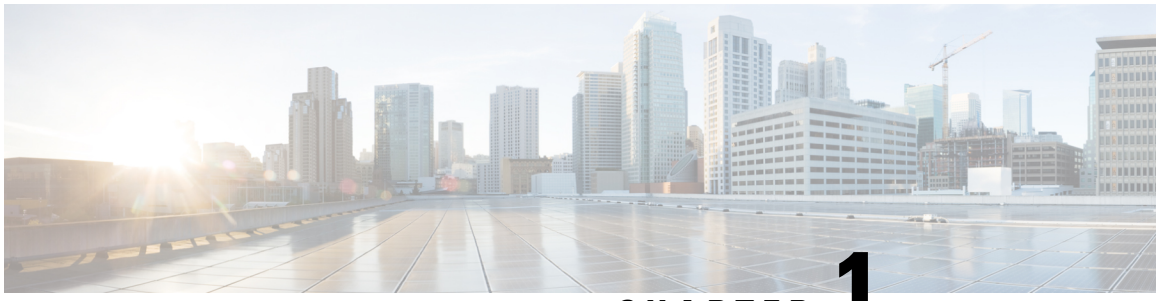
QoS: Classification Configuration Guide, Cisco IOS XE Everest 3.18SP (Cisco NCS 4200 Series)

First Published: 2016-07-29

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

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.

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

- Cos Marking is *not* supported for pop 0.
- IPv6 classification and marking are *not* supported on the Cisco RSP3 Module.
- You cannot configure QoS with empty class map and cannot attach a policy without any class map match condition.

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:

- Discard-class value
- DSCP value in the type of service (ToS) byte
- MPLS EXP field value in the topmost label on an input 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

Table 1: Feature History

Feature Name	Release	Description
DSCP Preservation of MLDP Traffic	Cisco IOS XE Amsterdam 17.1.1	The Differentiated Services Code Point (DSCP) value does not change on both the uniform and pipe modes.

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:

- The DSCP field (TAG to IP) value does not change in both the uniform mode and in pipe mode. This is applicable to both the Unicast and Multicast traffic scenario.
- 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 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.
- Weighted random early detection (WRED) uses precedence values or DSCP values to determine the probability that the traffic will be dropped. Therefore, the Precedence and DSCP can be used along with WRED.

How to Mark 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.

This method is further described in the section that follows.

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 2: 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 discard-class	discard-class value	Layer 2 **Note** **set discard-class** supports Layer 2 and Layer 3 on the Cisco RSP3 Module.	
set dscp	DSCP value in the ToS byte	Layer 3	IP
set mpls experimental imposition	MPLS EXP field on all imposed label entries	Layer 3	MPLS

set Commands[1]	Traffic Attribute	Network Layer	Protocol
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

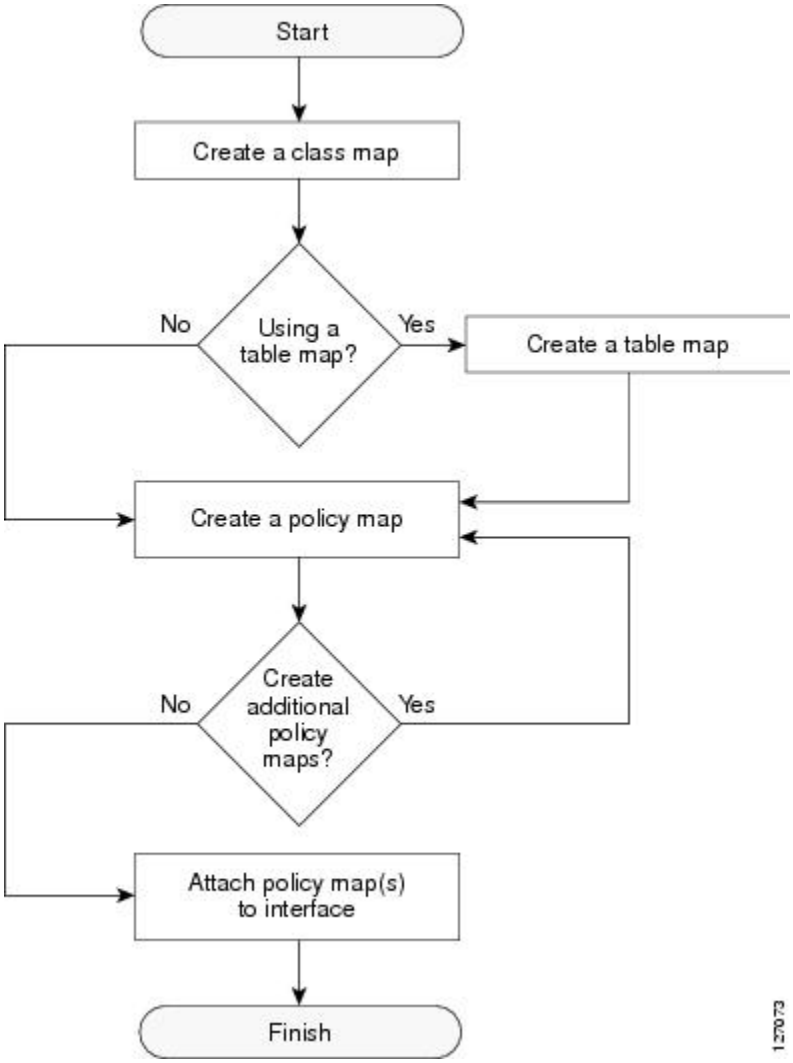
Note The **set qos-group** can be used for L2 traffic on the Cisco RSP3 Module.

Note **set dscp** command is *not* supported on the Cisco RSP3 Module for L2 EFP configuration.

Traffic Marking Procedure Flowchart

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

Figure 1: 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 3: 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	
set discard-class	discard-class value	Layer 2	
set dscp	DSCP value in the ToS byte	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, EFP, Trunk EFP, or Xconect 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 4: 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

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 **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 cos** *cos-value*

Example:

```
Router (config)# match cos 1
```

Matches with Cos value.

cos-value: Sets the Cos Value. The valid values are 1 and 2.

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

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.

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
```

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, EFP or Xconnect

Before you begin

Note Depending on the needs of your network, policy maps can be attached to targets that are supported. For information, see .

Step 1 **configure terminal**

Enter global configuration mode.

Example:

```
Router# configure terminal
```

Step 2 **interface** *interface-id*

Specify the port to attach to the policy map, and enter interface configuration mode. Valid interfaces are physical ports.

Example:

```
Router(config)# interface gigabitethernet 0/3/6
```

Step 3 **service instance** *number* **ethernet** [*name*]

Configure an EFP (service instance) and enter service instance configuration) mode.

- The number is the EFP identifier, an integer from 1 to 4000.
- (Optional) **ethernet** name is the name of a previously configured EVC. You do not need to use an EVC name in a service instance.

Example:

```
Rotuer(config)# service instance 1 ethernet
```

Step 4 **service-policy** {**input** | **output**} *policy-map-name*

Attaches the specified policy map to the input or output interfaces .

- *policy-map-name*

  : Specifies the policy map.

Example:

```
Router(config-if-srv)# service-policy input co1
```

Step 5 **encapsulation** {**default** | **dot1q** | **priority-tagged** | **untagged**}

Configure encapsulation type for the service instance.

- **default**—Configure to match all unmatched packets.
- **dot1q**—Configure 802.1Q encapsulation.
- **priority-tagged**—Specify priority-tagged frames, VLAN-ID 0 and CoS value of 0 to 7.
- **untagged**—Map to untagged VLANs. Only one EFP per port can have untagged encapsulation.

Example:

```
Router(config-if-srv)# encapsulation dot1q 1
```

Step 6 **bridge-domain** *bridge-id* [**split-horizon group** *group-id*]

Configure the bridge domain ID. The range is from 1 to 4000.

You can use the **split-horizon** keyword to configure the port as a member of a split horizon group. The *group-id* range is from 0 to 2.

Example:

```
Router(config-if-srv)# bridge-domain 1
```

Step 7 **end**

Return to privileged EXEC mode.

Example:

```
Router(config-if-srv)# end
```

Configuration Example

```
Router(config)# interface gigabitethernet 0/3/6
Router(config-if)# service instance 1 ethernet
Router(config-if-srv)# service-policy input co1
Router(config-if-srv)# encapsulation dot1q 1
Router(config-if-srv)# bridge-domain 1
Router(config-if-srv)# end
```

Configuration Examples for Marking Network Traffic

Example: Creating a Class Map for Marking Network Traffic

- The following is an example of configures a class map with using match-any .

```
Router> enable
Router# configure terminal
Router(config)# interface gigabitethernet0/3/6
Router(config-if)# service instance 1 ethernet
Router(config-if-srv)# encapsulation dot1q 1
Router(config-if-srv)# bridge-domain 1
```

```
Device(config)# class-map match-any class1
Device(config-cmap)# match cos 1
Device(config-cmap)# end
```

- The following is an example of configures a class map with using match-all .

```
Router> enable
Router# configure terminal
Router(config)# interface gigabitethernet0/3/6
Router(config-if)# service instance 1 ethernet
Router(config-if-srv)# encapsulation dot1q 1
Router(config-if-srv)# bridge-domain 1
Device(config)# class-map match-all class1
Device(config-cmap)# match cos 1
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.

```
Router> enable
Router# configure terminal
Router(config)# policy-map policy1
Router(config-pmap)# class class1
Router(config-pmap-c)# set cos 2
Router(config-pmap-c)# end
Router# exit
```

Example: Attaching a Traffic Policy to an Interface

The following example shows how to attach an existing traffic policy to an interface. After you define a traffic policy with the **policy-map** command, you can attach it to one or more interfaces by using the **service-policy** command in interface configuration mode. Although you can assign the same traffic policy to multiple interfaces, each interface can have only one traffic policy attached in the input direction and only one traffic policy attached in the output direction.

```
Router(config)# interface gigabitethernet0/3/6
Router(config-if)# service instance 1 ethernet
Router(config-if-srv)# service-policy input co1
Router(config-if-srv)# encapsulation dot1q 1
Router(config-if-srv)# bridge-domain 1
Router(config-if)# service-policy input policy1
Router(config-if)# end
```

Additional References for Marking Network Traffic

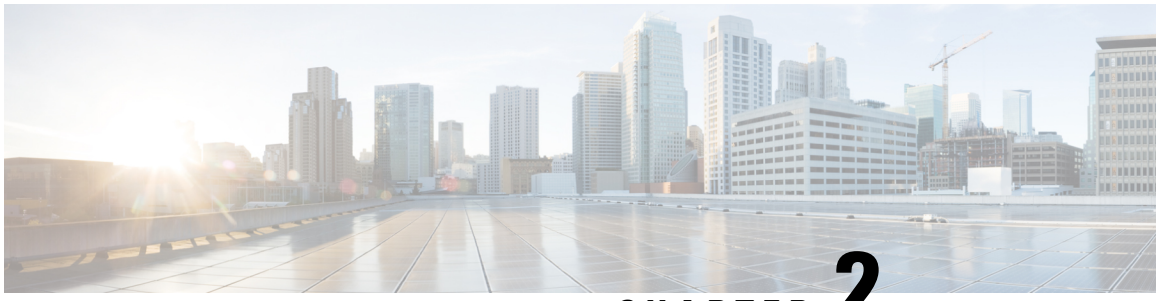
Related Documents

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*

Related Topic	Document Title
MQC	"Applying QoS Features Using the MQC" module
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



CHAPTER 2

Classifying and Marking MPLS EXP

The QoS EXP Matching feature allows you to classify and mark network traffic by modifying the Multiprotocol Label Switching (MPLS) experimental bits (EXP) field in IP packets. This module contains conceptual information and the configuration tasks for classifying and marking network traffic using the MPLS EXP field.

Prerequisites for Classifying and Marking MPLS EXP

- The router must be configured as an MPLS provider edge (PE) or provider (P) router, which can include the configuration of a valid label protocol and underlying IP routing protocols.

Restrictions for Classifying and Marking MPLS EXP

- MPLS classification and marking can only occur in an operational MPLS Network.
- MPLS EXP classification and marking is supported on the main router interfaces for MPLS packet switching and imposition (simple IP imposition and Ethernet over MPLS (EoMPLS) imposition) and on Ethernet virtual circuits (EVCs) or Ethernet flow points (EFPs) for EoMPLS imposition.
- MPLS EXP classification or marking for bridged MPLS packets on EVCs or EFPs is not supported.
- MPLS EXP marking is supported only in the ingress direction.
- If a packet is classified by IP type of service (ToS) or class of service (CoS) at ingress, it cannot be reclassified by MPLS EXP at egress (imposition case). However, if a packet is classified by MPLS at ingress it can be reclassified by IP ToS, CoS, or Quality of Service (QoS) group at egress (disposition case).
- If a packet is encapsulated in MPLS, the MPLS payload cannot be checked for other protocols such as IP for classification or marking. Only MPLS EXP marking affects packets encapsulated by MPLS.

Information About Classifying and Marking MPLS EXP

Classifying and Marking MPLS EXP Overview

The QoS EXP Matching feature allows you to organize network traffic by setting values for the MPLS EXP field in MPLS packets. By choosing different values for the MPLS EXP field, you can mark packets so that packets have the priority that they require during periods of congestion. Setting the MPLS EXP value allows you to:

- Classify traffic

  The classification process selects the traffic to be marked. Classification accomplishes this by partitioning traffic into multiple priority levels, or classes of service. Traffic classification is the primary component of class-based QoS provisioning. For more information, see the “Classifying Network Traffic” module.

- Police and mark traffic

  Policing causes traffic that exceeds the configured rate to be discarded or marked to a different drop level. Marking traffic is a way to identify packet flows to differentiate them. Packet marking allows you to partition your network into multiple priority levels or classes of service. For more information, see the “Marking Network Traffic” module.

MPLS Experimental Field

The MPLS experimental bits (EXP) field is a 3-bit field in the MPLS header that you can use to define the QoS treatment (per-hop behavior) that a node should give to a packet. In an IP network, the DiffServ Code Point (DSCP) (a 6-bit field) defines a class and drop precedence. The EXP bits can be used to carry some of the information encoded in the IP DSCP and can also be used to encode the dropping precedence.

By default, Cisco IOS Software copies the three most significant bits of the DSCP or the IP precedence of the IP packet to the EXP field in the MPLS header. This action happens when the MPLS header is initially imposed on the IP packet. However, you can also set the EXP field by defining a mapping between the DSCP or IP precedence and the EXP bits. This mapping is configured using the **set mpls experimental** or **police** commands. For more information, see the “How to Classify and Mark MPLS EXP” section.

Benefits of MPLS EXP Classification and Marking

If a service provider does not want to modify the value of the IP precedence field in packets transported through the network, they can use the MPLS EXP field value to classify and mark IP packets.

By choosing different values for the MPLS EXP field, you can mark critical packets so that those packets have priority if network congestion occurs.

How to Classify and Mark MPLS EXP

Classifying MPLS Encapsulated Packets

Note MPLS EXP topmost classification is not supported for bridged MPLS packets on Ethernet virtual circuits (EVC) or Ethernet flow points (EFP).

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map** [**match-all** | **match-any**] *class-map-name*
4. **match mpls experimental topmost** *mpls-exp-value*
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	**class-map** [**match-all**	**match-any**] *class-map-name* **Example:** `Router(config)# class-map exp3`	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 mpls experimental topmost** *mpls-exp-value* **Example:** `Router(config-cmap)# match mpls experimental topmost 3`	Specifies the match criteria. **Note** The **match mpls experimental topmost** command classifies traffic on the basis of the EXP value in the topmost label header.
Step 5	**end** **Example:** `Router(config-cmap)# end`	(Optional) Returns to privileged EXEC mode.

Marking MPLS EXP on All Imposed Labels

Perform this task to set the value of the MPLS EXP field on all imposed label entries.

Before you begin

The router supports MPLS EXP marking only in the ingress direction.

In typical configurations, marking MPLS packets at imposition is used with ingress classification on IP ToS or CoS fields. However, generic matching with the class default value is supported with other ingress attributes such as **vlan**.

Note For IP imposition marking, the IP precedence value is copied to the MPLS EXP value by default.

Note For EVC configuration, a policy map that performs matching based on the CoS and that sets the EXP imposition value should be used to copy CoS values to the EXP value.

Note The **set mpls experimental imposition** command works only on packets that have new or additional MPLS labels added to them.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map** *policy-map-name*
4. **class** *class-map-name*
5. **set mpls experimental imposition** *mpls-exp-value*
6. **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.

	Command or Action	Purpose
Step 3	**policy-map** *policy-map-name* **Example:** `Router(config)# policy-map mark-up-exp-2`	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-map-name* **Example:** `Router(config-pmap)# class prec012`	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 5	**set mpls experimental imposition** *mpls-exp-value* **Example:** `Router(config-pmap-c)# set mpls experimental imposition 2`	Sets the value of the MPLS EXP field on all imposed label entries.
Step 6	**end** **Example:** `Router(config-pmap-c)# end`	(Optional) Returns to privileged EXEC mode.

Marking MPLS EXP on Label Switched Packets

Perform this task to set the MPLS EXP field on label switched packets.

Before you begin

Note The **set mpls experimental topmost** command works only on packets that are already MPLS encapsulated.

Note The router supports MPLS EXP marking in the ingress direction only, and does not support MPLS EXP classification or marking for bridged MPLS packets on EVCs or EFPs.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map** *policy-map-name*
4. **class** *class-map-name*
5. **set mpls experimental topmost** *mpls-exp-value*
6. **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 mark-up-exp-2`	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-map-name* **Example:** `Router(config-pmap)# class-map exp012`	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 5	**set mpls experimental topmost** *mpls-exp-value* **Example:** `Router(config-pmap-c)# set mpls experimental topmost 2`	Sets the MPLS EXP field value in the topmost label on the output interface.
Step 6	**end** **Example:** `Router(config-pmap-c)# end`	(Optional) Returns to privileged EXEC mode.

Configuring Conditional Marking

To conditionally set the value of the MPLS EXP field on all imposed label, perform the following task:

Before you begin

Note The **set-mpls-exp-topmost-transmit** action affects MPLS encapsulated packets only. The **set-mpls-exp-imposition-transmit** action affects any new labels that are added to the packet.

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **policy-map** *policy-map-name*
4. **class** *class-map-name*
5. **police cir** *bps* **bc pir** *bps* **be**
6. **conform-action** [**set-mpls-exp-imposition-transmit** *mpls-exp-value* | **set-mpls-exp-topmost-transmit** *mpls-exp-value*]
7. **exceed-action** [**set-mpls-exp-imposition-transmit** *mpls-exp-value* | **set-mpls-exp-topmost-transmit** *mpls-exp-value*]
8. **violate-action drop**
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 ip2tag`	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-map-name* **Example:** `Router(config-pmap)# class iptcp`	Creates a class map to be used for matching traffic to a specified class, and enters policy-map class configuration mode. • Enter the class map name.
Step 5	**police cir** *bps* **bc pir** *bps* **be** **Example:** `Router(config-pmap-c)# police cir 1000000 pir 2000000`	Defines a policer for classified traffic and enters policy-map class police configuration mode.
Step 6	**conform-action** [**set-mpls-exp-imposition-transmit** *mpls-exp-value* | **set-mpls-exp-topmost-transmit** *mpls-exp-value*] **Example:** `Router(config-pmap-c-police)# conform-action set-mpls-exp-imposition-transmit 3`	Defines the action to take on packets that conform to the values specified by the policer. • In this example, if the packet conforms to the committed information rate (cir) or is within the conform burst (bc) size, the MPLS EXP field is set to 3.

	Command or Action	Purpose
Step 7	**exceed-action** [**set-mpls-exp-imposition-transmit** *mpls-exp-value* | **set-mpls-exp-topmost-transmit** *mpls-exp-value*] **Example:** `Router(config-pmap-c-police)# exceed-action set-mpls-exp-imposition-transmit 2`	Defines the action to take on packets that exceed the values specified by the policer. • In this example, if the packet exceeds the cir rate and the bc size, but is within the peak burst (be) size, the MPLS EXP field is set to 2.
Step 8	**violate-action drop** **Example:** `Router(config-pmap-c-police)# violate-action drop`	Defines the action to take on packets whose rate exceeds the peak information rate (pir) and is outside the bc and be ranges. • You must specify the exceed action before you specify the violate action. • In this example, if the packet rate exceeds the pir rate and is outside the bc and be ranges, the packet is dropped.
Step 9	**end** **Example:** `Router(config-pmap-c-police)# end`	(Optional) Returns to privileged EXEC mode.

Configuration Examples for Classifying and Marking MPLS EXP

Example: Classifying MPLS Encapsulated Packets

Defining an MPLS EXP Class Map

The following example defines a class map named exp3 that matches packets that contains MPLS experimental value 3:

```
Router(config)# class-map exp3
Router(config-cmap)# match mpls experimental topmost 3
Router(config-cmap)# exit
```

Defining a Policy Map and Applying the Policy Map to an Ingress Interface

The following example uses the class map created in the example above to define a policy map. This example also applies the policy map to a physical interface for ingress traffic.

```
Router(config)# policy-map change-exp-3-to-2
Router(config-pmap)# class exp3
Router(config-pmap-c)# set mpls experimental topmost 2
Router(config-pmap)# exit
Router(config)# interface GigabitEthernet 0/0/0
```

```
Router(config-if)# service-policy input change-exp-3-to-2
Router(config-if)# exit
```

Defining a Policy Map and Applying the Policy Map to an Egress Interface

The following example uses the class map created in the example above to define a policy map. This example also applies the policy map to a physical interface for egress traffic.

```
Router(config)# policy-map WAN-out
Router(config-pmap)# class exp3
Router(config-pmap-c)# shape average 10000000
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# interface GigabitEthernet 0/0/0
Router(config-if)# service-policy output WAN-out
Router(config-if)# exit
```

Example: Marking MPLS EXP on All Imposed Labels

Defining an MPLS EXP Imposition Policy Map

The following example defines a policy map that sets the MPLS EXP imposition value to 2 based on the IP precedence value of the forwarded packet:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# class-map prec012
Router(config-cmap)# match ip prec 0 1 2
Router(config-cmap)# exit
Router(config)# policy-map mark-up-exp-2
Router(config-pmap)# class prec012
Router(config-pmap-c)# set mpls experimental imposition 2
Router(config-pmap-c)# exit
Router(config-pmap)# exit
```

Applying the MPLS EXP Imposition Policy Map to a Main Interface

The following example applies a policy map to Gigabit Ethernet interface 0/0/0:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# interface GigabitEthernet 0/0/0
Router(config-if)# service-policy input mark-up-exp-2
Router(config-if)# exit
```

Applying the MPLS EXP Imposition Policy Map to an EVC

The following example applies a policy map to the Ethernet Virtual Connection specified by the **service instance** command:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# interface GigabitEthernet 0/0/0
Router(config-inf)# service instance 100 ethernet
```

```
Router(config-if-srv)# xconnect 100.0.0.1 encapsulation mpls 100
Router(config-if-srv)# service-policy input mark-up-exp-2
Router(config-if-srv)# exit
Router(config-if)# exit
```

Example: Marking MPLS EXP on Label Switched Packets

Defining an MPLS EXP Label Switched Packets Policy Map

The following example defines a policy map that sets the MPLS EXP topmost value to 2 according to the MPLS EXP value of the forwarded packet:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# class-map exp012
Router(config-cmap)# match mpls experimental topmost 0 1 2
Router(config-cmap)# exit
Router(config-cmap)# policy-map mark-up-exp-2
Router(config-pmap)# class exp012
Router(config-pmap-c)# set mpls experimental topmost 2
Router(config-pmap-c)# exit
Router(config-pmap)# exit
```

Applying the MPLS EXP Label Switched Packets Policy Map to a Main Interface

The following example shows how to apply the policy map to a main interface:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# interface GigabitEthernet 0/0/0
Router(config-if)# service-policy input mark-up-exp-2
Router(config-if)# exit
```

Example: Configuring Conditional Marking

The example in this section creates a policer for the **iptcp** class, which is part of the **ip2tag** policy map, and attaches the policy map to the Gigabit Ethernet interface.

```
Router(config)# policy-map ip2tag
Router(config-pmap)# class iptcp
Router(config-pmap-c)# police cir 1000000 pir 2000000
Router(config-pmap-c-police)# conform-action set-mpls-exp-imposition-transmit 3
Router(config-pmap-c-police)# exceed-action set-mpls-exp-imposition-transmit 2
Router(config-pmap-c-police)# violate-action drop
Router(config-pmap-c-police)# exit
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# interface GigabitEthernet 0/0/1
Router(config-if)# service-policy input ip2tag
```

Additional References

Related Documents

Related Topic	Document Title
QoS commands	*Cisco IOS Quality of Service Solutions Command Reference*
Classifying network traffic	"Classifying Network Traffic" module
Marking network traffic	"Marking Network Traffic" module

Standards and RFCs

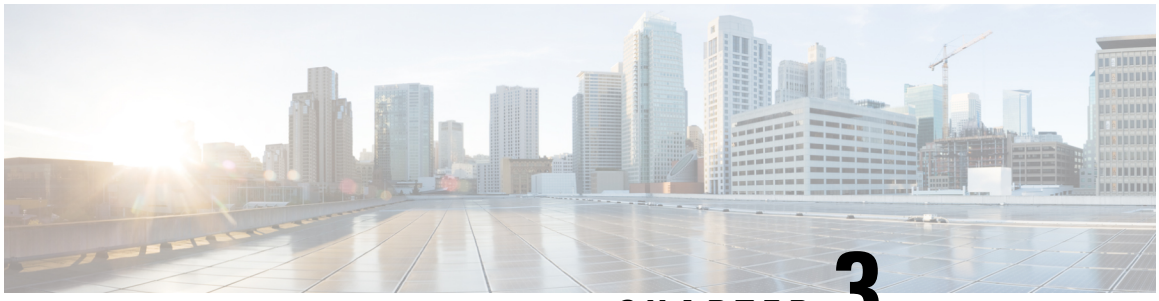
Standard/RFC	Title
No new or modified standards are supported, and support for existing standards has not been modified.	

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 software 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



CHAPTER 3

Priority Shaper

Earlier, when the priority of a queue at Per-Hop Behavior (PHB) was propagated all the way up the hierarchy towards the channel level, the PHB classes that had priority at PHB level would only be prioritized over other classes of subchannels. To avoid this, Priority Shaper feature is implemented.

Priory Shaper feature helps to balance the packet drops between the streams when multiple steams egress out of a priority queue. Egress QoS policy is supported on Priority Shaper.

Restrictions for Priority Shaper

- Priority Shaper is supported only for PHB level classes.
- Egress QoS Policy map with Priority Shaper can be applied only on the member interface of port channel and not at the logical level.
- Policer configuration is not supported with the Priority Shaper configuration under same class map.
- Priority Traffic Latency is increased during congestion with Priority Shaper configuration at Q level. Configure the queue limit with a lesser value for the priority queue to reduce the latency of priority traffic.
- If the packet is from a 10G interface to a 1G interface, the burstiness is introduced. Due to this, dequeuing rate of this strict priority queue may be sometimes more than enqueuing. As a result, very few packet counters are seen in other queues.

Configuring Priority Shaper

Perform the following steps to configure Priority Shaper.

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 class_priority
```

Configures class map and specifies the name of the class map to be created.

Step 4 **policy-map** *policy-map-name*

Example:

```
Device(config)#policy-map shape_priority
```

Configures the policy map.

Step 5 **class** *class-map-name*

Example:

```
Device(config-pmap)#class class_priority
```

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 that is created earlier.

Step 6 **priority level** <*level 1/2* > **percent** <*percentage 1-100* > or **priority level** <**level 1/2**> <*kbps*> <*burst size*>

Example:

```
Device(config-pmap-c)# priority <1-10000000> Kilo Bits per second
Device(config-pmap-c)# priority Percent <1-100>
Device(config-pmap-c)# priority level <1-2> <1-10000000> Kilo Bits per second
Device(config-pmap-c)# priority level <1-2> percent <1-100>
```

Assigns priority to a traffic class at the priority level specified.

Note **level** is the level of priority assigned to the priority class. Valid values are 1 (high priority) and 2 (low priority). The default value is 1. Do not specify the same priority level for two different classes in the same policy map.

Step 7 **interface** *interface-type interface-number*

Example:

```
Device(config)# interface gigabitethernet 0/0/1
```

Specifies the port to attach to the policy map and allows to enter the interface configuration mode. Valid interfaces are physical ports.

Step 8 **service-policy output** *policy-map-name*

Example:

```
Device(config-if)# service instance 1 ethernet
Device(config-if-srv)# service-policy output shape_priority
```

Applies output policy to the interface.

Note You can also attach the service policy over the service instance.

Step 9 end

Example:

```
Device(config)#end
```

Returns to privileged EXEC mode.

Configuration Examples for Priority Shaper

This section shows sample configurations for Priority Shaper.

Example: Configuring Priority Shaper

Verifying Priority Shaper

Use the following command to verify that the Priority Shaper feature is configured on your interface.