

# QoS Classification and Marking on Catalyst 6500/6000 Series Switches That Run Cisco IOS Software

Document ID: 24055

## Contents

### Introduction

#### Prerequisites

- Requirements
- Components Used
- Conventions
- Terminology

#### Input Port Handling

Switching Engine (PFC)

Configure the Service Policy to Classify or Mark a Packet in Cisco IOS Software Release 12.1(12c)E and Later

Configure the Service Policy to Classify or Mark a Packet in Cisco IOS Software Releases Earlier Than Cisco IOS Software Release 12.1(12c)E

Four Possible Sources for Internal DSCP

How Is the Internal DSCP Chosen?

#### Output Port Handling

#### Notes and Limitations

The Default ACL

Limitations of the WS-X61xx, WS-X6248-xx, WS-X6224-xx, and WS-X6348-xx Line Cards

Packets That Come from the MSFC1 or MSFC2 on Supervisor Engine 1A/PFC

Summary of Classification

#### Monitor and Verify a Configuration

Check the Port Configuration

Check Defined Classes

Check the Policy Map That Is Applied to an Interface

#### Sample Case Studies

Case 1: Marking at the Edge

Case 2: Trusting in the Core with Only Gigabit Ethernet Interfaces

#### Related Information

## Introduction

This document examines what happens with regard to the marking and classification of a packet at various stages within the Cisco Catalyst 6500/6000 chassis that runs Cisco IOS® Software. This document describes special cases and restrictions, and it provides short case studies.

This document does not provide an exhaustive list of all Cisco IOS Software commands that relate to QoS or marking. For more information on the Cisco IOS Software command-line interface (CLI), refer to *Configuring PFC QoS*.

# Prerequisites

## Requirements

There are no specific requirements for this document.

## Components Used

The information in this document is based on these hardware versions:

- Catalyst 6500/6000 series switches that run Cisco IOS Software and use one of these Supervisor Engines:
  - ◆ A Supervisor Engine 1A with a Policy Feature Card (PFC) and a Multilayer Switch Feature Card (MSFC)
  - ◆ A Supervisor Engine 1A with a PFC and an MSFC2
  - ◆ A Supervisor Engine 2 with a PFC2 and an MSFC2

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

## Conventions

Refer to Cisco Technical Tips Conventions for more information on document conventions.

## Terminology

The list provides terminology that this document uses:

- Differentiated services code point (DSCP) The first six bits of the type of service (ToS) byte in the IP header. DSCP is only present in the IP packet.

**Note:** The switch also assigns an internal DSCP to every packet, whether IP or non-IP. The Four Possible Sources for Internal DSCP section of this document details this internal DSCP assignment.

- IP precedence The first three bits of the ToS byte in the IP header.
- Class of service (CoS) The only field that can be used to mark a packet at Layer 2 (L2). CoS consists of any of these three bits:

- ◆ The three IEEE 802.1p (dot1p) bits in the IEEE 802.1Q (dot1q) tag for the dot1q packet.

**Note:** By default, Cisco switches do not tag native VLAN packets.

- ◆ The three bits called "User Field" in the Inter-Switch Link (ISL) header for an ISL-encapsulated packet.

**Note:** CoS is not present inside a non-dot1q or an ISL packet.

- Classification The process that is used to select the traffic to be marked.
- Marking The process that sets a Layer 3 (L3) DSCP value in a packet. This document extends the definition of marking to include the setting of L2 CoS values.

Catalyst 6500/6000 series switches can make classifications on the basis of these three parameters:

- DSCP

- IP precedence
- CoS

The Catalyst 6500/6000 series switches perform classification and marking at various stages. This is what occurs at different places:

- Input port (ingress application-specific integrated circuit [ASIC])
- Switching engine (PFC)
- Output port (egress ASIC)

## Input Port Handling

The main configuration parameter for the ingress port, with regard to classification, is the `trust` state of the port. Each port of the system can have one of these `trust` states:

- `trust-ip-precedence`
- `trust-dscp`
- `trust-cos`
- `untrusted`

In order to set or change the port `trust` state, issue this Cisco IOS Software command in interface mode:

```
6k(config-if)#mls qos trust ?
cos          cos keyword
dscp         dscp keyword
ip-precedence ip-precedence keyword
<cr>
```

**Note:** By default, all ports are in the `untrusted` state when QoS is enabled. In order to enable QoS on the Catalyst 6500 that runs Cisco IOS Software, issue the `mls qos` command in the main configuration mode.

At the input port level, you can also apply a default CoS per port. Here is an example:

```
6k(config-if)#mls qos cos cos-value
```

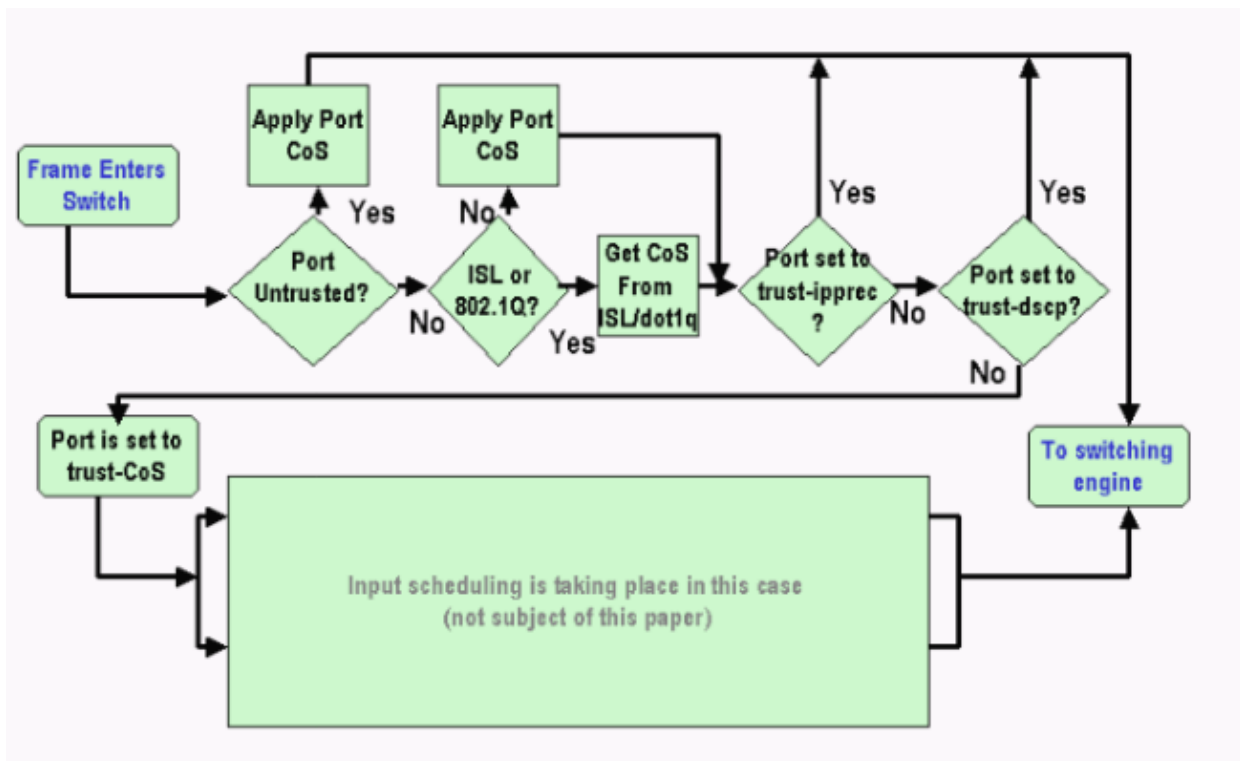
This default CoS applies to all packets, such as IP and Internetwork Packet Exchange (IPX). You can apply the default CoS to any physical port.

If the port is in the `untrusted` state, mark the frame with the port default CoS and pass the header to the switching engine (PFC). If the port is set to one of the `trust` states, perform one of these two options:

- If the frame does not have a received CoS (dot1q or ISL), apply the default port CoS.
- For dot1q and ISL frames, keep the CoS as it is.

Then, pass the frame to the switching engine.

This example illustrates the input classification and marking. The example shows how to assign an internal CoS to each frame:



**Note:** As this example shows, each frame is assigned an internal CoS. The assignment is based on either the received CoS or the default port CoS. The internal CoS includes untagged frames that do not carry any real CoS. The internal CoS is written in a special packet header, which is called a data bus header, and sent over the data bus to the switching engine.

## Switching Engine (PFC)

When the header reaches the switching engine, the switching engine Enhanced Address Recognition Logic (EARL) assigns each frame an internal DSCP. This internal DSCP is an internal priority that is assigned to the frame by the PFC as the frame transits the switch. This is not the DSCP in the IP version 4 (IPv4) header. The internal DSCP is derived from an existing CoS or ToS setting and is used to reset the CoS or ToS as the frame exits the switch. This internal DSCP is assigned to all frames that are switched or routed by the PFC, even non-IP frames.

This section discusses how you can assign a service policy to the interface in order to make a marking. The section also discusses the final setting of the internal DSCP, which depends on the port `trust` state and the service policy that is applied.

## Configure the Service Policy to Classify or Mark a Packet in Cisco IOS Software Release 12.1(12c)E and Later

Complete these steps in order to configure the service policy:

1. Configure an access control list (ACL) to define the traffic that you want to consider.

The ACL can be numbered or named, and the Catalyst 6500/6000 supports an extended ACL. Issue the `access-list xxx` Cisco IOS Software command, as this example shows:

```
(config)#access-list 101 permit ip any host 10.1.1.1
```

2. Configure a traffic class (class map) to match the traffic on the basis of the ACL that you have defined or on the basis of the received DSCP.

Issue the **class-map** Cisco IOS Software command. PFC QoS does not support more than one match statement per class map. Also, PFC QoS only supports these match statements:

- ◆ **match ip access-group**
- ◆ **match ip dscp**
- ◆ **match ip precedence**
- ◆ **match protocol**

**Note:** The **match protocol** command enables the use of Network Based Application Recognition (NBAR) to match traffic.

**Note:** Of these options, only the **match ip dscp** and **match ip precedence** statements are supported and work. These statements, however, are not useful in the marking or classification of the packets. You can use these statements, for example, to make policing on all packets that match a certain DSCP. However, this action is beyond the scope of this document.

```
(config)#class-map class-name  
  
(config-cmap)#match {access-group | input-interface | ip dscp}
```

**Note:** This example shows only three options for the **match** command. But you can configure many more options at this command prompt.

**Note:** Any one of the options in this **match** command is taken for match criteria and the other options are left out, according to the incoming packets.

Here is an example:

```
class-map match-any TEST  
  match access-group 101  
  
class-map match-all TEST2  
  match ip precedence 6
```

3. Configure a policy map to apply a policy to a class that you previously defined.

The policy map contains:

- ◆ A name
- ◆ A set of class statements
- ◆ For each class statement, the action that needs to be taken for that class

The supported actions in PFC1 and PFC2 QoS are:

- ◆ **trust dscp**
- ◆ **trust ip precedence**
- ◆ **trust cos**
- ◆ **set ip dscp** in Cisco IOS Software Release 12.1(12c)E1 and later
- ◆ **set ip precedence** in Cisco IOS Software Release 12.1(12c)E1 and later
- ◆ **police**

**Note:** This action is beyond the scope of this document.

```
(config)#policy-map policy-name  
  
(config-pmap)#class class-name  
  
(config-pmap-c)#{police | set ip dscp}
```

**Note:** This example shows only two options, but you can configure many more options at this

(config-pmap-c) # command prompt.

Here is an example:

```
policy-map test_policy
class TEST
    trust ip precedence
class TEST2
    set ip dscp 16
```

4. Configure a service policy input to apply a policy map that you previously defined to one or more interface.

**Note:** You can attach a service policy to either the physical interface or to the switched virtual interface (SVI) or VLAN interface. If you attach a service policy to a VLAN interface, the only ports that use this service policy are ports that belong to that VLAN and are configured for VLAN-based QoS. If the port is not set for VLAN-based QoS, the port still uses the default port-based QoS and only looks at the service policy that is attached to the physical interface.

This example applies the service policy `test_policy` to the port Gigabit Ethernet 1/1:

```
(config) interface gigabitethernet 1/1
(config-if)#service-policy input test_policy
```

This example applies the service policy `test_policy` to all ports in VLAN 10 that have a VLAN-based configuration from the QoS point of view:

```
(config) interface gigabitethernet 1/2
(config-if)#switchport mode access
(config-if)#switchport access vlan 10
(config-if)#mls qos vlan-based
(config-if)#exit
(config-if)#interface vlan 10
(config-if)#service-policy input test_policy
```

**Note:** You can combine Step 2 and Step 3 of this procedure if you skip the specific definition of the class and attach the ACL directly in the definition of the policy map. In this example, where the class `TEST police` has not been defined prior to the configuration of the policy map, the class is defined within the policy map:

```
(config)#policy-map policy-name

(config-pmap)#class class_name {access-group acl_index_or_name |
    dscp dscp_1 [dscp_2 [dscp_N]] | precedence ipp_1 [ipp_2 [ipp_N]]}

!--- Note: This command should be on one line.

policy-map TEST
class TEST police access-group 101
```

## Configure the Service Policy to Classify or Mark a Packet in Cisco IOS Software Releases Earlier Than Cisco IOS Software Release 12.1(12c)E

In Cisco IOS Software releases earlier than Cisco IOS Software Release 12.1(12c)E1, you cannot use the `set ip dscp` or `set ip precedence` action in a policy map. Therefore, the only way to make a marking of specific traffic that a class defines is to configure a policer with a very high rate. This rate should be, for example, at least the line rate of the port or something high enough to allow all the traffic to hit that policer. Then, use `set-dscp-transmit xx` as the conform action. Follow these steps in order to set up this configuration:

1. Configure an ACL to define the traffic that you want to consider.

The ACL can be numbered or named, and the Catalyst 6500/6000 supports an extended ACL. Issue the **access-list** *xxx* Cisco IOS Software command, as this example shows:

```
(config)#access-list 101 permit ip any host 10.1.1.1
```

2. Configure a traffic class (class map) to match the traffic on the basis of either the ACL that you have defined or on the basis of the received DSCP.

Issue the **class-map** Cisco IOS Software command. PFC QoS does not support more than one match statement per class map. Also, PFC QoS only supports these match statements:

- ◆ **match ip access-group**
- ◆ **match ip dscp**
- ◆ **match ip precedence**
- ◆ **match protocol**

**Note:** The **match protocol** command enables the use of NBAR to match traffic.

**Note:** Of these statements, only the **match ip dscp** and **match ip precedence** statements are supported and work. These statements, however, are not useful in marking or the classification of the packets. You can use these statements, for example, to make policing on all packets that match a certain DSCP. However, this action is beyond the scope of this document.

```
(config)#class-map class-name
```

```
(config-cmap)#match {access-group | input-interface | ip dscp}
```

**Note:** This example shows only three options for the **match** command. But you can configure many more options at this command prompt.

Here is an example:

```
class-map match-any TEST
  match access-group 101
```

```
class-map match-all TEST2
  match ip precedence 6
```

3. Configure a policy map to apply a policy to a class that you previously defined.

The policy map contains:

- ◆ A name
- ◆ A set of class statements
- ◆ For each class statement, the action that needs to be taken for that class

The supported actions in PFC1 or PFC2 QoS are:

- ◆ **trust dscp**
- ◆ **trust ip precedence**
- ◆ **trust cos**
- ◆ **police**

You must use the **police** statement because the **set ip dscp** and **set ip precedence** actions are not supported. Since you do not actually want to police the traffic, but just to mark it, use a policer that is defined to allow all traffic. Therefore, configure the policer with a large rate and burst. For example, you can configure the policer with the maximum allowed rate and burst. Here is an example:

```
policy-map test_policy
```

```

class TEST
  trust ip precedence
class TEST2
  police 4000000000 31250000 conform-action
  set-dscp-transmit 16 exceed-action policed-dscp-transmit

```

4. Configure a service policy input to apply a policy map that you previously defined to one or more interfaces.

**Note:** The service policy can be attached to either a physical interface or to the SVI or VLAN interface. If a service policy is attached to a VLAN interface, only ports that belong to that VLAN and that are configured for VLAN-based QoS use this service policy. If the port is not set for VLAN-based QoS, the port still uses the default port-based QoS and only looks at a service policy that is attached to the physical interface.

This example applies the service policy `test_policy` to the port Gigabit Ethernet 1/1:

```

(config) interface gigabitethernet 1/1
(config-if)#service-policy input test_policy

```

This example applies the service policy `test_policy` to all ports in VLAN 10 that have a VLAN-based configuration from the QoS point of view:

```

(config) interface gigabitethernet 1/2
(config-if)#switchport mode access
(config-if)#switchport access vlan 10
(config-if)#mls qos vlan-based
(config-if)#exit
(config-if)#interface vlan 10
(config-if)#service-policy input test_policy

```

## Four Possible Sources for Internal DSCP

The internal DSCP is derived from one of these:

1. An existing received DSCP value, which is set before the frame enters the switch

An example is **trust dscp**.

2. The received IP precedence bits that are already set in the IPv4 header

Because there are 64 DSCP values and only eight IP precedence values, the administrator configures a mapping that the switch uses to derive the DSCP. Default mappings are in place, in the case that the administrator does not configure the maps. An example is **trust ip precedence**.

3. The received CoS bits that are already set before the frame enters the switch and which are stored in the data bus header, or if there was no CoS in the incoming frame, from the default CoS of the incoming port

As with IP precedence, there are a maximum of eight CoS values, each of which must be mapped to one of the 64 DSCP values. The administrator can configure this map, or the switch can use the default map that is already in place.

4. The service policy can set the internal DSCP to a specific value.

For numbers 2 and 3 in this list, the static mapping is by default, in this manner:

- For CoS-to-DSCP mapping, the DSCP that is derived equals eight times the CoS.
- For IP precedence-to-DSCP mapping, the DSCP that is derived equals eight times the IP precedence.



You can issue these commands in order to override and verify this static mapping:

- **mls qos map ip-prec-dscp** *dscp\_1 dscp\_2 dscp\_3 dscp\_4 dscp\_5 dscp\_6 dscp\_7 dscp\_8*
- **mls qos map cos-dscp** *dscp\_1 dscp\_2 dscp\_3 dscp\_4 dscp\_5 dscp\_6 dscp\_7 dscp\_8*

The first value of the DSCP that corresponds to the mapping for the CoS (or IP precedence) is 0. The second value for the CoS (or IP precedence) is 1, and the pattern continues in this way. For example, this command changes the mapping so that the CoS 0 is mapped to the DSCP of 0, and the CoS of 1 is mapped to the DSCP of 8, and so on:

```
Cat65(config)#mls qos map cos-dscp 0 8 16 26 32 46 48 54
Cat65#show mls qos maps
CoS-dscp map:
cos:      0 1  2   3   4   5   6   7
-----
dscp:     0 8 16 26 32 46 48 54
```

## How Is the Internal DSCP Chosen?

The internal DSCP is chosen on the basis of these parameters:

- The QoS policy map that is applied to the packet

The QoS policy map is determined by these rules:

- ◆ If no service policy is attached to the incoming port or VLAN, use the default.

**Note:** This default action is to set the internal DSCP to 0.

- ◆ If a service policy is attached to the incoming port or VLAN, and if the traffic matches one of the classes that the policy defines, use this entry.
- ◆ If a service policy is attached to the incoming port or VLAN, and if the traffic does not match one of the classes that the policy defines, use the default.
- The `trust` state of the port and the action of the policy map

When the port has a specific `trust` state and a policy with a certain marking (trusting action at the same time), these rules apply:

- ◆ The **set ip dscp** command or the DSCP that is defined per policer in a policy map is only applied if the port is left in the `untrusted` state.

If the port has a `trust` state, this `trust` state is used to derive the internal DSCP. The port `trust` state always takes precedence over the **set ip dscp** command.

- ◆ The **trust** *xx* command in a policy map takes precedence over the port `trust` state.

If the port and the policy contain a different `trust` state, the `trust` state that comes from the policy map is considered.

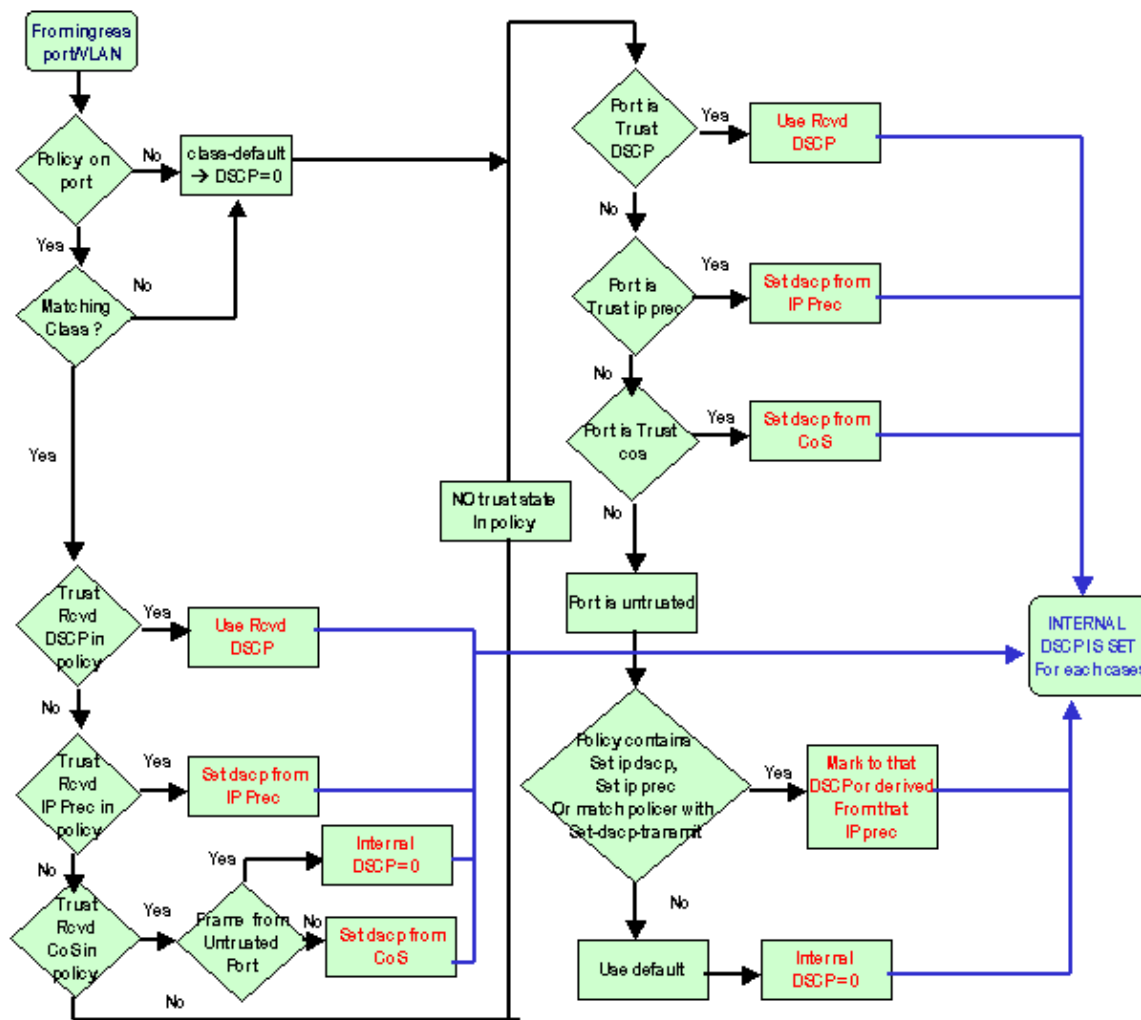
Therefore, the internal DSCP depends on these factors:

- The port `trust` state
- The service policy (with use of ACL) that is attached to the port
- The default policy map

**Note:** The default resets the DSCP to 0.

- Whether VLAN-based or port-based with regard to the ACL

This diagram summarizes how the internal DSCP is chosen on the basis of the configuration:



The PFC is also able to do policing. This can eventually result in a markdown of the internal DSCP. For more information on policing, refer to QoS Policing on Catalyst 6500/6000 Series Switches.

## Output Port Handling

You cannot do anything at the egress port level in order to change the classification. However, mark the packet on the basis of these rules:

- If the packet is an IPv4 packet, copy the internal DSCP that the switching engine assigns into the ToS byte of the IPv4 header.
- If the output port is configured for an ISL or dot1q encapsulation, use a CoS that is derived from the internal DSCP. Copy the CoS in the ISL or dot1q frame.

**Note:** The CoS is derived from the internal DSCP according to a static. Issue this command in order to configure the static:

```
Router(config)#mls qos map dscp-cos dscp1 [dscp2 [dscp3 [dscp4
[dscp5 [dscp6 [dscp7 [dscp8]]]]]] to cos_value
```

*!--- Note: This command should be on one line.*

The default configurations appear here. By default, the CoS is the integer part of the DSCP, divided by eight. Issue this command in order to see and verify the mapping:

```

cat6k#show mls qos maps
...
Dscp-cos map:                                     (dscp= d1d2)
  d1 :  d2 0  1  2  3  4  5  6  7  8  9
-----
  0 :    00 00 00 00 00 00 00 00 01 01
  1 :    01 01 01 01 01 01 02 02 02 02
  2 :    02 02 02 02 03 03 03 03 03 03
  3 :    03 03 04 04 04 04 04 04 04 04
  4 :    05 05 05 05 05 05 05 05 06 06
  5 :    06 06 06 06 06 06 07 07 07 07
  6 :    07 07 07 07

```

In order to change this mapping, issue this configuration command in the normal configuration mode:

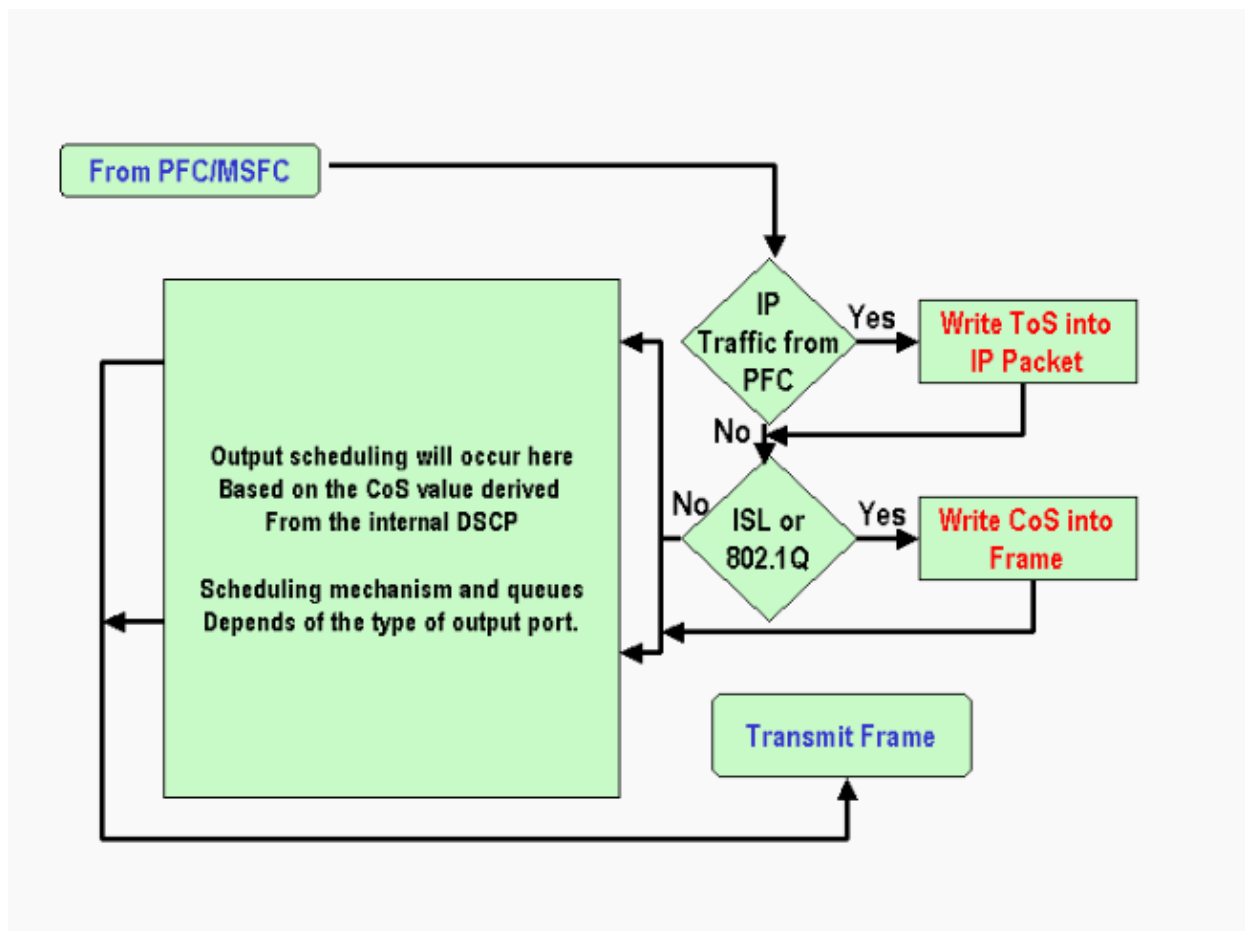
```

mls qos map dscp-cos 0 1 2 3 4 5 6 7 to 0
mls qos map dscp-cos 8 9 10 11 12 13 14 15 to 1
mls qos map dscp-cos 16 17 18 19 20 21 22 23 to 2
...

```

After the DSCP is written into the IP header and the CoS is derived from the DSCP, the packet is sent to one of the output queues for output scheduling on the basis of the CoS. This occurs even if the packet is not a dot1q or an ISL. For more information on output queue scheduling, refer to QoS Output Scheduling on Catalyst 6500/6000 Series Switches Running Cisco IOS System Software.

This diagram summarizes the processing of the packet with regard to marking in the output port:



# Notes and Limitations

## The Default ACL

The default ACL uses "dscp 0" as the classification keyword. All traffic that enters the switch through an untrusted port and does not hit a service policy entry is marked with a DSCP of 0 if QoS is enabled. Currently, you cannot change the default ACL in Cisco IOS Software.

**Note:** In Catalyst OS (CatOS) software, you can configure and change this default behavior. For more information, refer to the *The Default ACL* section of QoS Classification and Marking on Catalyst 6500/6000 Series Switches Running CatOS Software.

## Limitations of the WS-X61xx, WS-X6248-xx, WS-X6224-xx, and WS-X6348-xx Line Cards

This section only concerns these line cards:

- WS-X6224-100FX-MT: Catalyst 6000 24-Port 100 FX Multimode
- WS-X6248-RJ-45: Catalyst 6000 48-Port 10/100 RJ-45 Module
- WS-X6248-TEL: Catalyst 6000 48-Port 10/100 Telco Module
- WS-X6248A-RJ-45: Catalyst 6000 48-Port 10/100, Enhanced QoS
- WS-X6248A-TEL: Catalyst 6000 48-Port 10/100, Enhanced QoS
- WS-X6324-100FX-MM: Catalyst 6000 24-Port 100 FX, Enhanced QoS, MT
- WS-X6324-100FX-SM: Catalyst 6000 24-Port 100 FX, Enhanced QoS, MT
- WS-X6348-RJ-45: Catalyst 6000 48-Port 10/100, Enhanced QoS
- WS-X6348-RJ21V: Catalyst 6000 48-Port 10/100, Inline Power
- WS-X6348-RJ45V: Catalyst 6000 48-Port 10/100, Enhanced QoS, Inline Power
- WS-X6148-RJ21V: Catalyst 6500 48-Port 10/100 Inline Power
- WS-X6148-RJ45V: Catalyst 6500 48-Port 10/100 Inline Power

These line cards have a limitation. At the port level, you cannot configure the `trust` state with the use of any of these keywords:

- `trust-dscp`
- `trust-ipprec`
- `trust-cos`

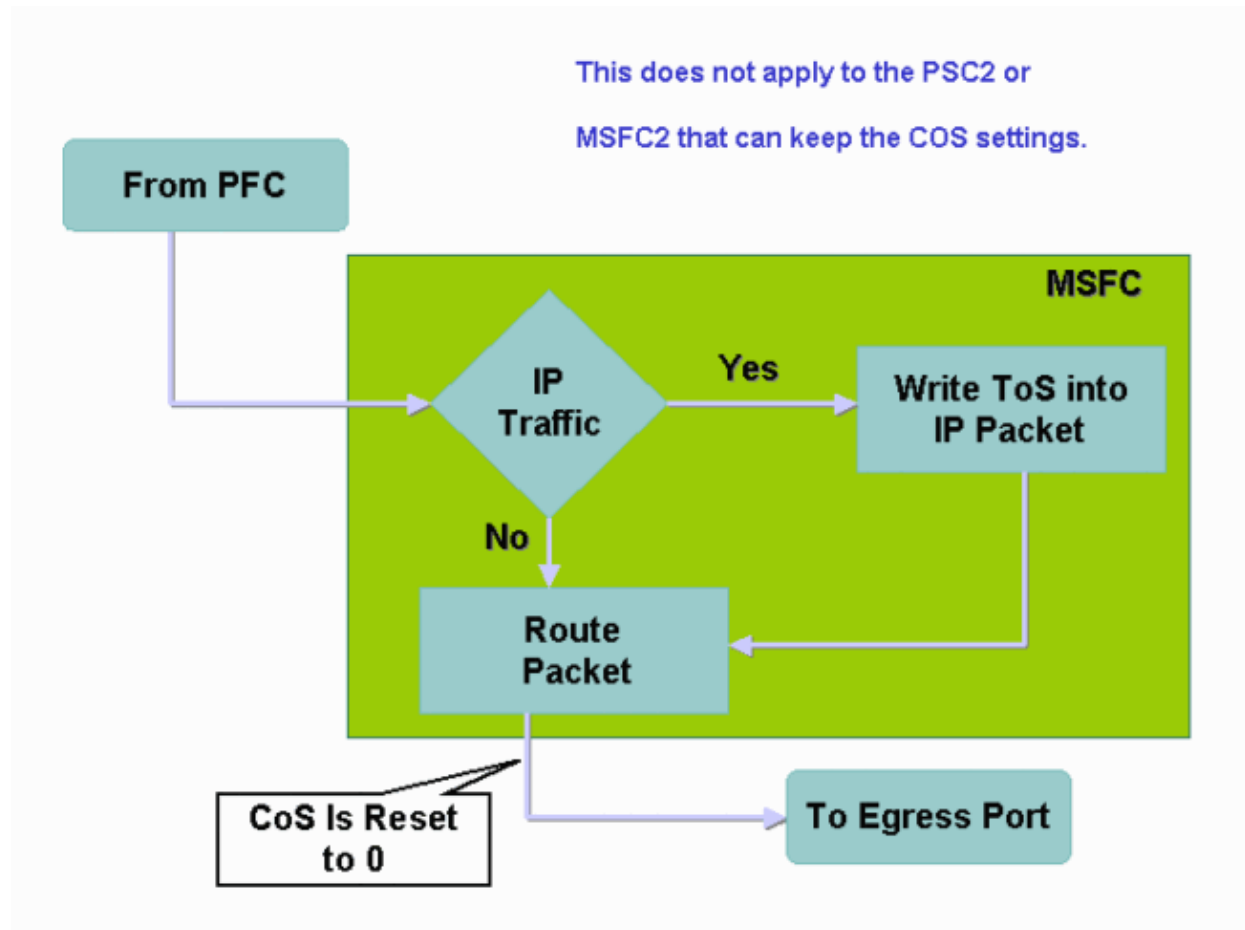
You can only use the `untrusted` state. Any attempt to configure a `trust` state on one of these ports displays one of these warning messages:

```
Tank(config-if)#mls qos trust ?
  extend  extend keyword
Tank(config-if)#mls qos trust
% Incomplete command.
Tank(config-if)#mls qos trust cos
                        ^
% Invalid input detected at '^' marker.
Tank(config-if)#mls qos trust ip-pre
                        ^
% Invalid input detected at '^' marker.
```

You must attach a service policy to the port or the VLAN if you want a trusting frame to come in on such a line card. Use the method in the Case 1: Marking at the Edge section of this document.

## Packets That Come from the MSFC1 or MSFC2 on Supervisor Engine 1A/PFC

All packets that come from the MSFC1 or MSFC2 have a CoS of 0. The packet can be either a software-routed packet or a packet that the MSFC issues. This is a limitation of the PFC because it resets the CoS of all packets that come from the MSFC. The DSCP and IP precedence are still maintained. The PFC2 does not have this limitation. The exiting CoS of the PFC2 is equal to the IP precedence of the packet.



## Summary of Classification

The tables in this section show the DSCP that results on the basis of these classifications:

- The incoming port trust state
- The classification keyword within the applied ACL

This table provides is a generic summary for all ports except WS-X62xx and WS-X63xx:

Policy Map Keyword				
Port Trust State	set-ip-dscp xx or set-dscp-transmit	trust-dscp	trust-ipprec	trust-cos
untrusted	xx <sup>1</sup>	Rx <sup>2</sup> DSCP	Derived from Rx ipprec	0
trust-dscp	Rx DSCP	Rx DSCP		

			Derived from Rx ipprec	Derived from Rx CoS or port CoS
<b>trust-ipprec</b>	Derived from Rx ipprec	Rx DSCP	Derived from Rx ipprec	Derived from Rx CoS or port CoS
<b>trust-cos</b>	Derived from Rx CoS or port CoS	Rx DSCP	Derived from Rx ipprec	Derived from Rx CoS or port CoS

<sup>1</sup> This is the only way to make a new marking of a frame.

<sup>2</sup> Rx = receive

This table provides a summary for the WS-X61xx, WS-X62xx, and WS-X63xx ports:

Policy Map Keyword				
Port Trust State	<b>set-ip-dscp xx or set-dscp-transmit</b>	<b>trust-dscp</b>	<b>trust-ipprec</b>	<b>trust-cos</b>
<b>untrusted</b>	xx xx	Rx DSCP	Derived from Rx ipprec	0
<b>trust-dscp</b>	Not supported	Not supported	Not supported	Not supported
<b>trust-ipprec</b>	Not supported	Not supported	Not supported	Not supported
<b>trust-cos</b>	Not supported	Not supported	Not supported	Not supported

## Monitor and Verify a Configuration

### Check the Port Configuration

Issue the **show queuing interface *interface-id*** command in order to verify the port settings and configurations.

When you issue this command, you can verify these classification parameters, among other parameters:

- Whether port-based or VLAN-based
- The `trust` port type
- The ACL that is attached to the port

Here is a sample of this command output. The important fields with regard to classification appear in boldface:

```
6500#show queuing interface gigabitethernet 3/2
```

```

Interface GigabitEthernet3/2 queuing strategy: Weighted Round-Robin
  Port QoS is enabled
  Trust state: trust COS
  Default COS is 0
  Transmit queues [type = 1p2q2t]:

```

The output shows that the configuration of this specific port is with `trust cos` on the port level. Also, the default port CoS is 0.

## Check Defined Classes

Issue the `show class-map` command in order to check the defined classes. Here is an example:

```

Boris#show class-map
Class Map match-all test (id 3)
  Match access-group 112

Class Map match-any class-default (id 0)
  Match any
Class Map match-all voice (id 4)

```

## Check the Policy Map That Is Applied to an Interface

Issue these commands in order to check the policy map that is applied and seen in previous commands:

- `show mls qos ip interface interface-id`
- `show policy-map interface interface-id`

Here are samples of the output from the issue of these commands:

```

Boris#show mls qos ip gigabitethernet 1/1
  [In] Default.  [Out] Default.
QoS Summary [IP]:          (* - shared aggregates, Mod - switch module)

Int  Mod Dir  Class-map  DSCP  AgId  Trust  FlId  AgForward-Pk  AgPoliced-k
-----
Gil/1 1  In   TEST       0     0*   No    0           1242120099           0

```

**Note:** You can look at these fields that relate to classification:

- **Class-map** Tells you which class is attached to the service policy that is attached to this interface.
- **Trust** Tells you whether the police action in that class contains a **trust** command and what is trusted in the class.
- **DSCP** Tells you the DSCP that is transmitted for the packets that hit that class.

```

Tank#show policy-map interface fastethernet 4/4

FastEthernet4/4

service-policy input: TEST_aggre2

class-map: Test_marking (match-all)
  27315332 packets
  5 minute offered rate 25726 pps
  match: access-group 101
  police :
    10000000 bps 10000 limit 10000 extended limit
    aggregate-forwarded 20155529 packets action: transmit
    exceeded 7159803 packets action: drop
    aggregate-forward 19498 pps exceed 6926 pps

```

# Sample Case Studies

This section provides sample configurations of common cases that can appear in a network.

## Case 1: Marking at the Edge

Assume that you configure a Catalyst 6000 that is used as an access switch. Many users connect to the switch slot 2, which is a WS-X6348 line card (10/100 Mbps). The users can send:

- Normal data traffic This traffic is always in VLAN 100 and needs to get a DSCP of 0.
- Voice traffic from an IP phone This traffic is always in the voice auxiliary VLAN 101 and needs to get a DSCP of 46.
- Mission-critical application traffic This traffic also comes in VLAN 100 and is directed to server 10.10.10.20. This traffic needs to get a DSCP of 32.

The application does not mark any of this traffic. Therefore, leave the port as `untrusted` and configure a specific ACL to classify the traffic. One ACL is applied to VLAN 100, and one ACL is applied to VLAN 101. You also need to configure all ports as VLAN-based. Here is an example of the configuration that results:

```
Boris(config)#mls qos
Boris(config)#interface range fastethernet 2/1-48
Boris(config-if)#mls qos vlan-based
Boris(config-if)#exit
Boris(config)#ip access-list extended Mission_critical
Boris(config-ext-nacl)#permit ip any host 10.10.10.20
Boris(config)#ip access-list extended Voice_traffic
Boris(config-ext-nacl)#permit ip any any
Boris(config)#class-map voice

Boris(config-cmap)#match access-group Voice_traffic
Boris(config)#class-map Critical

Boris(config-cmap)#match access-group Mission_critical
Boris(config)#policy-map Voice_vlan
Boris(config-pmap)#class voice
Boris(config-pmap-c)#set ip dscp 46
Boris(config)#policy-map Data_vlan
Boris(config-pmap)#class Critical
Boris(config-pmap-c)#set ip dscp 32
Boris(config)#interface vlan 100
Boris(config-if)#service-policy input Data_vlan
Boris(config)#interface vlan 101
Boris(config-if)#service-policy input Voice_vlan
```

## Case 2: Trusting in the Core with Only Gigabit Ethernet Interfaces

Assume that you configure a core Catalyst 6000 with only a Gigabit Ethernet interface in slot 1 and slot 2. The access switches previously marked traffic correctly. Therefore, you do not need to make any remarking. However, you need to ensure that the core switch does trust the incoming DSCP. This case is the easier case because all ports are marked as `trust-dscp`, which should be sufficient:

```
6k(config)#mls qos
6k(config)#interface range gigabitethernet 1/1-2 , gigabitethernet 2/1-2
6k(config-if)#mls qos trust dscp
```



## Related Information

- **Understanding Quality of Service on Catalyst 6000 Family Switches**
  - **QoS Classification and Marking on Catalyst 6500/6000 Series Switches Running CatOS Software**
  - **LAN Product Support**
  - **LAN Switching Technology Support**
  - **Technical Support & Documentation – Cisco Systems**
- 

[Contacts & Feedback](#) | [Help](#) | [Site Map](#)

© 2013 – 2014 Cisco Systems, Inc. All rights reserved. [Terms & Conditions](#) | [Privacy Statement](#) | [Cookie Policy](#) | [Trademarks of Cisco Systems, Inc.](#)

---

Updated: Dec 08, 2005

Document ID: 24055

---