only for packets that are switched by Cisco Express Forwarding. Packets that are process-switched, such as device-generated packets, are not marked when these options are used.

Using the Match Criteria to Manage IPv6 Traffic Flows

You can use multiple match statements. Depending on the type of class, you can specify whether to match all classes or any of the classes.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map** {*class-name*| **class-default**}
4. Do one of the following:
   - **match precedence** *precedence-value* [*precedence-value precedence-value*]
   - **match access-group name** *ipv6-access-group*
   - **match [ip] dscp** *dscp-value* [*dscp-value dscp-value dscp-value dscp-value dscp-value dscp-value dscp-value*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables such as privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**class-map** {*class-name*| **class-default**} **Example:** `Router(config-pmap-c)# class clsl`	Creates the specified class and enters QoS class-map configuration mode.
Step 4	Do one of the following: • **match precedence** *precedence-value* [*precedence-value precedence-value*] • **match access-group name** *ipv6-access-group* • **match [ip] dscp** *dscp-value* [*dscp-value dscp-value dscp-value dscp-value dscp-value dscp-value dscp-value*	Matches the precedence value. The precedence applies to both IPv4 and IPv6 packets. or Specifies the name of an IPv6 access list against whose contents packets are checked to determine if they belong to the traffic class. or Identifies a specific IP DSCP value as a match criterion.

	Command or Action	Purpose

	Example:
	`Router(config-pmap-c)#`
	`match precedence 5`
	Example:
	`Router(config-pmap-c)# match ip dscp 15`	

Confirming the Service Policy

Ensure that the traffic flow matches the input or output parameter of the policy. For example, downloading a file from an FTP server generates congestion in the receive direction because the server sends large MTU-sized frames, and the client PC returns small acknowledgments (ACKs).

Before you begin this task, simulate congestion with an extended ping using a large ping size and a large number of pings. Also, try downloading a large file from an FTP server. The file constitutes "disturbing" data and fills the interface bandwidth.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number* **multipoint** | **point-to-point**
4. **ip address ip-address mask** [*secondary*]
5. **pvc** [*name*] *vpi* / *vci* [**ces** | **ilmi** | **qsaal** | **smds**]
6. **tx-ring-limit** *ring-limit*
7. **service-policy** {**input** | **output**} *policy-map-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **multipoint** | **point-to-point** **Example:** `Router(config)# interface ethernet1/1 point-to-point`	Enters interface configuration mode.
Step 4	**ip address ip-address mask** [*secondary*] **Example:** `Router(config-if)# ip address 10.1.1.1 255.255.255.0`	Specifies the IP address of the interface you want to test.

Step 5	**pvc** [*name*] *vpi* / *vci* [**ces** | **ilmi** | **qsaal** | **smds**]
	
	Example:
	
	`Router(config-if)#`
	
	`pvc cisco 0/5`
		Creates or assigns a name to an ATM PVC, optionally specifies the encapsulation type on an ATM PVC, and enters interface-ATM-VC configuration mode.

Step 6	**tx-ring-limit** *ring-limit* **Example:** `Router(config-if-atm-vc)# tx-ring-limit 10`	Reduces the size of the transmit ring of the interface. Lowering this value accelerates the use of the QoS in the Cisco IOS software. • Specify the ring limit as the number of packets for 2600 and 3600 series routers, or as the number of memory particles for 7200 and 7500 series routers.
Step 7	**service-policy** {**input** | **output**} *policy-map-name* **Example:** `Router(config-if-atm-vc)# service-policy output policy9`	Attaches a policy map to an input interface or VC, or an output interface or VC, to be used as the service policy for that interface or VC. • The packets-matched counter is a part of queueing feature and is available only on service policies attached in output direction.

Configuration Examples for IPv6 QoS: MQC Packet Classification

Example: Matching DSCP Value

The following example shows how to configure the service policy called priority50 and attach service policy priority50 to an interface. In this example, the **match dscp** command includes the optional **ip** keyword, meaning that the match is for IPv4 packets only. The class map called ipdscp15 will evaluate all packets entering interface Fast Ethernet 1/0. If the packet is an IPv4 packet and has a DSCP value of 15, the packet will be treated as priority traffic and will be allocated with bandwidth of 50 kbps.

```
Router(config)#
 class-map ipdscp15
Router(config-cmap)#
 match ip dscp 15
Router(config)#
 exit
Router(config)#
policy-map priority50
Router(config-pmap)#
 class ipdscp15
Router(config-pmap-c)#
priority 50
Router(config-pmap-c)#
exit
Router(config-pmap)#
 exit
Router(config)#
interface fa1/0
Router(config-if)#
service-policy input priority55
```

To match on IPv6 packets only, use the **match dscp** command without the **ip** keyword preceded by the **match protocol** command. Ensure that the class map has the **match-all** attribute (which is the default).

```
Router(config)#
 class-map ipdscp15
Router(config-cmap)#
 match protocol ipv6
Router(config-cmap)#
 match dscp 15
Router(config)#
 exit
```

To match packets on both IPv4 and IPv6 protocols, use the **match dscp** command:

```
Router(config)#
 class-map ipdscp15
Router(config-cmap)#
 match dscp 15
```

Additional References

Related Documents

Related Topic	Document Title
IPv6 addressing and connectivity	*IPv6 Configuration Guide*

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
IPv6 commands	Cisco IOS IPv6 Command Reference
Cisco IOS IPv6 features	Cisco IOS IPv6 Feature Mapping
Classifying Network Traffic	"Classifying Network Traffic" module

Standards and RFCs

Standard/RFC	Title
RFCs for IPv6	*IPv6 RFCs*

MIBs

MIB	MIBs Link
No new or modified MIBs are supported, and support for existing MIBs has not been modified.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for IPv6 QoS: MQC Packet Classification

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

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

Table 22: Feature Information for IPv6 QoS: MQC Packet Classification

Feature Name	Releases	Feature Information
IPv6 QoS: MQC Packet Classification	12.2(33)SRA 12.2(18)SXE2 12.2(13)T 12.3 12.3(2)T 12.4 12.4(2)T	The modular QoS CLI allows you to define traffic classes, create and configure traffic policies, and then attach those traffic policies to interfaces. The following commands were introduced or modified: **match access-group name**, **match dscp**, **match precedence**, **set dscp**, **set precedence**.

CHAPTER 8

Classifying Network Traffic

Classifying network traffic allows you to organize traffic (that is, packets) into traffic classes or categories on the basis of whether the traffic matches specific criteria. Classifying network traffic is the foundation for enabling many quality of service (QoS) features on your network. This module contains conceptual information and the configuration tasks for classifying network traffic.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

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

Information About Classifying Network Traffic

Purpose of Classifying Network Traffic

Classifying network traffic allows you to organize traffic (that is, packets) into traffic classes or categories on the basis of whether the traffic matches specific criteria. Classifying network traffic is the foundation for enabling other QoS features such as traffic shaping and traffic policing on your network.

The goal of network traffic classification is to group traffic based on user-defined criteria so that the resulting groups of network traffic can then be subjected to specific QoS treatments. The QoS treatments might include faster forwarding by intermediate routers and switches or reduced probability of the traffic being dropped due to lack of buffering resources.

Identifying and categorizing network traffic into traffic classes (that is, classifying packets) enables distinct handling for different types of traffic, effectively separating network traffic into different categories. This classification can be associated with a variety of match criteria such as the IP Precedence value, differentiated services code point (DSCP) value, class of service (CoS) value, source and destination MAC addresses, input interface, or protocol type. You classify network traffic by using class maps and policy maps with the Modular Quality of Service Command-Line Interface (MQC). For example, you can configure class maps and policy maps to classify network traffic on the basis of the QoS group, Frame Relay DLCI number, Layer 3 packet length, or other criteria that you specify.

Benefits of Classifying Network Traffic

Classifying network traffic allows you to see what kinds of traffic you have, organize the various kinds of network traffic into traffic classes, and treat some types of traffic differently than others. Identifying and organizing network traffic is the foundation for applying the appropriate QoS feature to that traffic, enabling you to allocate network resources to deliver optimal performance for different types of traffic. For example, high-priority network traffic or traffic matching specific criteria can be singled out for special handling, and thus, help to achieve peak application performance.

MQC and Network Traffic Classification

To configure network traffic classification, you use the Modular Quality of Service Command-Line Interface (MQC).

The MQC is a CLI structure that allows you to complete the following tasks:

- Specify the matching criteria used to define a traffic class.
- Create a traffic policy (policy map). The traffic policy defines the QoS policy actions to be taken for each traffic class.
- Apply the policy actions specified in the policy map to an interface, subinterface, or ATM permanent virtual circuit (PVC) by using the **service-policy** command.

Network Traffic Classification match Commands and Match Criteria

Network traffic classification allows you to group or categorize traffic on the basis of whether the traffic meets one or more specific criteria. For example, network traffic with a specific IP precedence can be placed into one traffic class, while traffic with a specific DSCP value can be placed into another traffic class. The network traffic within that traffic class can be given the appropriate QoS treatment, which you can configure in a policy map later.

You specify the criteria used to classify traffic with a **match** command. The table below lists the available **match** commands and the corresponding match criterion.

Table 23: match Commands and Corresponding Match Criterion

match Commands[3]	Match Criterion
match access group	Access control list (ACL) number
match any	Any match criteria
match atm clp	ATM cell loss priority (CLP)
match class-map	Traffic class name
match cos	Layer 2 class of service (CoS) value
match destination-address mac	MAC address
match discard-class	Discard class value
match dscp	DSCP value
match field	Fields defined in the protocol header description files (PHDFs)
match fr-de	Frame Relay discard eligibility (DE) bit setting
match fr-dlci	Frame Relay data-link connection identifier (DLCI) number
match input-interface	Input interface name
match ip rtp	Real-Time Transport Protocol (RTP) port
match mpls experimental	Multiprotocol Label Switching (MPLS) experimental (EXP) value
match mpls experimental topmost	MPLS EXP value in the topmost label
match not	Single match criterion value to use as an unsuccessful match criterion
match packet length (class-map)	Layer 3 packet length in the IP header
match port-type	Port type
match precedence	IP precedence values
match protocol	Protocol type
match protocol (NBAR)	Protocol type known to network-based application recognition (NBAR)

match Commands[3]	Match Criterion
match protocol citrix	Citrix protocol
match protocol fasttrack	FastTrack peer-to-peer traffic
match protocol gnutella	Gnutella peer-to-peer traffic
match protocol http	Hypertext Transfer Protocol
match protocol rtp	RTP traffic
match qos-group	QoS group value
match source-address mac	Source Media Access Control (MAC) address
match start	Datagram header (Layer 2) or the network header (Layer 3)
match tag (class-map)	Tag type of class map
match vlan (QoS)	Layer 2 virtual local-area network (VLAN) identification number

3 Cisco match commands can vary by release and platform. For more information, see the command documentation for the Cisco release and platform that you are using.

Traffic Classification Compared with Traffic Marking

Traffic classification and traffic marking are closely related and can be used together. Traffic marking can be viewed as an additional action, specified in a policy map, to be taken on a traffic class.

Traffic classification allows you to organize into traffic classes on the basis of whether the traffic matches specific criteria. For example, all traffic with a CoS value of 2 is grouped into one class, and traffic with a DSCP value of 3 is grouped into another class. The match criteria are user-defined.

After the traffic is organized into traffic classes, traffic marking allows you to mark (that is, set or change) an attribute for the traffic belonging to that specific class. For instance, you may want to change the CoS value from 2 to 1, or you may want to change the DSCP value from 3 to 2.

The match criteria used by traffic classification are specified by configuring a **match** command in a class map. The marking action taken by traffic marking is specified by configuring a **set** command in a policy map. These class maps and policy maps are configured using the MQC.

The table below compares the features of traffic classification and traffic marking.

Table 24: Traffic Classification Compared with Traffic Marking

Feature	Traffic Classification	Traffic Marking
Goal	Groups network traffic into specific traffic classes on the basis of whether the traffic matches the user-defined criterion.	After the network traffic is grouped into traffic classes, modifies the attributes for the traffic in a particular traffic class.
Configuration Mechanism	Uses class maps and policy maps in the MQC.	Uses class maps and policy maps in the MQC.
CLI	In a class map, uses **match** commands (for example, **match cos**) to define the traffic matching criteria.	Uses the traffic classes and matching criteria specified by traffic classification. In addition, uses **set** commands (for example, **set cos**) in a policy map to modify the attributes for the network traffic.

How to Classify Network Traffic

Creating a Class Map for Classifying Network Traffic

Note In the following task, the **matchfr-dlci**command is shown in Step 4. The **matchfr-dlci**command matches traffic on the basis of the Frame Relay DLCI number. The **matchfr-dlci**command is just an example of one of the **match** commands that can be used. For a list of other **match** commands, see the Network Traffic Classification match Commands and Match Criteria section.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map** *class-map-name* [**match-all**| **match-any**]
4. **match fr-dlci** *dlci-number*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: `Router> enable`	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**class-map** *class-map-name* [**match-all**| **match-any**] **Example:** `Router(config)# class-map class1`	Creates a class map to be used for matching traffic to a specified class, and enters class-map configuration mode. • Enter the class map name.
Step 4	**match fr-dlci** *dlci-number* **Example:** `Router(config-cmap)# match fr-dlci 500`	(Optional) Specifies the match criteria in a class map. **Note** The **matchfr-dlci** command classifies traffic on the basis of the Frame Relay DLCI number. The **matchfr-dlci**command is just an example of one of the **match** commands that can be used. For a list of other **match** commands, see the Network Traffic Classification match Commands and Match Criteria section.
Step 5	**end** **Example:** `Router(config-cmap)# end`	(Optional) Returns to privileged EXEC mode.

Creating a Policy Map for Applying a QoS Feature to Network Traffic

Note In the following task, the **bandwidth** command is shown at Creating a Policy Map for Applying a QoS Feature to Network Traffic. The **bandwidth** command configures the QoS feature class-based weighted fair queuing (CBWFQ). CBWFQ is just an example of a QoS feature that can be configured. Use the appropriate command for the QoS feature you want to use.

Note Configuring bandwidth on policies that have the class-default class is supported on physical interfaces such as Gigabit Ethernet (GigE), Serial, Mobile Location Protocol (MLP), and Multilink Frame-Relay (MFR).

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map** *policy-map-name*
4. **class** {*class-name* | **class-default**}
5. **bandwidth** {*bandwidth-kbps*| **remaining percent** *percentage*| **percent** *percentage*}
6. **end**
7. **show policy-map**
8.
9. **show policy-map** *policy-map* **class** *class-name*
10. Router# show policy-map
11.
12. Router# show policy-map policy1 class class1
13. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**policy-map** *policy-map-name* **Example:** `Router(config)# policy-map policy1`	Specifies the name of the policy map to be created and enters policy-map configuration mode. • Enter the policy map name.
Step 4	**class** {*class-name* | **class-default**} **Example:** `Router(config-pmap)# class class1`	Specifies the name of the class and enters policy-map class configuration mode. This class is associated with the class map created earlier. • Enter the name of the class or enter the **class-default**keyword.
Step 5	**bandwidth** {*bandwidth-kbps*| **remaining percent** *percentage*| **percent** *percentage*}	(Optional) Specifies or modifies the bandwidth allocated for a class belonging to a policy map.

	Command or Action	Purpose
	Example: `Router(config-pmap-c)# bandwidth percent 50`	• Enter the amount of bandwidth as a number of kbps, a relative percentage of bandwidth, or an absolute amount of bandwidth. **Note** The **bandwidth** command configures the QoS feature class-based weighted fair queuing (CBWFQ). CBWFQ is just an example of a QoS feature that can be configured. Use the appropriate command for the QoS feature that you want to use.
Step 6	**end** **Example:** `Router(config-pmap-c)# end`	Returns to privileged EXEC mode.
Step 7	**show policy-map**	(Optional) Displays all configured policy maps.
Step 8		or
Step 9	**show policy-map** *policy-map* **class** *class-name* **Example:**	(Optional) Displays the configuration for the specified class of the specified policy map. • Enter the policy map name and the class name.
Step 10	Router# show policy-map	
Step 11		
Step 12	Router# show policy-map policy1 class class1	
Step 13	**exit** **Example:** `Router# exit`	(Optional) Exits privileged EXEC mode.

What to Do Next

Create and configure as many policy maps as you need for your network. To create and configure additional policy maps, repeat the steps in the “Creating a Policy Map for Applying a QoS Feature to Network Traffic” section. Then attach the policy maps to the appropriate interface, following the instructions in the “Attaching the Policy Map to an Interface” section.

Attaching the Policy Map to an Interface

Note Depending on the needs of your network, policy maps can be attached to an interface, a subinterface, or an ATM PVC.

Note A policy with the command **match fr-dlic** can only be attached to a Frame Relay main interface with point-to-point connections.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number* [**name-tag**]
4. **pvc** [*name*] *vpi/vci* [**ilmi**|**qsaal**|**smds**| **l2transport**]
5. **exit**
6. **service-policy** {**input** | **output**}*policy-map-name*
7. **end**
8. **show policy-map interface** *type number*
9. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* [**name-tag**] **Example:** `Router(config)# interface serial4/0/0`	Configures an interface type and enters interface configuration mode. • Enter the interface type and number.

	Command or Action	Purpose
Step 4	**pvc** [*name*] *vpi*/*vci* [**ilmi**|**qsaal**|**smds**|**l2transport**] **Example:** `Router(config-if)# pvc cisco 0/16`	(Optional) Creates or assigns a name to an ATM PVC, specifies the encapsulation type on an ATM PVC, and enters ATM virtual circuit configuration mode. • Enter the PVC name, the ATM network virtual path identifier, and the network virtual channel identifier. **Note** This step is required only if you are attaching the policy map to an ATM PVC. If you are not attaching the policy map to an ATM PVC, advance to Attaching the Policy Map to an Interface.
Step 5	**exit** **Example:** `Router(config-atm-vc)# exit`	(Optional) Returns to interface configuration mode. **Note** This step is required only if you are attaching the policy map to an ATM PVC and you completed Attaching the Policy Map to an Interface. If you are not attaching the policy map to an ATM PVC, advance to Attaching the Policy Map to an Interface.
Step 6	**service-policy** {**input** | **output**}*policy-map-name* **Example:** `Router(config-if)# service-policy input policy1`	Attaches a policy map to an input or output interface. • Enter the policy map name. **Note** Policy maps can be configured on ingress or egress routers. They can also be attached in the input or output direction of an interface. The direction (input or output) and the router (ingress or egress) to which the policy map should be attached varies according your network configuration. When using the **service-policy** command to attach the policy map to an interface, be sure to choose the router and the interface direction that are appropriate for your network configuration.
Step 7	**end** **Example:** `Router(config-if)# end`	Returns to privileged EXEC mode.

Step 8	**show policy-map interface** *type number*
	
	Example:
	`Router#`
	`show policy-map interface serial4/0/0`
		(Optional) Displays the traffic statistics of all traffic classes that are configured for all service policies either on the specified interface or subinterface or on a specific PVC on the interface.
		• Enter the type and number.

Step 9	**exit** **Example:** `Router# exit`	(Optional) Exits privileged EXEC mode.

Configuration Examples for Classifying Network Traffic

Example Creating a Class Map for Classifying Network Traffic

The following is an example of creating a class map to be used for traffic classification. In this example, a traffic class called class1 has been created. Traffic with a Frame Relay DLCI value of 500 will be put in this traffic class.

```
Router> enable

Router# configure terminal

Router(config)# class-map class1

Router(config-cmap)# match fr-dlci 500

Router(config-cmap)# end
```

Note This example uses the **matchfr-dlci** command. The **matchfr-dlci** command is just an example of one of the **match** commands that can be used. For a list of other **match** commands, see Network Traffic Classification match Commands and Match Criteria.

A policy with match fr-dlic can only be attached to a Frame Relay main interface with point-to-point connections.

Example Creating a Policy Map for Applying a QoS Feature to Network Traffic

The following is an example of creating a policy map to be used for traffic classification. In this example, a policy map called policy1 has been created, and the **bandwidth** command has been configured for class1. The **bandwidth** command configures the QoS feature CBWFQ.

```
Router> enable
Router# configure terminal
Router(config)# policy-map policy1
Router(config-pmap)# class class1
Router(config-pmap-c)# bandwidth percent 50
Router(config-pmap-c)# end
Router#
show policy-map policy1 class class1
Router# exit
```

Note This example uses the **bandwidth** command. The **bandwidth** command configures the QoS feature class-based weighted fair queuing (CBWFQ). CBWFQ is just an example of a QoS feature that can be configured. Use the appropriate command for the QoS feature that you want to use.

Example Attaching the Policy Map to an Interface

The following is an example of attaching the policy map to an interface. In this example, the policy map called policy1 has been attached in the input direction of serial interface 4/0.

```
Router> enable
Router# configure terminal
Router(config)# interface serial4/0/0
Router(config-if)# service-policy input policy1
Router(config-if)# end
Router#
show policy-map interface serial4/0/0
Router# exit
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
QoS commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco IOS Quality of Service Solutions Command Reference*
MQC	"Applying QoS Features Using the MQC" module
Marking network traffic	"Marking Network Traffic" module
IPsec and VPNs	"Configuring Security for VPNs with IPsec" module
NBAR	"Classifying Network Traffic Using NBAR" module
IPv6 QoS	“IPv6 Quality of Service” module
IPv6 MQC Packet Classification	“IPv6 QoS: MQC Packet Classification” module

Standards

Standard	Title
No new or modified standards are supported, and support for existing standards has not been modified.	--

MIBs

MIB	MIBs Link
No new or modified MIBs are supported, and support for existing MIBs has not been modified.	To locate and download MIBs for selected platforms, Cisco IOS XE Software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
No new or modified RFCs are supported, and support for existing RFCs has not been modified.	--

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Classifying Network Traffic

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

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

Table 25: Feature Information for Classifying Network Traffic

Feature Name	Releases	Feature Information
Packet Classification Using Frame Relay DLCI Number	12.2(13)T Cisco IOS XE Release 2.1 Cisco IOS XE Release 3.12	The Packet Classification Using the Frame Relay DLCI Number feature allows customers to match and classify traffic based on the Frame Relay data-link connection identifier (DLCI) number associated with a packet. This new match criteria is in addition to the other match criteria, such as the IP Precedence, differentiated services code point (DSCP) value, class of service (CoS), currently available. The following commands were added or modified:**matchfr-dlci**
QoS: Local Traffic Matching Through MQC	Cisco IOS XE Release 2.1	This feature was introduced on Cisco ASR 1000 Series Routers.
QoS: Match ATM CLP	Cisco IOS XE Release 2.3	The QoS: Match ATM CLP features allows you to classify traffic on the basis of the ATM cell loss priority (CLP) value. The following command was introduced or modified: **matchatm-clp**.
QoS: MPLS EXP Bit Traffic Classification	Cisco IOS XE Release 2.3	The QoS: MPLS EXP Bit Traffic Classification feature allows you to classify traffic on the basis of the Multiprotocol Label Switching (MPLS) experimental (EXP) value. The following command was introduced or modified: **matchmplsexperimental**.

CHAPTER 9

Class-Based Ethernet CoS Matching and Marking

The Class-Based Ethernet CoS Matching and Marking (801.1p and ISL CoS) feature allows you to mark and match packets using Class of Service (CoS) values.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

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

Prerequisites for Class-Based Ethernet CoS Matching and Marking

When configuring this feature, you must first create a policy map (sometimes referred to as a service policy or a traffic policy) using the Modular QoS Command-Line Interface (CLI) (MQC). Therefore, you should be familiar with the procedure for creating a policy map using the MQC.

For more information about creating a policy map (traffic policy) using the MQC, see the “Applying QoS Features Using the MQC” module.

Information About Class-Based Ethernet CoS Matching and Marking

Layer 2 CoS Values

Layer 2 (L2) Class of Service (CoS) values are relevant for IEEE 802.1Q and Interswitch Link (ISL) types of frames. The Class-based Ethernet CoS Matching and Marking feature extends Cisco software capabilities to match packets by looking at the CoS value of the packet and marking packets with user-defined CoS values. This feature can be used for L2 CoS to L3 Terms of Service (TOS) mapping. CoS matching and marking can be configured via the Cisco Modular QoS CLI framework.

How to Configure Class-Based Ethernet CoS Matching and Marking

Configuring Class-Based Ethernet CoS Matching

In the following task, classes named voice and video-and-data are created to classify traffic based on the CoS values. The classes are configured in the CoS-based-treatment policy map, and the service policy is attached to all packets leaving Gigabit Ethernet interface 1/0/1.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map** *class-map-name*
4. **match cos** *cos-value*
5. **exit**
6. **class-map** *class-map-name*
7. **match cos** *cos-value*
8. **exit**
9. **policy-map** *policy-map-name*
10. **class** {*class-name* | **class-default**}
11. **priority level** *level*
12. **exit**
13. **class** {*class-name* | **class-default**}
14. **bandwidth remaining percent** *percentage*
15. **exit**
16. **exit**
17. **interface** *type number*
18. **service-policy** {**input**| **output**} *policy-map-name*
19. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**class-map** *class-map-name* **Example:** `Device(config)# class-map voice`	Specifies the name of the class map to be created and enters class-map configuration mode.

	Command or Action	Purpose
Step 4	**match cos** *cos-value* **Example:** `Device(config-cmap)# match cos 7`	Configures the class map to match traffic on the basis of the CoS value.
Step 5	**exit** **Example:** `Device(config-cmap)# exit`	(Optional) Exits class-map configuration mode.
Step 6	**class-map** *class-map-name* **Example:** `Device(config)# class-map video-and-data`	Specifies the name of the class map to be created and enters class-map configuration mode. • Enter the class map name.
Step 7	**match cos** *cos-value* **Example:** `Device(config-cmap)# match cos 5`	Configures the class map to match traffic on the basis of the CoS value.
Step 8	**exit** **Example:** `Device(config-cmap)# exit`	(Optional) Exits class-map configuration mode.
Step 9	**policy-map** *policy-map-name* **Example:** `Device(config)# policy-map cos-based-treatment`	Specifies the name of the policy map created earlier and enters policy-map configuration mode.
Step 10	**class** {*class-name* | **class-default**} **Example:** `Device(config-pmap)# class voice`	Specifies the name of the class whose policy you want to create and enters policy-map class configuration mode. This class is associated with the class map created earlier.
Step 11	**priority level** *level* **Example:** `Device(config-pmap-c)# priority level 1`	Specifies the level of the priority service.

	Command or Action	Purpose
Step 12	**exit** **Example:** `Device(config-pmap-c)# exit`	(Optional) Exits policy-map class configuration mode.
Step 13	**class** {*class-name* | **class-default**} **Example:** `Device(config-pmap)# class video-and-data`	Specifies the name of the class whose policy you want to create and enters policy-map class configuration mode. This class is associated with the class map created earlier.
Step 14	**bandwidth remaining percent** *percentage* **Example:** `Device(config-pmap-c)# bandwidth remaining percent 20`	Specifies the amount of bandwidth assigned to the class.
Step 15	**exit** **Example:** `Device(config-pmap-c)# exit`	(Optional) Exits policy-map class configuration mode.
Step 16	**exit** **Example:** `Device(config-pmap)# exit`	(Optional) Exits policy-map configuration mode.
Step 17	**interface** *type number* **Example:** `Device(config)# interface gigabitethernet 1/0/1`	Configures an interface (or subinterface) type and enters interface configuration mode.
Step 18	**service-policy** {**input**| **output**} *policy-map-name* **Example:** `Device(config-if)# service-policy output cos-based-treatment`	Specifies the name of the policy map to be attached to either the input or output direction of the interface. **Note** Policy maps can be configured on ingress or egress devices. They can also be attached in the input or output direction of an interface. The direction (input or output) and the device (ingress or egress) to which the policy map should be attached vary according your network configuration. When using the **service-policy** command to attach the policy map to an interface, be sure to choose the device and the interface direction that are appropriate for your network configuration.

	Command or Action	Purpose
Step 19	**end** **Example:** `Device(config-if)# end`	(Optional) Exits interface configuration mode and returns to privileged EXEC mode.

Configuring Class-Based Ethernet CoS Marking

In the following task, the policy map called cos-set is created to assign different CoS values for different types of traffic.

Note This task assumes that the class maps called voice and video-and-data have already been created.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map** *policy-map-name*
4. **class** {*class-name* | **class-default**}
5. **set cos** *cos-value*
6. **exit**
7. **class** {*class-name* | **class-default**}
8. **set cos** *cos-value*
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**policy-map** *policy-map-name* **Example:** `Device(config)# policy-map cos-set`	Specifies the name of the policy map created earlier and enters policy-map configuration mode.
Step 4	**class** {*class-name* | **class-default**} **Example:** `Device(config-pmap)# class voice`	Specifies the name of the class whose policy you want to create and enters policy-map class configuration mode. This class is associated with the class map created earlier.
Step 5	**set cos** *cos-value* **Example:** `Device(config-pmap-c)# set cos 1`	Sets the packet's CoS value.
Step 6	**exit** **Example:** `Device(config-pmap-c)# exit`	Exits policy-map class configuration mode.
Step 7	**class** {*class-name* | **class-default**} **Example:** `Device(config-pmap)# class video-and-data`	Specifies the name of the class whose policy you want to create and enters policy-map class configuration mode. This class is associated with the class map created earlier.
Step 8	**set cos** *cos-value* **Example:** `Device(config-pmap-c)# set cos 2`	Sets the packet's CoS value.
Step 9	**end** **Example:** `Device(config-pmap-c)# end`	(Optional) Exits policy-map class configuration mode and returns to privileged EXEC mode.

Configuration Examples for Class-Based Ethernet CoS Matching and Marking

Example: Configuring Class-Based Ethernet CoS Matching

This example creates two classes, voice and video-and-data, to classify traffic based on the CoS values. The CoS-based-treatment policy map is used to set priority and bandwidth values for the classes. The service policy is attached to all packets leaving interface Gigabit Ethernet1/0/1.

Note The service policy can be attached to any interface that supports service policies.

```
Device(config)# class-map voice
Device(config-cmap)# match cos 7
Device(config-cmap)# exit
Device(config)# class-map video-and-data
Device(config-cmap)# match cos 5
Device(config-cmap)# exit
Device(config)# policy-map cos-based-treatment
Device(config-pmap)# class voice
Device(config-pmap-c)# priority level 1
Device(config-pmap-c)# exit
Device(config-pmap)# class video-and-data
Device(config-pmap-c)# bandwidth remaining percent 20
Device(config-pmap-c)# exit
Device(config-pmap)# exit
Device(config)#  interface gigabitethernet1/0/1
Device(config-if)# service-policy output cos-based-treatment
```

Example: Class-Based Ethernet CoS Marking

```
Device(config)# policy-map cos-set
Device(config-pmap)# class voice
Device(config-pmap-c)# set cos 1
Device(config-pmap-c)# exit
Device(config-pmap)# class video-and-data
Device(config-pmap-c)# set cos 2
Device(config-pmap-c)# end
```

Additional References for Class-Based Ethernet CoS Matching and Marking

Related Documents

Related Topic	Document Title
Cisco commands	Cisco IOS Master Command List, All Releases

Related Topic	Document Title
QoS commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco IOS Quality of Service Solutions Command Reference*
Classifying network traffic	“Classifying Network Traffic” module
MQC	“Applying QoS Features Using the MQC” module
Marking network traffic	“Marking Network Traffic” module

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Class-Based Ethernet CoS Matching & Marking

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

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

Table 26: Feature Information for Class-Based Ethernet CoS Matching and Marking

Feature Name	Releases	Feature Information
Class-Based Ethernet CoS Matching and Marking	12.2(5)T 15.0(1)S Cisco IOS XE Release 2.1 Cisco IOS XE Release 3.2SE	This feature allows you to mark and match packets using Class of Service (CoS) values. The following commands were introduced or modified: **match cos**, **set cos**.

Feature Name	Releases	Feature Information
User Priority Based QoS Marking for Wireless Deployments	Cisco IOS XE Release 3.2SE	This features allows you to mark and match packets on wireless deployments using the user-priority (CoS) vlaues.

CHAPTER 10

Quality of Service for VPNs

The QoS for VPNs feature provides a solution for making Cisco IOS QoS services operate in conjunction with tunneling and encryption on an interface. Cisco IOS software can classify packets and apply the appropriate QoS service before the data is encrypted and tunneled. The QoS for VPN feature allows users to look inside the packet so that packet classification can be done based on original port numbers and based on source and destination IP addresses. This allows the service provider to treat mission critical or multi-service traffic with higher priority across their network.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

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

Information About Quality of Service for Virtual Private Networks

QoS for VPNs

The QoS for VPNs feature provides a solution for making Cisco IOS QoS services operate in conjunction with tunneling and encryption on an interface. Cisco IOS software can classify packets and apply the appropriate QoS service before the data is encrypted and tunneled. The QoS for VPN feature allows users to look inside the packet so that packet classification can be done based on original port numbers and based on source and destination IP addresses. This allows the service provider to treat mission critical or multi-service traffic with higher priority across their network.

How to Configure QoS for VPNs

Configuring QoS When Using IPsec VPNs

This task uses the **qos pre-classify** command to enable QoS preclassification for the packet. QoS preclassification is not supported for all fragmented packets. If a packet is fragmented, each fragment might received different preclassifications.

Note This task is required only if you are using IPsec Virtual Private Networks (VPNs). Otherwise, this task is not necessary. For information about IPsec VPNs, see the "Configuring Security for VPNs with IPsec" module.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto map** *map-name seq-num*
4. **exit**
5. **interface** *type number* [**name-tag**]
6. **qos pre-classify**
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: `Router> enable`	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**crypto map** *map-name seq-num* **Example:** `Router(config)# crypto map mymap 10`	Enters crypto map configuration mode and creates or modifies a crypto map entry. • Enter the crypto map name and sequence number.
Step 4	**exit** **Example:** `Router(config-crypto-map)# exit`	Returns to global configuration mode.
Step 5	**interface** *type number* [**name-tag**] **Example:** `Router(config)# interface serial4/0/0`	Configures an interface type and enters interface configuration mode. • Enter the interface type and number.
Step 6	**qos pre-classify** **Example:** `Router(config-if)# qos pre-classify`	Enables QoS preclassification.
Step 7	**end** **Example:** `Router(config-if)# end`	(Optional) Exits interface configuration mode and returns to privileged EXEC mode.

Configuration Examples for QoS for VPNs

Example Configuring QoS When Using IPsec VPNs

The following is an example of configuring QoS when using IPsec VPNs. In this example, the **crypto map** command specifies the IPsec crypto map (mymap 10) to which the **qos pre-classify** command will be applied.

```
Router> enable
Router# configure terminal
Router(config)# crypto map mymap 10

Router(config-crypto-map)# qos pre-classify
Router(config-crypto-map)# exit
```

Additional References for QoS for VPNs

Related Documents

Related Topic	Document Title
Cisco commands	Cisco IOS Master Command List, All Releases
QoS commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco IOS Quality of Service Solutions Command Reference*
Classifying network traffic	"Classifying Network Traffic" module
MQC	"Applying QoS Features Using the MQC" module
Marking network traffic	"Marking Network Traffic" module

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for QoS for VPNs

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

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

Table 27: Feature Information for QoS for VPNs

Feature Name	Releases	Feature Information
Quality of Service for Virtual Private Networks	12.2(2)T Cisco IOS XE Release 3.9S	The QoS for VPNs feature provides a solution for making Cisco IOS QoS services operate in conjunction with tunneling and encryption on an interface. Cisco IOS software can classify packets and apply the appropriate QoS service before the data is encrypted and tunneled. The QoS for VPN feature allows users to look inside the packet so that packet classification can be done based on original port numbers and based on source and destination IP addresses. This allows the service provider to treat mission critical or multi-service traffic with higher priority across their network.
QoS: Traffic Pre-classification	Cisco IOS XE Release 2.1	This feature was introduced on Cisco ASR 1000 Series Routers.