Modular QoS Configuration Guide for Cisco NCS 5000 Series Routers, IOS XR Release 6.6.x

First Published: 2019-04-01

Last Modified: 2019-12-01

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

Preface

This preface contains these sections:

Changes to this Document

Note *This software release has reached end-of-life status. For more information, see the* End-of-Life and End-of-Sale Notices.

This table lists the changes made to this document since it was first published.

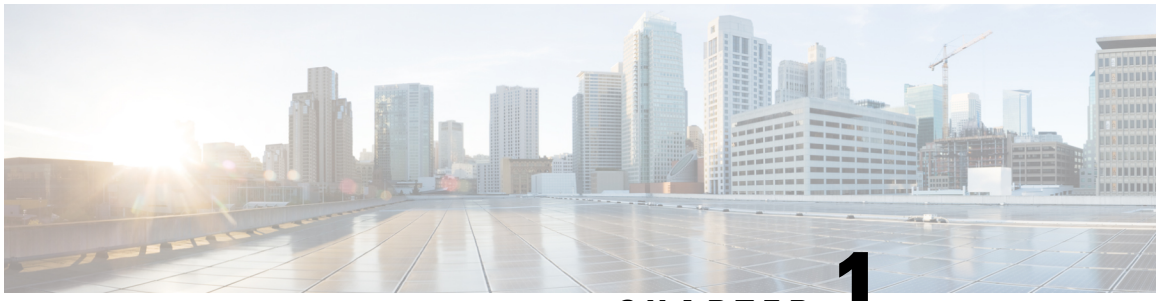
Date	Summary
April 2019	Initial release of this document.
December 2019	Republished for Release 6.6.3

Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco Marketplace.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.



CHAPTER 1

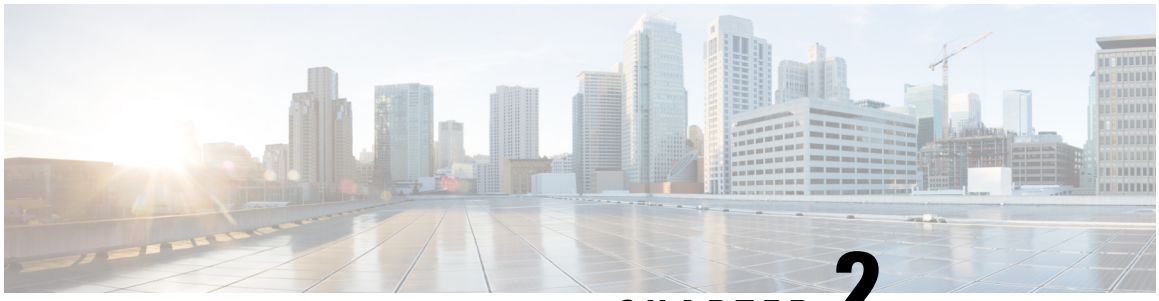
New and Changed QoS Features

New and Changed QoS Features

Table 1: QoS Features Added or Modified in IOS XR Release 6.6.x

Feature	Description	Changed in Release	Where Documented
QoS L2 Re-Marking of Ethernet Packets on L3 Flows in Egress Direction	This features enables you to perform Layer 2 (802.1p) marking on Layer 3 flows in the egress direction. This allows you to re-mark the priority of Ethernet packets on L3VPN traffic.	Release 6.6.3	QoS L2 Re-Marking of Ethernet Packets on L3 Flows in Egress Direction, on page 14

New and Changed QoS Features



CHAPTER 2

Configuring Modular QoS Service Packet Classification

This chapter covers these topics:

Packet Classification Overview

Packet classification involves categorizing a packet within a specific group (or class) and assigning it a traffic descriptor to make it accessible for QoS handling on the network. The traffic descriptor contains information about the forwarding treatment (quality of service) that the packet should receive. Using packet classification, you can partition network traffic into multiple priority levels or classes of service. The source agrees to adhere to the contracted terms and the network promises a quality of service. Traffic policers and traffic shapers use the traffic descriptor of a packet to ensure adherence to the contract.

Traffic policers and traffic shapers rely on packet classification features, such as IP precedence, to select packets (or traffic flows) traversing a router or interface for different types of QoS service. After you classify packets, you can use other QoS features to assign the appropriate traffic handling policies including congestion management, bandwidth allocation, and delay bounds for each traffic class.

The Modular Quality of Service (QoS) CLI (MQC) defines the traffic flows that must be classified, where each traffic flow is called a class of service, or class. Later, a traffic policy is created and applied to a class. All traffic not identified by defined classes fall into the category of a default class.

Traffic Class Elements

The purpose of a traffic class is to classify traffic on your router. Use the **class-map** command to define a traffic class.

A traffic class contains three major elements:

- A name
- A series of **match** commands - to specify various criteria for classifying packets.
- An instruction on how to evaluate these **match** commands (if more than one **match** command exists in the traffic class)

Packets are checked to determine whether they match the criteria that are specified in the **match** commands. If a packet matches the specified criteria, that packet is considered a member of the class and is forwarded according to the QoS specifications set in the traffic policy. Packets that fail to meet any of the matching criteria are classified as members of the default traffic class.

This table shows the details of match types that are supported on the router.

Match Type Supported	Min, Max	Max Entries	Support for Match NOT	Support for Ranges	Direction Supported on Interfaces
IPv4 DSCP IPv6 DSCP DSCP	(0,63)	64	Yes	Yes	Ingress
IPv4 Precedence IPv6 Precedence Precedence	(0,7)	8	Yes	Yes	Ingress
MPLS Experimental Topmost	(0,7)	8	Yes	Yes	Ingress
Access-group	Not applicable	Not applicable	No	Not applicable	Ingress
QoS-group	(1,7) (1,511) for peering profile	7	No	No	• Egress • Ingress for QoS Policy Propagation Using Border Gateway Protocol (QPPB) • Ingress for peering profile
Traffic-class	(1,7)	7	No	No	• Egress
Protocol	Not applicable	Not applicable	Yes	Not applicable	Ingress

Default Traffic Class

Unclassified traffic (traffic that does not meet the match criteria specified in the traffic classes) is treated as belonging to the default traffic class.

If the user does not configure a default class, packets are still treated as members of the default class. However, by default, the default class has no enabled features. Therefore, packets belonging to a default class with no configured features have no QoS functionality. These packets are then placed into a first in, first out (FIFO) queue and forwarded at a rate determined by the available underlying link bandwidth.

For egress classification, match on **traffic-class** (1-7) is supported. Match **traffic-class 0** cannot be configured. The class-default in the egress policy maps to **traffic-class 0**.

This example shows how to configure a traffic policy for the default class:

```
configure
 policy-map ingress_policy1
 class class-default
  police rate percent 30
 !
```

Create a Traffic Class

To create a traffic class containing match criteria, use the **class-map** command to specify the traffic class name, and then use the **match** commands in class-map configuration mode, as needed.

Guidelines

- Users can provide multiple values for a match type in a single line of configuration; that is, if the first value does not meet the match criteria, then the next value indicated in the match statement is considered for classification.
- Use the **not** keyword with the **match** command to perform a match based on the values of a field that are not specified.
- All **match** commands specified in this configuration task are considered optional, but you must configure at least one match criterion for a class.
- If you specify **match-any**, one of the match criteria must be met for traffic entering the traffic class to be classified as part of the traffic class. This is the default. If you specify **match-all**, the traffic must match all the match criteria.
- For the **match access-group** command, QoS classification based on the packet length or TTL (time to live) field in the IPv4 and IPv6 headers is not supported.
- For the **match access-group** command, when an ACL list is used within a class-map, the deny action of the ACL is ignored and the traffic is classified based on the specified ACL match parameters.

  An empty ACL (contains no rules, only remarks), when used within a class-map permits all traffic by default, and the implicit deny condition doesn't work with an empty ACL. The corresponding **class-map** matches all traffic not yet matched by the preceding traffic classes.
- The **traffic-class** and **discard-class** are supported only in egress direction, and these are the only match criteria supported in egress direction.
- The egress default class implicitly matches **qos-group** 0 for marking policy and **traffic-class** 0 for queuing policy.
- For egress classification, you must configure all 8 (qos-group) classes including class-default.

- If you set a traffic class at the ingress policy and do not have a matching class at egress for the corresponding traffic class value, then the traffic at ingress with this class will not be accounted for in the default class at the egress policy map.
- Only traffic class 0 falls in the default class. A non-zero traffic class assigned on ingress but with no assigned egress queue, falls neither in the default class nor any other class.

Configuration Example

You have to accomplish the following to complete the traffic class configuration:

1. Creating a class map
2. Specifying the match criteria for classifying the packet as a member of that particular class

   (For a list of supported match types, see Traffic Class Elements, on page 3.)

```
Router# configure
Router(config)# class-map match-any qos-1
Router(config-cmap)# match qos-group 1
Router(config-cmap)# end-class-map
Router(config-cmap)# commit
```

Also see, Running Configuration, on page 9.

Also see, Verification, on page 9.

Related Topics

- Traffic Class Elements, on page 3
- Traffic Policy Elements, on page 6

Associated Commands

- class-map
- match access-group
- match cos
- match dscp
- match mpls experimental topmost
- match precedence
- match protocol
- match qos-group

Traffic Policy Elements

A traffic policy contains three elements:

- Name
- Traffic class
- QoS policies

After choosing the traffic class that is used to classify traffic to the traffic policy, the user can enter the QoS features to be applied to the classified traffic.

The MQC does not necessarily require that the users associate only one traffic class to one traffic policy.

The order in which classes are configured in a policy map is important. The match rules of the classes are programmed into the TCAM in the order in which the classes are specified in a policy map. Therefore, if a packet can possibly match multiple classes, only the first matching class is returned and the corresponding policy is applied.

The router supports 32 classes per policy-map in the ingress direction and 8 classes per policy-map in the egress direction.

This table shows the supported class-actions on the router.

Supported Action Types	Direction supported on Interfaces
bandwidth-remaining	egress
mark	(See Packet Marking, on page 13)
police	ingress
priority	egress (level 1)
shape	egress

Create a Traffic Policy

The purpose of a traffic policy is to configure the QoS features that should be associated with the traffic that has been classified in a user-specified traffic class or classes.

To configure a traffic class, see Create a Traffic Class, on page 5.

After you define a traffic policy with the **policy-map** command, you can attach it to one or more interfaces to specify the traffic policy for those interfaces by using the **service-policy** command in interface configuration mode. With dual policy support, you can have two traffic policies, one marking and one queuing attached at the output. See, Attach a Traffic Policy to an Interface, on page 8.

Configuration Example

You have to accomplish the following to complete the traffic policy configuration:

1. Creating a policy map that can be attached to one or more interfaces to specify a service policy
2. Associating the traffic class with the traffic policy
3. Specifying the class-action(s) (see Traffic Policy Elements, on page 6)

```
Router# configure
```

```
Router(config)# policy-map egress_policy2
Router(config-pmap)# class qos-1

/* Configure class-action ('bandwidth remaining' in this example).
Repeat as required, to specify other class-actions */
Router(config-pmap-c)# bandwidth remaining ratio 50
Router(config-pmap-c)# exit

/* Repeat class configuration as required, to specify other classes */

Router(config-pmap)# end-policy-map
Router(config)# commit
```

See, Running Configuration, on page 9.

See, Verification, on page 9.

Related Topics

- Traffic Policy Elements, on page 6
- Traffic Class Elements, on page 3

Associated Commands

- bandwidth remaining
- class
- police
- policy-map
- priority
- set cos
- set dscp
- set qos-group
- shape

Attach a Traffic Policy to an Interface

After the traffic class and the traffic policy are created, you must attach the traffic policy to interface, and specify the direction in which the policy should be applied.

Configuration Example

You have to accomplish the following to attach a traffic policy to an interface:

1. Creating a traffic class and the associated rules that match packets to the class (see Create a Traffic Class, on page 5)
2. Creating a traffic policy that can be attached to one or more interfaces to specify a service policy (see Create a Traffic Policy, on page 7)

3. Associating the traffic class with the traffic policy
4. Attaching the traffic policy to an interface, in the ingress or egress direction

```
Router# configure
Router(config)# interface TenGig 0/0/0/0
Router(config-int)# service-policy output egress_policy2
Router(config-int)# commit
```

Running Configuration

```
/* Class-map configuration */

class-map match-any qos-1
 match qos-group 1
 end-class-map
!
- - -
- - -
!
class-map match-any qos-7
 match qos-group 7
 end-class-map

/* Policy-map configuration */
policy-map egress_policy2
 class qos-1
  bandwidth remaining ratio 50
 !
- - -
- - -
 class qos-7
  priority level 1
 !
 class class-default
  bandwidth remaining ratio 2
 !
 end-policy-map

/* Attaching traffic policy to an interface in egress direction */
interface TenGig 0/0/0/0
 service-policy output egress_policy2
 !
```

Verification

```
Router# show policy-map interface tenGigE 0/0/0/0

TenGigE0/0/0/0 direction input: Service Policy not installed

TenGigE0/0/0/0 output: egress_policy2

Class qos-1
  Classification statistics          (packets/bytes)     (rate - kbps)
    Matched             :             4867412/3598770345           0
    Transmitted         :             2901271/2142905575           0
    Total Dropped       :             1966141/1455864770           0
  Queueing statistics
    Queue ID                              : 9
```

```
    High watermark                          : N/A
    Inst-queue-len  (cells)                 : 10709
    Avg-queue-len                           : N/A
    Taildropped(packets/bytes)              : 1966141/1455864770
    Queue(conform)      :              2901271/2142905575           0
    RED random drops(packets/bytes)         : 0/0

- - -
- - -

Class class-default
  Classification statistics          (packets/bytes)     (rate - kbps)
    Matched             :              262482/237940215             0
    Transmitted         :               97172/88070686              0
    Total Dropped       :              165310/149869529             0
  Queueing statistics
    Queue ID                                : 8
    High watermark                          : N/A
    Inst-queue-len  (cells)                 : 10714
    Avg-queue-len                           : N/A
    Taildropped(packets/bytes)              : 165310/149869529
    Queue(conform)      :               97172/88070686              0
    RED random drops(packets/bytes)         : 0/0
```

Related Topics

- Traffic Policy Elements, on page 6
- Traffic Class Elements, on page 3

Associated Commands

- service-policy

Bundle Traffic Policies

A policy can be bound to bundles. When a policy is bound to a bundle, the same policy is programmed on every bundle member (port). For example, if there is a policer or shaper rate, the same rate is configured on every port. Traffic is scheduled to bundle members based on the load balancing algorithm.

Both ingress and egress traffic is supported. Percentage-based policies and absolute rate-based policies are supported. However, for ease of use, it is recommended to use percentage-based policies.

For details, see Configure QoS on Link Bundles, on page 37.

Class-based Unconditional Packet Marking

The class-based, unconditional packet marking feature provides users with a means to differentiate packets based on the designated markings. This feature allows you to partition your network into multiple priority levels or classes of service.

These tasks can be performed with the packet marking feature:

- Mark packets by setting the IP differentiated services code point (DSCP) in the IP ToS byte.

- Mark packets by setting the Layer 2 class-of-service (CoS) value.
- Mark packets by setting outer CoS tags for an IEEE 802.1Q tunneling (QinQ) configuration.
- Mark packets by setting the value of the *qos-group* argument.

Note *qos-group* is a variable internal to the router, and is not transmitted.

For more details, see Packet Marking, on page 13.

Configure Class-based Unconditional Packet Marking

The Cisco NCS 5000 Series Router support unconditional packet marking in ingress direction.

Guidelines

- Only ingress markings are supported.
- A maximum of only two **set** commands are allowed per class.
- You must configure all 8 classes (including default-class) for a policy-map.

Configuration Example

You have to accomplish the following to complete the unconditional packet marking configuration:

1. Creating or modifying a policy-map that can be attached to one or more interfaces
2. Specifying the traffic class whose policy has to be created or changed
3. Specifying the marking action for the packet
4. Attaching the policy-map to an input interface

```
Router# configure
Router(config)# policy-map policy1
Router(config-pmap)# class class1
/* Specify the marking action. Repeat the set command to specify another marking action */
Router(config-pmap-c)# set dscp 5
Router(config-pmap-c)# exit
/* Repeat the above steps to configure the remaining classes of the policy-map*/
Router(config-pmap)# exit

Router(config)# interface TenGigE 0/2/0/0
Router(config-if)# service-policy input policy1
Router(config-if)# no shutdown
Router(config-if)# commit
```

Running Configuration

```
policy-map policy1
 class class1
```

```
  set dscp 5
  set cos 7
  !
  - - -
  - - -
 !
 end-policy-map

interface TenGigE 0/2/0/0
 service-policy input policy1
!
```

Related Topics

- Class-based Unconditional Packet Marking, on page 10

Associated Commands

- set cos
- set dscp
- set qos-group

In-Place Policy Modification

The In-Place policy modification feature allows you to modify a QoS policy even when the QoS policy is attached to one or more interfaces. A modified policy is subjected to the same checks that a new policy is subject to when it is bound to an interface. If the policy-modification is successful, the modified policy takes effect on all the interfaces to which the policy is attached. However, if the policy modification fails on any one of the interfaces, an automatic rollback is initiated to ensure that the pre-modification policy is in effect on all the interfaces.

You can also modify any class map used in the policy map. The changes made to the class map take effect on all the interfaces to which the policy is attached.

Note

- The QoS statistics for the policy that is attached to an interface are lost (reset to 0) when the policy is modified.
- When a QoS policy attached to an interface is modified, there might not be any policy in effect on the interfaces in which the modified policy is used for a short period of time.
- The system does not support the show policy-map statistics for marking policies.
- An in-place modification of an ACL does not reset the policy-map statistics counter.

Note

- For QOS EXP-Egress marking applied on L3 interface, there is a limit of 3 unique policy-maps per NPU. When the maximum limit for policy-maps is reached and you try to modify a policy-map which is shared between different interfaces, you may get an error.
- For QOS egress marking (CoS, DEI) applied on L2 interface, there is a limit of 13 unique policy-maps per NPU. When the maximum limit for policy-maps is reached and you try to modify a policy-map which is shared between different interfaces, you may get an error

Verification

If unrecoverable errors occur during in-place policy modification, the policy is put into an inconsistent state on target interfaces. No new configuration is possible until the configuration session is unblocked. It is recommended to remove the policy from the interface, check the modified policy and then re-apply accordingly.

Use the **show qos inconsistency** command to view inconsistency in each location. The configuration session is blocked until the modified policy is effective on all interfaces that are using the policy. Output from the show policy-map targets command indicates that the TenGigabit Ethernet interface 0/1/0/0 has one policy map attached as a main policy (as opposed to being attached to a child policy in a hierarchical QoS configuration). Outgoing traffic on this interface is affected if the policy is modified

```
router# show policy-map targets

1) Policymap: policy1    Type: qos
     Targets (applied as main policy):
       TenGigabitEthernet0/1/0/0 output
     Total targets: 1

     Targets (applied as child policy):
     Total targets: 0
```

References for Modular QoS Service Packet Classification

Packet Marking

The packet marking feature provides users with a means to differentiate packets based on the designated markings. The router supports egress packet marking. match on **discard-class** on egress, if configured, can be used for a marking policy only.

The router also supports L2 ingress marking.

Supported Packet Marking Operations

This table shows the supported packet marking operations.

Supported Mark Types	Range	Support for Unconditional Marking	Support for Conditional Marking
set dscp	0-63	ingress	No
set QoS-group	0-7	ingress	No

Supported Mark Types	Range	Support for Unconditional Marking	Support for Conditional Marking
set traffic-class	0-7	ingress	No

Class-based Unconditional Packet Marking

The packet marking feature allows you to partition your network into multiple priority levels or classes of service, as follows:

- Use QoS unconditional packet marking to set the IP precedence or IP DSCP values for packets entering the network. Routers within your network can then use the newly marked IP precedence values to determine how the traffic should be treated.

  On ingress direction, after matching the traffic based on either the IP Precedence or DSCP value, you can set it to a particular discard-class. Weighted random early detection (WRED), a congestion avoidance technique, thereby uses discard-class values to determine the probability that a packet is dropped.

- Use QoS unconditional packet marking to assign MPLS packets to a QoS group. The router uses the QoS group to determine how to prioritize packets for transmission. To set the traffic class identifier on MPLS packets, use the **set traffic-class** command in policy map class configuration mode.

Note Setting the traffic class identifier does not automatically prioritize the packets for transmission. You must first configure an egress policy that uses the traffic class.

Note

- Unless otherwise indicated, the class-based unconditional packet marking for Layer 3 physical interfaces applies to bundle interfaces.

QoS L2 Re-Marking of Ethernet Packets on L3 Flows in Egress Direction

The router supports Layer 2 marking of Ethernet packets on Layer 3 flows in the egress direction. To enable this feature, you must:

- Configure the policy maps for marking at the egress interface.
- Ensure that the **set qos-group** command is configured in ingress policy and the corresponding **match qos-group** command is configured in the egress marking policy. If there is no corresponding QoS group, you will experience traffic failure.

Restrictions

The following restrictions apply while configuring the Layer 2 marking of Ethernet packets on Layer 3 flows in the egress direction.

- Egress marking statistics are not available.
- Layer 2 (802.1p) Egress marking is supported on Layer 3 flows only for MPLS-to-IP traffic.

Running Configuration

Ingress Policy:

You must first set up the qos-group at ingress.

```
class-map match-any Class0
 match mpls experimental topmost 0

 end-class-map
class-map match-any Class1
 match mpls experimental topmost 1

 end-class-map
class-map match-any Class2
 match mpls experimental topmost 2

 end-class-map
class-map match-any Class3
 match mpls experimental topmost 3

 end-class-map
class-map match-any Class4
 match mpls experimental topmost 4

 end-class-map
class-map match-any Class5
 match mpls experimental topmost 5

 end-class-map
class-map match-any Class6
 match mpls experimental topmost 6
end-class-map
class-map match-any Class7
 match mpls experimental topmost 7

 end-class-map
!

policy-map ncs_input
 class Class7
  set traffic-class 7
    set qos-group 7
!
 class Class6
  set traffic-class 6
  set qos-group 6
!
 class Class5
  set traffic-class 5
    set qos-group 5
!
 class Class4
  set traffic-class 4
  set qos-group 4
 !
 class Class3
  set traffic-class 4
    set qos-group 3
 !
 class Class2
  set traffic-class 2
    set qos-group 2
 !
 class Class1
```

```
  set traffic-class 2
    set qos-group 1
 !
 class Class0
  set traffic-class 0
    set qos-group 0
 !
 end-policy-map
!
```

Egress Policy:

At the egress, run these commands to mark the packets.

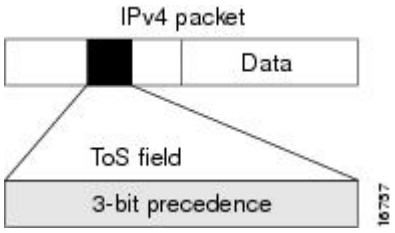
```
class-map match-any qos7
match gos-group 7
 end-class-map
!
class-map match-any qos6
match gos-group 6
 end-class-map
!
class-map match-any qos5
match qos-group 5
 end-class-map
!
class-map match-any qos4
match gos-group 4
 end-class-map
!
class-map match-any qos3
match gos-group 3
 end-class-map
!
class-map match-any qos2
 match gos-group 2
 end-class-map
!
class-map match-any qos1
match gos-group 1
 end-class-map
!

policy-map ncs_output
 class qos7
  set cos 7
  !
 class qos6
  set cos 6
!
 class qos5
  set cos 5
  !
 class qos4
  set cos 4
!
 class qos3
  set cos 3
!
 class qos2
  set cos 2

!
 class qos1
  set cos 1
```

```
  !
 end-policy-map
!
```

Specification of the CoS for a Packet with IP Precedence

Use of IP precedence allows you to specify the CoS for a packet. You can create differentiated service by setting precedence levels on incoming traffic and using them in combination with the QoS queuing features. 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 administrative domain as possible. This allows the rest of the core or backbone to implement QoS based on precedence.

Figure 1: IPv4 Packet Type of Service Field



You can use the three precedence bits in the type-of-service (ToS) field of the IPv4 header for this purpose. Using the ToS bits, you can define up to eight classes of service. Other features configured throughout the network can then use these bits to determine how to treat the packet in regard to the ToS to grant it. These other QoS features can assign appropriate traffic-handling policies, including congestion management strategy and bandwidth allocation. For example, queuing features such as LLQ can use the IP precedence setting of the packet to prioritize traffic.

IP Precedence Bits Used to Classify Packets

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

Each precedence corresponds to a name. IP precedence bit settings 6 and 7 are reserved for network control information, such as routing updates. These names are defined in RFC 791.

IP Precedence Value Settings

By default, the routers leave the IP precedence value untouched. This preserves the precedence value set in the header and allows all internal network devices to provide service based on the IP precedence setting. This policy follows the standard approach stipulating that network traffic should be sorted into various types of service at the edge 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 to determine the order of transmission, the likelihood of packet drop, and so on.

Because traffic coming into your network can have the precedence set by outside devices, we recommend that you reset the precedence for all traffic entering 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.

The class-based unconditional packet marking and LLQ features can use the IP precedence bits.

Dynamic Modification of Interface Bandwidth

Policy States

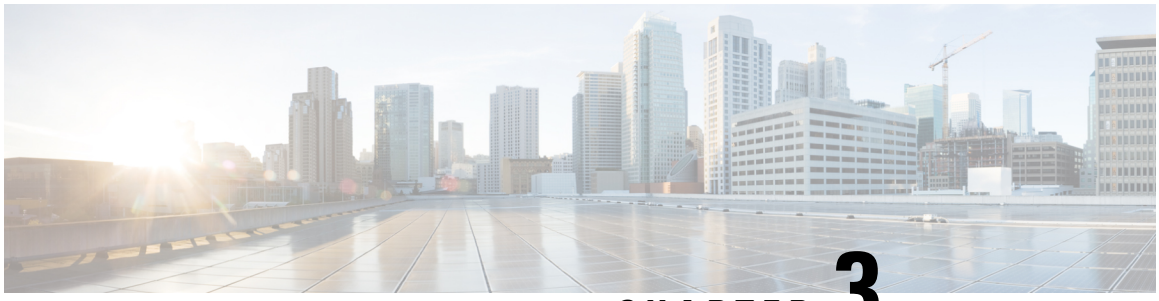
During the dynamic bandwidth modification process, if the modification is successful, the system does not display policy state information. However, if an error occurs, the system places the interface in one of these states and provides a policy-state error notification:

- Verification—This state indicates an incompatibility of the configured QoS policy with respect to the new interface bandwidth value. The system handles traffic on a best-efforts basis and some traffic drops can occur.
- Hardware programming—This state indicates a hardware programming failure caused by one of these conditions:
  - The modification of the interface default QoS resources encountered hardware update failures.
  - The modification of QoS resources (associated with the applied QoS policy) encountered hardware update failures.

    With either of these failures, hardware programming could be in an inconsistent state, which can impact features such as policing, queueing and marking. Therefore, the system disables QoS policy in hardware for these error conditions.
- Reset—In response to user reconfiguration of QoS policy, the system attempts to apply the new policy but fails; the fallback to the previous QoS policy also fails.

If you receive notification of any of these policy states, you need to reconfigure the QoS policy to clear this condition.

Use the **show qos interface** and **show policy-map interface** commands to query the QoS state of the interface. The system displays the QoS policy status if the interface is in one of the error states (verification, hardware programming, or reset), but does not display it if the state is active.



CHAPTER 3

Configuring Modular QoS Congestion Management

This chapter covers the following topics:

Congestion Management Overview

Congestion management features allow you to control congestion by determining the order in which a traffic flow (or packets) is sent out an interface based on priorities assigned to packets. Congestion management entails the creation of queues, assignment of packets to those queues based on the classification of the packet, and scheduling of the packets in a queue for transmission.

The types of traffic regulation mechanisms supported are:

- Modified Deficit Round Robin Queueing, on page 19
- Low-Latency Queuing with Strict Priority Queuing, on page 24
- Traffic Shaping, on page 27
- Traffic Policing, on page 30

Among the ones listed above, traffic policing is the one used for congestion management on the ingress side. Others are used for congestion management on the egress side.

Modified Deficit Round Robin Queueing

Modified Deficit Round Robin (MDRR) is a class-based composite scheduling mechanism that allows for queueing of up to eight traffic classes. It operates in the same manner as class-based weighted fair queueing (CBWFQ) and allows definition of traffic classes based on customer match criteria. When MDRR is configured in the queuing strategy, non-empty queues are served one after the other. Each time a queue is served, a fixed amount of data is dequeued. The algorithm then services the next queue.

Bandwidth Remaining

The MDRR algorithm derives the weight for each class from the bandwidth remaining value allocated to the class. The **bandwidth remaining** option specifies a weight for the class to the MDRR . After the priority-queue is serviced, the leftover bandwidth is distributed as per bandwidth remaining ratio (BWRR) or percentage. If you do not configure this command for any class, the default value of the BWRR is considered as 1 (one). In the case of **bandwidth remaining percent**, the remaining bandwidth is equally distributed among other classes, to make it 100 percentage (100%).

Restrictions

- The **bandwidth remaining** command is supported only for egress policies.

Configure Bandwidth Remaining

Guidelines

- It is mandatory to configure all the eight qos-group classes (including class-default) for the egress policies in Cisco NCS 5000 Series Routers.

Configuration Example

You have to accomplish the following to complete the bandwidth remaining configuration:

1. Creating or modifying a policy-map that can be attached to one or more interfaces
2. Specifying the traffic class whose policy has to be created or changed
3. Allocating the leftover bandwidth for the class
4. Attaching the policy-map to an output interface

```
Router# configure
Router(config)# policy-map egress_policy1
Router(config-pmap)# class class1
Router(config-pmap-c)# bandwidth remaining percent 2
Router(config-pmap-c# exit
Router(config-pmap)# exit
Router(config)# interface TenGigE 0/0/0/0
Router(config-if)# service-policy output egress_policy1
Router(config-if)# commit

/* Bandwidth remaining can also be configured as ratio, instead of percentage */
Router(config-pmap-c)# bandwidth remaining ratio 50
```

Running Configuration

Here bandwidth remaining is configured as percentage:

```
/* Class-map configuration */

class-map match-any class1
 match qos-group 1
 end-class-map
!
```

```
class-map match-any class2
 match qos-group 2
 end-class-map
!
class-map match-any class3
 match qos-group 3
 end-class-map
!
class-map match-any class4
 match qos-group 4
 end-class-map
!
class-map match-any class5
 match qos-group 5
 end-class-map
!
class-map match-any class6
 match qos-group 6
 end-class-map
!
class-map match-any class7
 match qos-group 7
 end-class-map

/* Policy-map configuration */

policy-map egress_policy1
 class class1
  bandwidth remaining percent 2
 !
 class class2
  bandwidth remaining percent 3
 !
 class class3
  priority level 1
 !
 class class4
  bandwidth remaining percent 50
  shape average 100 mbps
 !
 class class5
  bandwidth remaining percent 2
 !
 class class6
  bandwidth remaining percent 25
 !
 class class7
  bandwidth remaining percent 6
 !
 class class-default
  bandwidth remaining percent 12
 !
 end-policy-map

interface TenGigE 0/0/0/0
 service-policy output egress_policy1
!
```

Here bandwidth remaining is configured as ratio:

```
/* Class-map configuration */

class-map match-any class1
```

```
 match qos-group 1
 end-class-map
!
class-map match-any class2
 match qos-group 2
 end-class-map
!
class-map match-any class3
 match qos-group 3
 end-class-map
!
class-map match-any class4
 match qos-group 4
 end-class-map
!
class-map match-any class5
 match qos-group 5
 end-class-map
!
class-map match-any class6
 match qos-group 6
 end-class-map
!
class-map match-any class7
 match qos-group 7
 end-class-map

/* Policy-map configuration */

policy-map egress_policy2
 class class1
  bandwidth remaining ratio 50
 !
 class class2
  bandwidth remaining ratio 25
 !
 class class3
  bandwidth remaining ratio 12
 !
 class class4
  bandwidth remaining ratio 6
 !
 class class5
  bandwidth remaining ratio 3
 !
 class class6
  bandwidth remaining ratio 2
 !
 class class7
  priority level 1
 !
 class class-default
  bandwidth remaining ratio 2
 !
 end-policy-map
```

Verification

Each class has a classification statistics (that include total transmitted and dropped counts of that class) and a queueing statistics (that include the tail drop counts, the conformed queue statistics and the instantaneous queue length).

Verify if the queue is growing, by checking the values of *Inst-queue-len (cells)*. Also, check the values of *Total Dropped* field to see if there are any queue drops.

```
Router# show policy-map interface tenGigE 0/0/0/0
TenGigE0/0/0/0 direction input: Service Policy not installed

TenGigE0/0/0/0 output: egress_policy1

Class class1
  Classification statistics          (packets/bytes)     (rate - kbps)
    Matched             :            17312125/12799683759          0
    Transmitted         :            10774292/7953712189           0
    Total Dropped       :             6537833/4845971570           0
  Queueing statistics
    Queue ID                            : 9
    High watermark                      : N/A
    Inst-queue-len  (cells)             : 22981
    Avg-queue-len                       : N/A
    Taildropped(packets/bytes)          : 6537833/4845971570
    Queue(conform)      :            10774292/7953712189           0
    RED random drops(packets/bytes)     : 0/0

- - -
- - -

Class class-default
  Classification statistics          (packets/bytes)     (rate - kbps)
    Matched             :              941172/853166230            0
    Transmitted         :              926830/840112420            0
    Total Dropped       :               14342/13053810             0
  Queueing statistics
    Queue ID                            : 8
    High watermark                      : N/A
    Inst-queue-len  (cells)             : 0
    Avg-queue-len                       : N/A
    Taildropped(packets/bytes)          : 14342/13053810
    Queue(conform)      :              926830/840112420            0
    RED random drops(packets/bytes)     : 0/0
```

This is the show command output, if bandwidth remaining is configured as ratio:

```
Router# show policy-map interface tenGigE 0/0/0/0
TenGigE0/0/0/0 direction input: Service Policy not installed

TenGigE0/0/0/0 output: egress_policy2

Class class1
  Classification statistics          (packets/bytes)     (rate - kbps)
    Matched             :             4867412/3598770345           0
    Transmitted         :             2901271/2142905575           0
    Total Dropped       :             1966141/1455864770           0
  Queueing statistics
    Queue ID                            : 9
    High watermark                      : N/A
    Inst-queue-len  (cells)             : 10709
    Avg-queue-len                       : N/A
    Taildropped(packets/bytes)          : 1966141/1455864770
    Queue(conform)      :             2901271/2142905575           0
    RED random drops(packets/bytes)     : 0/0

- - -
- - -
```

```
Class class-default
  Classification statistics          (packets/bytes)     (rate - kbps)
    Matched             :              262482/237940215            0
    Transmitted         :               97172/88070686             0
    Total Dropped       :              165310/149869529            0
  Queueing statistics
    Queue ID                             : 8
    High watermark                       : N/A
    Inst-queue-len  (cells)              : 10714
    Avg-queue-len                        : N/A
    Taildropped(packets/bytes)           : 165310/149869529
    Queue(conform)      :               97172/88070686             0
    RED random drops(packets/bytes)      : 0/0
```

Related Topics

- Bandwidth Remaining, on page 20

Associated Commands

- bandwidth remaining

Low-Latency Queuing with Strict Priority Queuing

Priority queuing (PQ) in strict priority mode ensures that one type of traffic is sent, possibly at the expense of all others. For PQ, a low-priority queue can be detrimentally affected, and, in the worst case, never allowed to send its packets if a limited amount of bandwidth is available or the transmission rate of critical traffic is high.

Configuring Low Latency Queuing with Strict Priority queuing

Configuring low latency queuing (LLQ) with strict priority queuing (PQ) allows delay-sensitive data such as voice to be de-queued and sent before the packets in other queues are de-queued.

Guidelines

- Only priority level 1 is supported.
- Egress policing is not supported. Hence, in the case of strict priority queuing, there are chances that the other queues do not get serviced. Therefore, in order to minimize this, the user can police the traffic at the ingress side itself or design the network in a such a way that the priority traffic doesn't impact the other traffic on the egress port.
- You can configure **shape average** and **queue-limit** commands along with **priority**.
- You can configure **shape average**, **random-detect**, and **queue-limit** commands along with **priority**.
- There can be a minimal traffic disruption when priority level 1 configuration is applied on any of the 8 queues.
- Any one of the eight egress class-maps (queues) can have priority level 1 configuration.

Configuration Example

You have to accomplish the following to complete the LLQ with strict priority queuing:

1. Creating or modifying a policy-map that can be attached to one or more interfaces
2. Specifying the traffic class whose policy has to be created or changed.
3. Specifying priority to the traffic class
4. Attaching the policy-map to an output interface

```
Router# configure
Router(config)#class-map qos-1
Router(config-cmap)#match traffic-class 1
Router(config-cmap)#commit

Router(config)#class-map qos-2
Router(config-cmap)#match traffic-class 2
Router(config-cmap)#commit

Router(config)# policy-map egress_policy1
Router(config-pmap)# class qos1
Router(config-pmap-c)# priority level 1

Router(config-pmap-c# exit
Router(config-pmap)# exit

Router(config)# interface TenGigE 0/0/0/0
Router(config-if)# service-policy output egress_policy1
Router(config-if)# commit
```

Running Configuration

```
/* Class-map configuration */

class-map match-any qos-1
 match qos-group 1
 end-class-map
!
- - -
- - -
- - -
!
class-map match-any qos-7
 match qos-group 7
 end-class-map

/* Policy-map configuration */

policy-map egress_policy2
 class qos-1
  priority level 1
 !
 class qos-2
  bandwidth remaining ratio 1
 !
 class qos-3
  bandwidth remaining ratio 1
 !
```

```
 class qos-4
  bandwidth remaining ratio 1
 !
 class qos-5
  bandwidth remaining ratio 1
 !
 class qos-6
  bandwidth remaining ratio 1
 !
 class qos-7
  bandwidth remaining ratio 1
 !
 class class-default
  bandwidth remaining ratio 2
 !
 end-policy-map
```

Verification

Verify if the queue is growing, by checking the values of *Inst-queue-len (cells)*. This should ideally be 0 (zero). Also, check the values of *Total Dropped* field and ensure that there are no queue drops.

```
Router# show policy-map interface tenGigE 0/0/0/0
TenGigE0/0/0/0 direction input: Service Policy not installed

TenGigE0/0/0/0 output: egress_policy2

Class qos-1
  Classification statistics          (packets/bytes)     (rate - kbps)
    Matched             :            17312125/12799683759           0
    Transmitted         :            17312125/12799683759            0
    Total Dropped       :             0/0           0
  Queueing statistics
    Queue ID                            : 9
    High watermark                      : N/A
    Inst-queue-len  (cells)             : 0
    Avg-queue-len                       : N/A
    Taildropped(packets/bytes)          : 6537833/4845971570
    Queue(conform)      :            10774292/7953712189           0
    RED random drops(packets/bytes)     : 0/0

- - -
- - -

Class class-default
  Classification statistics          (packets/bytes)     (rate - kbps)
    Matched             :              941172/853166230            0
    Transmitted         :              926830/840112420            0
    Total Dropped       :               14342/13053810             0
  Queueing statistics
    Queue ID                            : 8
    High watermark                      : N/A
    Inst-queue-len  (cells)             : 0
    Avg-queue-len                       : N/A
    Taildropped(packets/bytes)          : 14342/13053810
    Queue(conform)      :              926830/840112420            0
    RED random drops(packets/bytes)     : 0/0
```

Related Topics

- Congestion Management Overview, on page 19
- Configure Traffic Shaping, on page 27
- Configure Bandwidth Remaining, on page 20

Associated Commands

- priority

Traffic Shaping

Traffic shaping allows you to control the traffic flow exiting an interface to match its transmission to the speed of the remote target interface and ensure that the traffic conforms to policies contracted for it. Traffic adhering to a particular profile can be shaped to meet downstream requirements, hence eliminating bottlenecks in topologies with data-rate mismatches.

Note Traffic shaping is supported only in egress direction.

Configure Traffic Shaping

The traffic shaping performed on outgoing interfaces is done at the Layer 1 level and includes the Layer 1 header in the rate calculation.

Guidelines

- Only egress traffic shaping is supported.
- It is mandatory to configure all the eight traffic-class classes (including class-default) for the egress policies.
- The **priority** and **shape average** commands must not be configured together in the same class.

Configuration Example

You have to accomplish the following to complete the traffic shaping configuration:

1. Creating or modifying a policy-map that can be attached to one or more interfaces
2. Specifying the traffic class whose policy has to be created or changed
3. Shaping the traffic to a specific bit rate
4. Attaching the policy-map to an output interface

```
Router(config)# policy-map egress_policy1
Router(config-pmap)# class c5
```

```
Router(config-pmap-c)# shape average 40 percent100 mbps
Router(config-pmap-c# exit
Router(config-pmap)# exit
Router(config)# interface TenGigE 0/0/0/0
Router(config-if)# service-policy output egress_policy1
Router(config-if)# commit
```

Running Configuration

```
policy-map egress_policy1
 class c1
  bandwidth remaining percent 10
 !
 class c3
  bandwidth remaining percent 10
 !
 class c5
  bandwidth remaining percent 20
  shape average 2 gbps  ===>
 !
 class c4
  bandwidth remaining percent 5
 !
 class c2
  priority level 1
 !
 class c7
  bandwidth remaining percent 20
 !
 class c6
  bandwidth remaining percent 15
 !
 class class-default
 !
 end-policy-map


class-map c5
 match traffic-class 5
commit

policy-map egress_policy1
 class c5
  shape average percent 40
 !
 class class-default
 !
 end-policy-map
!

interface HundredGigE0/6/0/18
 service-policy output egress_policy1
!
```

Verification

```
Router#  show qos interface  TenGigE 0/0/0/9 output

Mon Nov 16 15:41:15.738 UTC
```

```
Interface: TenGigE0_0_0_9 output
Bandwidth configured: 10000000 kbps Bandwidth programed: 10000000 kbps
ANCP user configured: 0 kbps ANCP programed in HW: 0 kbps
Port Shaper programed in HW: 0 kbps
Policy: egress Total number of classes: 8
------------------------------------------------------------------------
Level: 0 Policy: egress_policy1 Class: c1
QueueID: 81 (Priority Normal)
Committed Weight: 0 Excess Weight: 10
Bandwidth: 0 kbps, BW sum for Level 0: 0 kbps, Excess Ratio: 10
------------------------------------------------------------------------
Level: 0 Policy: egress_policy1 Class: c3
QueueID: 83 (Priority Normal)
Committed Weight: 0 Excess Weight: 10
Bandwidth: 0 kbps, BW sum for Level 0: 0 kbps, Excess Ratio: 10
------------------------------------------------------------------------
Level: 0 Policy: egress_policy1 Class: c5<<<<<<<<<<<
QueueID: 85 (Priority Normal)
PIR: 2000128 kbps  PBS: 25001875 bytes
Committed Weight: 0 Excess Weight: 20
Bandwidth: 0 kbps, BW sum for Level 0: 0 kbps, Excess Ratio: 20
------------------------------------------------------------------------
Level: 0 Policy: egress_policy1 Class: c4
QueueID: 84 (Priority Normal)
Committed Weight: 0 Excess Weight: 5
Bandwidth: 0 kbps, BW sum for Level 0: 0 kbps, Excess Ratio: 5
------------------------------------------------------------------------
Level: 0 Policy: egress_policy1 Class: c2
QueueID: 82 (Priority 1)
------------------------------------------------------------------------
Level: 0 Policy: egress_policy1 Class: c7
QueueID: 87 (Priority Normal)
Committed Weight: 0 Excess Weight: 20
Bandwidth: 0 kbps, BW sum for Level 0: 0 kbps, Excess Ratio: 20
------------------------------------------------------------------------
Level: 0 Policy: egress_policy1 Class: c6
QueueID: 86 (Priority Normal)
Committed Weight: 0 Excess Weight: 15
Bandwidth: 0 kbps, BW sum for Level 0: 0 kbps, Excess Ratio: 15
------------------------------------------------------------------------
Level: 0 Policy: egress_policy1 Class: class-default
QueueID: 80 (Priority Normal)
Committed Weight: 0 Excess Weight: 20
Bandwidth: 0 kbps, BW sum for Level 0: 0 kbps, Excess Ratio: 20
------------------------------------------------------------------------
```

```
Router#  show qos interface hundredGigE 0/6/0/18 output

NOTE:- Configured values are displayed within parentheses
Interface HundredGigE0/6/0/18 ifh 0x3000220  -- output policy
NPU Id:                        3
Total number of classes:       2
Interface Bandwidth:           100000000 kbps
VOQ Base:                      11176
VOQ Stats Handle:              0x88550ea0
Accounting Type:               Layer1 (Include Layer 1 encapsulation and above)
------------------------------------------------------------------------------
Level1 Class                             =   c5
Egressq Queue ID                         =   11177 (LP queue)
Queue Max. BW.                           =   40329846 kbps (40 %)
Queue Min. BW.                           =   0 kbps (default)
Inverse Weight / Weight                  =   1 (BWR not configured)
Guaranteed service rate                  =   40000000 kbps
```

```
TailDrop Threshold                       =   50069504 bytes / 10 ms (default)
WRED not configured for this class

Level1 Class                             =   class-default
Egressq Queue ID                         =   11176 (Default LP queue)
Queue Max. BW.                           =   101803495 kbps (default)
Queue Min. BW.                           =   0 kbps (default)
Inverse Weight / Weight                  =   1 (BWR not configured)
Guaranteed service rate                  =   50000000 kbps
TailDrop Threshold                       =   62652416 bytes / 10 ms (default)
WRED not configured for this class
```

Important Notes

Related Topics

- Congestion Management Overview, on page 19

Associated Commands

- shape average

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).Traffic policing manages the maximum rate of traffic through a token bucket algorithm. The token bucket algorithm uses user-configured values to determine the maximum rate of traffic allowed on an interface at a given moment in time. The token bucket algorithm is affected by all traffic entering or leaving the interface (depending on where the traffic policy with traffic policing is configured) and is useful in managing network bandwidth in cases where several large packets are sent in the same traffic stream. By default, the configured bandwidth value takes into account the Layer 2 encapsulation that is applied to traffic leaving the interface.

Traffic policing also provides a certain amount of bandwidth management by allowing you to set the burst size (Bc) for the committed information rate (CIR). See, Committed Bursts , on page 31.

In addition to rate-limiting, traffic policing also allows you to independently mark (or classify) the packets. See Policer Marking, on page 31.

The router supports the following traffic policing mode(s):

- Single-Rate Two-Color (SR2C) in color-blind mode. See Single-Rate Policer, on page 34.

Restrictions

- Traffic policing is supported only in ingress direction, and only color-blind mode is supported.
- The policing rate accuracy may vary up to +/-2% from the configured policer value.
- Policing credits are lost for traffic dropped by other features.

Committed Bursts

Unlike a traffic shaper, a traffic policer does not buffer excess packets and transmit them later. Instead, the policer executes a “send or do not send” policy without buffering. Policing uses normal or committed burst (bc) values to ensure that the router reaches the configured committed information rate (CIR). Policing decides if a packet conforms or exceeds the CIR based on the burst values you configure. Burst parameters are based on a generic buffering rule for routers, which recommends that you configure buffering to be equal to the round-trip time bit-rate to accommodate the outstanding TCP windows of all connections in times of congestion. During periods of congestion, proper configuration of the burst parameter enables the policer to drop packets less aggressively.

Committed Burst Calculation

To calculate committed burst, use the following formula:

bc (in bytes) = CIR bps * (1 byte / 8 bits) * 0.1 seconds

The standard time to be used in this calculation is 100 milliseconds (0.1 seconds).

For example, if the committed information rate is 20,00,000 bps, then using the committed burst formula, the committed burst is 25,000 bytes.

bc = 2000000 * (1/8)* (100/1000)

bc = 25,000 bytes

It is important that you set the burst values high enough to ensure good throughput. If your router drops packets and reports an exceeded rate even though the conformed rate is less than the configured CIR, use the **show interface** command to monitor the current burst, determine whether the displayed value is consistently close to the committed burst (bc) value, and if the actual rates (the committed rate) are close to the configured committed rate. If not, the burst values might be too low. Try reconfiguring the burst rates using the suggested calculations.

Policer Marking

In addition to rate-limiting, traffic policing allows you to independently mark (or classify) the packet according to whether the packet conforms or violates a specified rate. Packet marking also allows you to partition your network into multiple priority levels or CoS.

Policer marking is also referred as conditional marking, as the marking is done based on the policer state (conform or exceed). Policer marking is done by setting the IP precedence value or IP DSCP value. Use the traffic policer to set this value for the packets that enter the network. The networking devices within your network can then use this setting to determine how the traffic should be treated.

If you want to mark traffic but do not want to use traffic policing, you can use class-based, unconditional packet marking. See, Class-based Unconditional Packet Marking, on page 10.

Note Egress packet marking is not supported.

Multiple Action Set

The Multiple Action Set feature allows you to mark packets with multiple action sets (conditional and unconditional) through a class map.

These are the supported action sets:

- set-qos-group
- set-dscp
- set-cos

At least two set of actions for each policer action can be configured by using the **conform-action** command and the **exceed-action** command, within a class map for IP, MPLS, or Layer 2 data paths.

Note The ingress policy action **set qos-group** affects the marking of the packet.

Configure Conditional Policer Marking

The Cisco NCS 5000 Series Router support conditional policer marking in ingress direction.

Configuration Example

You have to accomplish the following to complete the conditional policer marking configuration:

1. Creating or modifying a policy-map that can be attached to one or more interfaces
2. Specifying the traffic class whose policy has to be created or changed
3. Specifying the policy rate for the traffic
4. Specifying the action(s) to be take on the packets that conform the rate limit
5. Specifying the action(s) to be take on the packets that exceed the rate limit
6. Attaching the policy-map to an input interface

```
Router# configure
Router(config)# policy-map ingress_policy1
Router(config-pmap)# class ic1
Router(config-pmap-c)# set qos-group 5
Router(config-pmap-c)# police rate 5 gbps
Router(config-pmap-c-police)# conform-action set qos-group 4
Router(config-pmap-c-police)# conform-action set dscp 3
Router(config-pmap-c-police)# exceed-action set qos-group 6
Router(config-pmap-c-police)# exit
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# interface TenGigE 0/0/0/5
Router(config-if)# service-policy input ingress_policy1
Router(config-if)# commit
```

Running Configuration

```
policy-map ingress
 class ic1
  set qos-group 5
  police rate 5 gbps
   conform-action set qos-group 4
   conform-action set dscp 3
   exceed-action set qos-group 6
  !
 !
 class class-default
 !
 end-policy-map

interface TenGigE 0/0/0/5
 service-policy input ingress_policy1
!
```

Verification

```
Router# show qos interface TenGigE 0/0/0/5 input

Interface: TenGigE0_0_0_5 input
Bandwidth configured: 10000000 kbps Bandwidth programed: 10000000 kbps
ANCP user configured: 0 kbps ANCP programed in HW: 0 kbps
Port Shaper programed in HW: 0 kbps
Policy: ingress Total number of classes: 2
----------------------------------------------------------------------
Level: 0 Policy: ingress_policy1 Class: ic1
QueueID: 0 (Port Default)
Policer Profile: 112 (Single)
Conform: 5000000 kbps (5 gbps) Burst: 62500000 bytes (0 Default)
Child Policer Conform: set qos-grp 4   set dscp 3
Child Policer Exceed: set qos-grp 6
----------------------------------------------------------------------
Level: 0 Policy: ingress_policy1 Class: class-default
QueueID: 0 (Port Default)
```

Related Topics

- Policer Marking, on page 31

Associated Commands

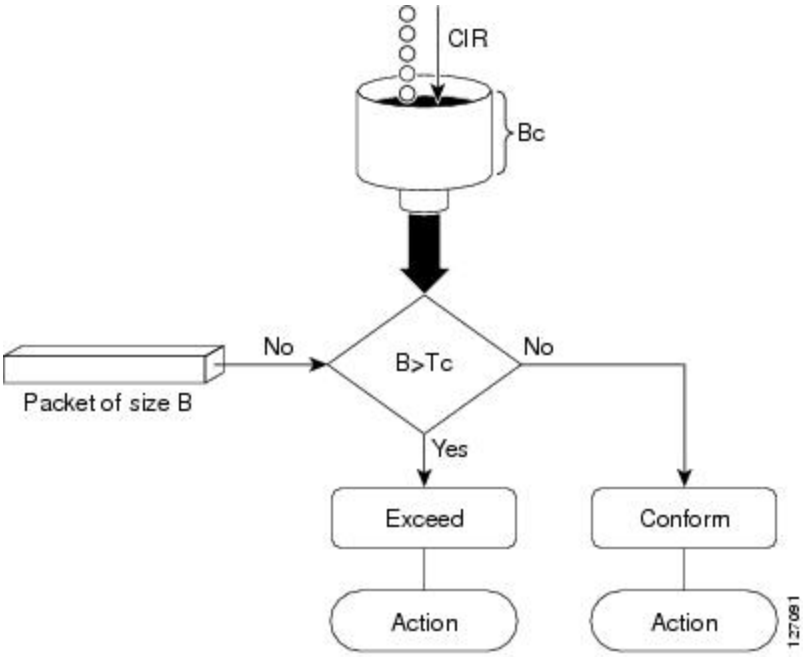
- conform-action
- exceed-action
- police rate

Single-Rate Policer

Single-Rate Two-Color Policer

A single-rate two-color (SR2C) policer provides one token bucket with two actions for each packet: a conform action and an exceed action.

Figure 2: Workflow of Single-Rate Two-Color Policer



Based on the committed information rate (CIR) value, the token bucket is updated at every refresh time interval. The Tc token bucket can contain up to the Bc value, which can be a certain number of bytes or a period of time. If a packet of size B is greater than the Tc token bucket, then the packet exceeds the CIR value and a configured action is performed. If a packet of size B is less than the Tc token bucket, then the packet conforms and a different configured action is performed.

Configure Traffic Policing (Single-Rate Two-Color)

Traffic policing is often configured on interfaces at the edge of a network to limit the rate of traffic entering or leaving the network. In the most common traffic policing configurations, traffic that conforms to the CIR is sent and traffic that exceeds is sent with a decreased priority or is dropped. Users can change these configuration options to suit their network needs.

Configuration Example

You have to accomplish the following to complete the Single-Rate Two-Color (SR2C) traffic policing configuration:

1. Creating or modifying a policy-map that can be attached to one or more interfaces
2. Specifying the traffic class whose policy has to be created or changed

3. Specifying the policy rate for the traffic
4. Specifying the action to be take on the packets that conform the rate limit
5. Specifying the action to be take on the packets that exceed the rate limit
6. Attaching the policy-map to an input interface

```
Router# configure
Router(config)# policy-map ingress_policy1
Router(config-pmap)# class class1
Router(config-pmap-c)# police rate 100 mbps burst 6000 bytes
Router(config-pmap-c-police)# conform-action set qos-group 2
Router(config-pmap-c-police)# conform-action set cos 2
Router(config-pmap-c-police)# exceed-action set qos-group 3
Router(config-pmap-c-police)# exceed-action set cos 3
Router(config-pmap-c-police)# exit
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# interface0/0/0/14
Router(config-if)# service-policy input policy1
Router(config-if)# no shutdown
Router(config-if)# commit
```

Running Configuration

```
/* class-map configuration */
class-map match-any class1
 match dscp 10
 end-class-map

/* Traffic policing configuration */
policy-map ingress_policy1
 class class1
  police rate 100 mbps burst 6000 bytes
   conform-action set qos-group 2
   conform-action set cos 2
   exceed-action set qos-group 3
   exceed-action set cos 3
  !
 !
 class class-default
 !
 end-policy-map

interface TenGigE 0/0/0/14
 service-policy input ingress_policy1
!
```

Verification

```
Router# show policy-map interface TenGigE 0/0/0/14 input
TenGigE0/0/0/14 input: ingress_policy1

Class class1
  Classification statistics          (packets/bytes)     (rate - kbps)
    Matched             :              410861/308109455            0
```

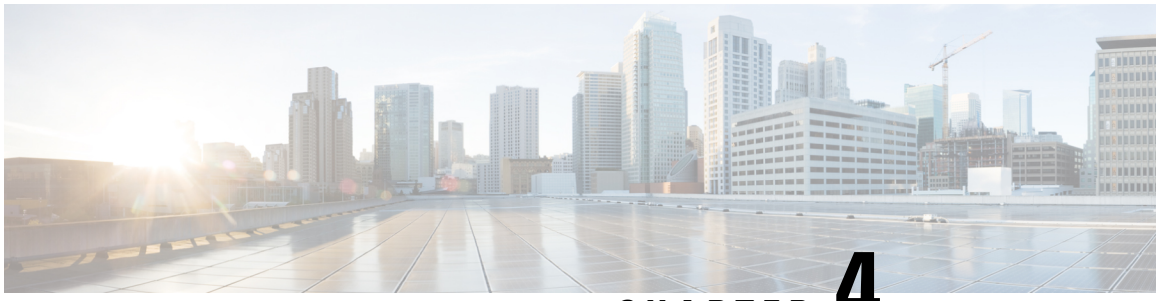
```
    Transmitted         : N/A
    Total Dropped       :                 0/0                   0
  Policing statistics                (packets/bytes)     (rate - kbps)
    Policed(conform)    :              27262/20543870            0
    Policed(exceed)     :             383599/287565585           0
    Policed(violate)    :                 0/0                   0
    Policed and dropped :                 0/0
Class class-default
  Classification statistics          (packets/bytes)     (rate - kbps)
    Matched             :                 0/0                   0
    Transmitted         : N/A
    Total Dropped       : N/A
```

Related Topics

- Single-Rate Policer, on page 34

Associated Commands

- conform-action
- exceed-action
- police rate
- policy-map
- service-policy



CHAPTER 4

Configuring Modular QoS on Link Bundles

This chapter covers the following topics:

QoS on Link Bundles

A bundle is a group of one or more ports that are aggregated together and treated as a single link. The router supports Ethernet interfaces and VLAN interfaces (bundle sub-interfaces) bundles. All QoS features currently supported on physical interfaces, are also supported on all link bundle interfaces. Applying QoS on bundle members is not supported.

For more info on link bundles, see *Configuring Link Bundles* chapter in *L2VPN and Ethernet Services Configuration Guide for the Cisco NCS 5000 Series Routers, IOS XR Release 6.0.x*

Load Balancing

Load balancing function is a forwarding mechanism to distribute traffic over multiple links based on Layer 3 routing information in the router. Per-destination load balancing isonly supported on the router, where the router is allowed to distribute packets over one of the links in the bundle. When the per-destination load balancing is enabled, all packets for a certain source-destination pair goes through the same link, though there are multiple links available. In other words, per-destination load balancing can ensure that packets for a certain source-destination pair could arrive in order.

Layer 3 Load Balancing on Link Bundles

Layer 3 load balancing for link bundles is done on Ethernet Flow Points (EFPs) and is based on the IPv4 source and destination addresses in the packet. When Layer 3 service-specific load balancing is configured, all egress bundles are load balanced based on the IPv4 source and destination addresses. When packets do not have IPv4 addresses, default load-balancing (based on the MAC SA/DA fields in the packet header) is used.

Configure QoS on Link Bundles

QoS is configured on link bundles in the same way that it is configured on individual interfaces.

Guidelines

- When a QoS policy is applied on a bundle in the egress direction, it's also applied at each member interface.
- When a QoS policy is applied on a bundle (ingress direction), it's replicated at each NPU core.
- If a QoS policy is not applied to a bundle interface, both the ingress and egress traffic use the default queue of the per link member port.
- The shape rate that is specified in the bundle policy-map is not an aggregate for all bundle members. The shape rate applied to the bundle depends on the load balancing of the links. For example, if a policy map with a shape rate of 10 Mbps is applied to a bundle with two member links, and if the traffic is always load-balanced to the same member link, then an overall rate of 10 Mbps applies to the bundle. However, if the traffic is load-balanced evenly between the two links, the overall shape rate for the bundle becomes 20 Mbps.
- If a member is deleted from a bundle, the total bundle statistics changes because the statistics that belongs to the detached link is lost.
- The QoS policy that is applied on bundle is inherited to all its member links and the reference bandwidth that is used to calculate shaper/bandwidth is applied as per the physical member interface bandwidth, and not the bundle as a whole.

Configuration Example

You have to accomplish the following to complete the QoS configuration on link bundles:

Note The policy works only if it is applied on the ingress direction. The egress is supported on COS, DEI and MPLS exp marking. So the below policy may not work when it is applied on egress.

1. Creating a class-map
2. Creating a policy-map and specifying the respective class-map
3. Specifying the action type for the traffic

   Refer Attach a Traffic Policy to an Interface, on page 8 for details on step 1, 2 and 3.
4. Creating a link bundle
5. Applying traffic policy to the link bundle

```
/* Configure a class-map */
Router# configure
Router(config)# class-map match-any c1
Router(config-cmap)# match dscp af11
Router(config-cmap)# match precedence 7
Router(config-cmap)# end-class-map
Router(config)# commit

/* Configure a policy-map and specify the police rate */
Router(config)# policy-map p1
Router(config-pmap)# class c1
Router(config-pmap-c)# set qos-group 5
```

```
Router(config-pmap-c)# police rate percent 50 burst 1600 bytes
Router(config-pmap-c-police)# exit
Router(config-pmap-c)# exit
Router(config-pmap)# class class-default
Router(config-pmap)# end-policy-map
Router(config)# commit

/* Configure Ether-Bundle and apply traffic policy */
Router(config)# interface Bundle-Ether1
Router(config-if)# service-policy input p1
Router(config-if)# ipv4 address 192.1.1.1 255.255.255.0
Router(config-if)# mac-address 1212.1212.1212 (optional)
Router(config-if)# bundle minimum-active links 1
Router(config-if)# commit
```

Running Configuration

This example shows how a traffic policy is applied on an Ethernet link bundle, in the ingress direction. The policy is applied to all interfaces that are members of the Ethernet link bundle.

```
/* Class-map */
configure
 class-map match-any c1
  match dscp af11
  match precedence 7
  end-class-map
 !

/* Policy-map */
policy-map p1
 class c1
  set qos-group 5
  police rate percent 50 burst 1600 bytes
  !
 !
 class class-default
 !
 end-policy-map
!

/* Ether Bundle */
interface Bundle-Ether1
 service-policy input p1
 ipv4 address 192.1.1.1 255.255.255.0
 mac-address 1212.1212.1212
 bundle minimum-active links 1
!
```

Verification

- Verify that the bundle status is UP.

```
router# show bundle bundle-ether 1

Bundle-Ether1
  Status:                                  Up
  Local links <active/standby/configured>:   2 / 0 / 3
  Local bandwidth <effective/available>:     20000000 (20000000) kbps
```

```
  MAC address (source):                       1212.1212.1212 (Configured)
  Inter-chassis link:                         No
  Minimum active links / bandwidth:           1 / 1 kbps
  Maximum active links:                       32
  Wait while timer:                           2000 ms
  Load balancing:                             Default
  LACP:                                       Not operational
    Flap suppression timer:                   Off
    Cisco extensions:                         Disabled
    Non-revertive:                            Disabled
  mLACP:                                      Not configured
  IPv4 BFD:                                   Not configured

  Port                  Device           State        Port ID         B/W, kbps
  --------------------  ---------------  -----------  --------------  ----------
  Te0/0/0/1             Local            Active       0x8000, 0x0000    10000000
      Link is Active
  Te0/0/0/2             Local            Configured   0x8000, 0x0000    10000000
      Link is down
  Te0/0/0/7             Local            Active       0x8000, 0x0000    10000000
```

- Check the statistics of bundle member links.

```
router# show policy-map interface Bundle-Ether 1 input member TenGigE0/0/0/7

Interface:Bundle-Ether1 Member:TenGigE0/0/0/7 input: p1

Class c1
  Classification statistics          (packets/bytes)     (rate - kbps)
    Matched             :               2759/176576               0
    Transmitted         : N/A
    Total Dropped       :                   0/0                   0
  Policing statistics                (packets/bytes)     (rate - kbps)
    Policed(conform)    :               2759/176576               0
    Policed(exceed)     :                   0/0                   0
    Policed(violate)    :                   0/0                   0
    Policed and dropped :                   0/0
Class class-default
  Classification statistics          (packets/bytes)     (rate - kbps)
    Matched             :                255/16372                0
    Transmitted         : N/A
    Total Dropped       : N/A
```

- Verify that the bundle statistics show the cumulative of member link 1 and 2.

```
router# show policy-map interface bundle-ether 1

Tue Feb 11 11:57:22.532 UTC

Bundle-Ether1 input: p1

Class c1
  Classification statistics          (packets/bytes)     (rate - kbps)
    Matched             :               7588/485632               0
    Transmitted         : N/A
    Total Dropped       :                   0/0                   0
  Policing statistics                (packets/bytes)     (rate - kbps)
    Policed(conform)    :               7588/485632               0
    Policed(exceed)     :                   0/0                   0
    Policed(violate)    :                   0/0                   0
    Policed and dropped :                   0/0
```

```
Class class-default
  Classification statistics          (packets/bytes)     (rate - kbps)
    Matched             :                 257/16552                0
    Transmitted         : N/A
    Total Dropped       : N/A
Bundle-Ether1 direction output: Service Policy not installed
```

Related Topics

- QoS on Link Bundles, on page 37

Associated Commands

- bundle maximu-active links
- interface Bundle-Ether