

Class-Based Weighted Fair Queueing

Feature Overview

Class-based weighted fair queueing (CBWFQ) extends the standard WFQ functionality to provide support for user-defined traffic classes. For CBWFQ, you define traffic classes based on match criteria including protocols, access control lists (ACLs), and input interfaces. Packets satisfying the match criteria for a class constitute the traffic for that class. A queue is reserved for each class, and traffic belonging to a class is directed to the queue for that class.

Once a class has been defined according to its match criteria, you can assign it characteristics. To characterize a class, you assign it bandwidth, weight, and maximum packet limit. The bandwidth assigned to a class is the guaranteed bandwidth delivered to the class during congestion.

To characterize a class, you also specify the queue limit for that class, which is the maximum number of packets allowed to accumulate in the queue for the class. Packets belonging to a class are subject to the bandwidth and queue limits that characterize the class.

After a queue has reached its configured queue limit, enqueueing of additional packets to the class causes tail drop or packet drop to take effect, depending on how class policy is configured.

Tail drop is used for CBWFQ classes unless you explicitly configure policy for a class to use Weighted Random Early Detection (WRED) to drop packets as a means of avoiding congestion. Note that if you use WRED packet drop instead of tail drop for one or more classes comprising a policy map, you must ensure that WRED is not configured for the interface to which you attach that service policy.

If a default class is configured with the **bandwidth** policy-map class configuration command, all unclassified traffic is put into a single queue and given treatment according to the configured bandwidth. If a default class is configured with the **fair-queue** command, all unclassified traffic is flow classified and given best-effort treatment. If no default class is configured, then by default the traffic that does not match any of the configured classes is flow classified and given best-effort treatment. Once a packet is classified, all of the standard mechanisms that can be used to differentiate service among the classes apply.

Flow classification is standard WFQ treatment. That is, packets with the same source IP address, destination IP address, source Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) port, or destination TCP or UDP port are classified as belonging to the same flow. WFQ allocates an equal share of bandwidth to each flow. Flow-based WFQ is also called fair queueing because all flows are equally weighted.

For CBWFQ, which extends the standard WFQ fair queuing, the weight specified for the class becomes the weight of each packet that meets the match criteria of the class. Packets that arrive at the output interface are classified according to the match criteria filters you define, then each one is assigned the appropriate weight. The weight for a packet belonging to a specific class is derived from the bandwidth you assigned to the class when you configured it; in this sense the weight for a class is user-configurable.

After the weight for a packet is assigned, the packet is enqueued in the appropriate class queue. CBWFQ uses the weights assigned to the queued packets to ensure that the class queue is serviced fairly.

Configuring a class policy—thus, configuring CBWFQ—entails these three processes:

- Defining traffic classes to specify the classification policy (class maps).
This process determines how many types of packets are to be differentiated from one another.
- Associating policies—that is, class characteristics—with each traffic class (policy maps).
This process entails configuration of policies to be applied to packets belonging to one of the classes previously defined through a class map. For this process, you configure a policy map that specifies the policy for each traffic class.
- Attaching policies to interfaces (service policies).
This process requires that you associate an existing policy map, or service policy, with an interface to apply the particular set of policies for the map to that interface.

Benefits

Bandwidth Allocation

CBWFQ allows you to specify the exact amount of bandwidth to be allocated for a specific class of traffic. Taking into account available bandwidth on the interface, you can configure up to 64 classes and control distribution among them, which is not the case with flow-based WFQ.

Flow-based WFQ applies weights to traffic to classify it into conversations and determine how much bandwidth each conversation is allowed relative to other conversations. For flow-based WFQ, these weights, and traffic classification, are dependent on and limited to the seven IP Precedence levels.

Coarser Granularity and Scalability

CBWFQ allows you to define what constitutes a class based on criteria that exceed the confines of flow. CBWFQ allows you to use access control lists and protocols or input interface names to define how traffic will be classified, thereby providing coarser granularity. You need not maintain traffic classification on a flow basis. Moreover, you can configure up to 64 discrete classes in a service policy.

Restrictions

- Configuring CBWFQ on a physical interface is only possible if the interface is in the default queueing mode. Serial interfaces at E1 (2.048 Mbps) and below use WFQ by default—other interfaces use FIFO by default. Enabling CBWFQ on a physical interface overrides the default interface queueing method. Enabling CBWFQ on an ATM PVC does not override the default queueing method.
- If you configure a class in a policy map to use WRED for packet drop instead of tail drop, you must ensure that WRED is not configured on the interface to which you intend to attach that service policy.

- Traffic shaping and policing are not currently supported with CBWFQ.
- CBWFQ is supported on variable bit rate (VBR) and available bit rate (ABR) ATM connections. It is not supported on unspecified bit rate (UBR) connections.
- CBWFQ is not supported on subinterfaces.

Related Features and Technologies

Resource Reservation Protocol (RSVP) can be used in conjunction with CBWFQ. When both RSVP and CBWFQ are configured for an interface, RSVP and CBWFQ act independently, exhibiting the same behavior that they would if each were running alone. RSVP continues to work as it does when CBWFQ is not present, even in regard to bandwidth availability assessment and allocation.

Related Documents

For related information on this feature, refer to the following documents:

- Cisco IOS Release 12.0 *Quality of Service Solutions Configuration Guide*
- Cisco IOS Release 12.0 *Quality of Service Solutions Command Reference*

CBWFQ supports standard and extended numbered access lists only. For information on creating access lists, see the appropriate Cisco IOS Release 12.0 configuration guides and command references.

Supported Platforms

- Cisco 1600 series
- Cisco 1700 series
- Cisco 2500 series
- Cisco 2600 series
- Cisco 3600 series
- Cisco 3800 series
- Cisco 4500 series
- Cisco 7100 router
- Cisco 7200 series
- Cisco 7500 series
- Cisco AS5300 router

Prerequisites

Weighted Fair Queueing

Attaching a service policy to an interface disables WFQ on that interface if WFQ is configured for the interface. For this reason, you should ensure that WFQ is not enabled on such an interface.

For information on WFQ, see the “Configuring Weighted Fair Queueing” chapter of the Cisco IOS Release 12.0 *Quality of Service Solutions Configuration Guide*.

Weighted Random Early Detection

Attaching a service policy containing classes configured to use WRED to an interface disables WRED on that interface. If any of the classes that you configure in a policy map use WRED for packet drop instead of tail drop, you must ensure that WRED is not configured on the interface to which you intend to attach that service policy.

Access Control Lists

You can specify a numbered access list as the match criterion for any class that you create. For this reason, you should know how to configure access lists.

Supported MIBs and RFCs

MIBs

None

RFCs

None

Configuration Tasks

Perform the following tasks to configure CBWFQ:

- Defining Class Maps
- Configuring Class Policy in the Policy Map
- Attaching the Service Policy and Enabling CBWFQ
- Modifying the Bandwidth for an Existing Policy Map Class
- Modifying the Queue Limit for an Existing Policy Map Class
- Configuring the Bandwidth Limiting Factor
- Deleting Classes
- Deleting Policy Maps
- Verifying Configuration of Policy Maps and Their Classes

Defining Class Maps

To create a class map containing match criteria against which a packet is checked to determine if it belongs to a class—and to effectively create the class whose policy can be specified in one or more policy maps—use the first command in global configuration mode to specify the class map name, then one of the following commands in class map configuration mode:

Step	Command	Purpose
1	Router(config)# class-map <i>class-map-name</i>	Specifies the name of the class map to be created.
2	Router(config-cmap)# match access-group { <i>access-group</i> name <i>access-group-name</i> }	Specifies the name of the ACL against whose contents packets are checked to determine if they belong to the class. CBWFQ supports numbered and named ACLs.
	or	
	Router(config-cmap)# match input-interface <i>interface-name</i>	Specifies the name of the input interface used as a match criterion against which packets are checked to determine if they belong to the class.
	or	
	Router(config-cmap)# match protocol <i>protocol</i>	Specifies the name of the protocol used as a match criterion against which packets are checked to determine if they belong to the class.
	or	
	Router(config-cmap)# match mpls experimental <i>number</i>	Specifies the value of the EXP field to be used as a match criterion against which packets are checked to determine if they belong to the class.

Configuring Class Policy in the Policy Map

To configure a policy map and create class policies that make up the service policy, use the **policy-map** command to specify the policy map name, then use one or more of the following commands to configure policy for a standard class or the default class:

- **class**
- **bandwidth**
- **fair-queue** (for class-default class only)
- **queue-limit or random-detect**

For each class that you define, you can use one or more of the listed commands to configure class policy. For example, you might specify bandwidth for one class and both bandwidth and queue limit for another class.

The default class of the policy map (commonly known as the class-default class) is the class to which traffic is directed if that traffic does not satisfy the match criteria of other classes whose policy is defined in the policy map.

You can configure class policies for as many classes as are defined on the router, up to the maximum of 64. However, the total amount of bandwidth allocated for all classes included in a policy map must not exceed 75 percent of the available bandwidth on the interface. The other 25 percent is used for control and routing traffic. (To override the 75 percent limitation, use the **max-reserved bandwidth** command.) If not all of the bandwidth is allocated, the remaining bandwidth is proportionally allocated among the classes, based on their configured bandwidth.

To configure class policies in a policy map, perform the tasks in the following sections:

- Configuring Class Policy Using Tail Drop
- Configuring Class Policy Using WRED Packet Drop
- Configuring the Class-Default Class Policy

Configuring Class Policy Using Tail Drop

To configure a policy map and create class policies that make up the service policy, use the first command in global configuration mode to specify the policy map name, then use the following commands in policy-map class configuration mode to configure policy for a standard class. To configure policy for the default class, see the section “Configuring the Class-Default Class Policy.”

Step	Command	Purpose
1	Router(config)# policy-map <i>policy-map</i>	Specifies the name of the policy map to be created or modified.
2	Router(config-pmap)# class <i>class-name</i>	Specifies the name of a class to be created and included in the service policy.
3	Router(config-pmap-c)# bandwidth <i>bandwidth-kbps</i>	Specifies the amount of bandwidth in kbps to be assigned to the class. The amount of bandwidth configured should be large enough to also accommodate Layer 2 overhead.
4	Router(config-pmap-c)# queue-limit <i>number-of-packets</i>	Specifies the maximum number of packets that can be enqueued for the class.

To configure policy for more than one class in the same policy map, repeat Step 2 through Step 4. Note that because this set of commands uses the **queue-limit** command, the policy map uses tail drop, not Weighted Random Early Detection (WRED) packet drop.

Configuring Class Policy Using WRED Packet Drop

To configure a policy map and create class policies comprising the service policy, use the first global configuration command to specify the policy map name, then use the following policy-map class configuration commands to configure policy for a standard class. To configure policy for the default class, see the section “Configuring the Class-Default Class Policy.”

Step	Command	Purpose
1	Router(config)# policy-map <i>policy-map</i>	Specifies the name of the policy map to be created or modified.
2	Router(config-pmap)# class <i>class-name</i>	Specifies the name of a class to be created and included in the service policy.
3	Router(config-pmap-c)# bandwidth <i>bandwidth-kbps</i>	Specifies the amount of bandwidth in kbps to be assigned to the class. The amount of bandwidth configured should be large enough to also accommodate Layer 2 overhead.
4	Router(config-pmap-c)# random-detect	Enables WRED. The class policy will drop packets using WRED instead of tail drop.

Step	Command	Purpose
5	<pre>Router(config-pmap-c)# random-detect exponential-weighting-constant exponent and/or Router(config-pmap-c)# random-detect precedence precedence min-threshold max-threshold mark-prob-denominator</pre>	<p>Configures the exponential weight factor used in calculating the average queue length.</p> <p>Configures WRED parameters for packets with a specific IP Precedence. Repeat this command for each precedence.</p>

To configure policy for more than one class in the same policy map, repeat Step 2 through Step 5. Note that this set of commands uses WRED packet drop, not tail drop.

Note If you configure a class in a policy map to use WRED for packet drop instead of tail drop, you must ensure that WRED is not configured on the interface to which you intend to attach that service policy.

Configuring the Class-Default Class Policy

The class-default class is used to classify traffic that does not fall into one of the defined classes. Once a packet is classified, all of the standard mechanisms that can be used to differentiate service among the classes apply. The class-default class was predefined when you created the policy map, but you must configure it. If no default class is configured, then by default the traffic that does not match any of the configured classes is flow classified and given best-effort treatment.

By default, the class-default class is defined as flow-based WFQ. However, configuring the default class with the **bandwidth** policy-map class configuration command disqualifies the default class as flow-based WFQ.

To configure a policy map and configure the class-default class to use tail drop, use the first global configuration command to specify the policy map name, then use the following policy-map class configuration commands to configure policy for the default class:

Step	Command	Purpose
1	<pre>Router(config)# policy-map policy-map</pre>	Specifies the name of the policy map to be created or modified.
2	<pre>Router(config-pmap)# class class-default default-class-name</pre>	Specifies the default class so that you can configure or modify its policy.
3	<pre>Router(config-pmap-c)# bandwidth bandwidth-kbps or Router(config-pmap-c)# fair-queue [number-of-dynamic-queues]</pre>	<p>Specifies the amount of bandwidth in kbps to be assigned to the class. The amount of bandwidth configured should be large enough to also accommodate Layer 2 overhead.</p> <p>Specifies the number of dynamic queues to be reserved for use by flow-based WFQ running on the default class. The number of dynamic queues is derived from the bandwidth of the interface. See the tables accompanying the description of the fair-queue (WFQ) command in the <i>Cisco IOS Quality of Service Solutions Command Reference</i> for the default number of dynamic queues that WFQ and CBWFQ use when they are enabled on an interface or ATM VC.</p>
4	<pre>Router(config-pmap-c)# queue-limit number-of-packets</pre>	Specifies the maximum number of packets that the queue for the default class can accumulate.

Attaching the Service Policy and Enabling CBWFQ

To configure a policy map and configure the class-default class to use WRED packet drop, use the first global configuration command to specify the policy map name, then use the following policy-map class configuration commands to configure policy for the default class:

Step	Command	Purpose
1	Router(config)# policy-map <i>policy-map</i>	Specifies the name of the policy map to be created or modified.
2	Router(config-pmap)# class class-default <i>default-class-name</i>	Specifies the default class so that you can configure or modify its policy.
3	Router(config-pmap-c)# bandwidth <i>bandwidth-kbps</i> or Router(config-pmap-c)# fair-queue <i>[number-of-dynamic-queues]</i>	Specifies the amount of bandwidth in kbps to be assigned to the class. The amount of bandwidth configured should be large enough to also accommodate Layer 2 overhead. Specifies the number of dynamic queues to be reserved for use by flow-based WFQ running on the default class. The number of dynamic queues is derived from the bandwidth of the interface. See the tables accompanying the description of the fair-queue (WFQ) command in the <i>Cisco IOS Quality of Service Solutions Command Reference</i> for the default number of dynamic queues that WFQ and CBWFQ use when they are enabled on an interface or ATM VC.
4	Router(config-pmap-c)# random-detect	Enables WRED. The class policy will drop packets using WRED instead of tail drop.
5	Router(config-pmap-c)# random-detect exponential-weighting-constant <i>exponent</i> and/or Router(config-pmap-c)# random-detect precedence <i>precedence min-threshold max-threshold</i> <i>mark-prob-denominator</i>	Configures the exponential weight factor used in calculating the average queue length. Configures WRED parameters for packets with a specific IP Precedence. Repeat this command for each precedence.

Attaching the Service Policy and Enabling CBWFQ

To attach a service policy to the output interface and enable CBWFQ on the interface, use the following interface configuration command. When CBWFQ is enabled, all classes configured as part of the service policy map are installed in the fair queueing system.

Command	Purpose
Router(config-if)# service-policy output <i>policy-map</i>	Enables CBWFQ and attaches the specified service policy map to the output interface.

Modifying the Bandwidth for an Existing Policy Map Class

To change the amount of bandwidth allocated for an existing class, use the following commands beginning in global configuration mode:

Step	Command	Purpose
1	Router(config)# policy-map <i>policy-map</i>	Specifies the name of the policy map containing the class to be modified.
2	Router(config-pmap)# class <i>class-name</i>	Specifies the name of a class whose bandwidth you want to modify.
3	Router(config-pmap-c)# bandwidth <i>bandwidth-kbps</i>	Specifies the new amount of bandwidth in kbps per second to be used to reconfigure the class.

Modifying the Queue Limit for an Existing Policy Map Class

To change the maximum number of packets that can accrue in a queue reserved for an existing class, use the following commands beginning in global configuration mode:

Step	Command	Purpose
1	Router(config)# policy-map <i>policy-map</i>	Specifies the name of the policy map containing the class to be modified.
2	Router(config-pmap)# class <i>class-name</i>	Specifies the name of a class whose queue limit you want to modify.
3	Router(config-pmap-c)# queue-limit <i>number-of-packets</i>	Specifies the new maximum number of packets that can be enqueued for the class to be reconfigured. The default and maximum number of packets is 64.

Configuring the Bandwidth Limiting Factor

To change the maximum reserved bandwidth allocated for CBWFQ, LLQ, and the IP RTP Priority feature, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# max-reserved-bandwidth <i>percent</i>	Changes the maximum configurable bandwidth for CBWFQ, LLQ, and IP RTP Priority. The default is 75 percent.

Deleting Classes

To delete one or more class maps from a service policy map, use the following commands beginning in global configuration mode:

Step	Command	Purpose
1	Router(config)# policy-map <i>policy-map</i>	Specifies the name of the policy map containing the classes to be deleted.
2	Router(config-pmap)# no class <i>class-name</i>	Specifies the name of the class(es) to be deleted.
3	Router(config-pmap-c)# no class <i>class-default</i>	Deletes the default class.

Deleting Policy Maps

To delete a policy map, use the following command in global configuration mode:

Command	Purpose
Router(config)# no policy-map <i>policy-map</i>	Specifies the name of the policy map to be deleted.

Verifying Configuration of Policy Maps and Their Classes

To display the contents of a specific policy map, a specific class from a specific policy map, or all policy maps configured on an interface, use one of the following global configuration commands:

Command	Purpose
Router# show policy <i>policy-map</i>	Displays the configuration of all classes comprising the specified policy map.
Router# show policy <i>policy-map</i> class <i>class-name</i>	Displays the configuration of the specified class of the specified policy map.
Router# show policy interface <i>interface-name</i>	Displays the configuration of all classes configured for all policy maps on the specified interface.
Router# show queue <i>interface-type</i> <i>interface-number</i>	Displays queueing configuration and statistics for a particular interface.

The counters displayed after issuing the **show policy-map interface** command are updated only if congestion is present on the interface.

Configuration Examples

This section provides the following configuration examples:

- Class Map Configuration Example
- Policy Creation Example
- Policy Attachment to Interfaces Example
- CBWFQ Using WRED Packet Drop Example
- Display Service Policy Map Content Examples

Class Map Configuration Example

In the following example, ACLs 101 and 102 are created. Next, two class maps are created and their match criteria are defined. For the first map class, called class1, the numbered ACL 101 is used as the match criterion. For the second map class, called class2, the numbered ACL 102 is used as the match criterion. Packets are checked against the contents of these ACLs to determine if they belong to the class.

```
Router(config)# access-list 101 permit udp host 10.10.10.10 host 10.10.10.20 range
16384 20000
Router(config)# access-list 102 permit udp host 10.10.10.10 host 10.10.10.20 range
53000 56000

Router(config)# class-map class1
Router(config-cmap)# match access-group 101
Router(config-cmap)# exit

Router(config-cmap)# class-map class2
Router(config-cmap)# match access-group 102
Router(config-cmap)# exit
```

Policy Creation Example

In the following example, a policy map called `policy1` is defined to contain policy specification for the two classes—`class1` and `class2`. The match criteria for these classes were defined in the previous section “Class Map Configuration Example.”

For `class1`, the policy specifies the bandwidth allocation request and the maximum number of packets that the queue for this class can accumulate. For `class2`, the policy specifies only the bandwidth allocation request, so the default queue limit of 64 packets is assumed.

```
Router(config)# policy-map policy1

Router(config-pmap)# class class1
Router(config-pmap-c)# bandwidth 3000
Router(config-pmap-c)# queue-limit 30
Router(config-pmap-c)# exit

Router(config-pmap)# class class2
Router(config-pmap-c)# bandwidth 2000
Router(config-pmap-c)# exit
```

Policy Attachment to Interfaces Example

The following example shows how to attach an existing policy map. After you define a policy map, you can attach it to one or more interfaces to specify the service policy for those interfaces. Although you can assign the same policy map to multiple interfaces, each interface can have only one policy map attached at the input and one policy map attached at the output.

The policy map in this example was defined in the previous section, “Policy Creation Example.”

```
Router(config)# interface e1/1
Router(config-if)# service output policy1
Router(config-if)# exit

Router(config)# interface fa1/0/0
Router(config-if)# service output policy1
Router(config-if)# exit
```

CBWFQ Using WRED Packet Drop Example

In the following example, the class map `class1` is created and defined to use the input interface `FastEthernet0/1` as a match criterion to determine if packets belong to the class. Next, the policy map `policy1` is defined to contain policy specification for `class1`, which is configured for WRED packet drop.

```
Router(config)# class-map class1
Router(config-cmap)# match input-interface FastEthernet0/1
!
Router(config)# policy-map policy1
Router(config-pmap)# class class1
Router(config-pmap-c)# bandwidth 1000
Router(config-pmap-c)# random-detect
!
Router(config)# interface serial10/0
Router(config-if)# service-policy output policy1
!
```

Display Service Policy Map Content Examples

The following examples show how to display the contents of service policy maps. There are four methods to display the contents:

- Display all classes that make up a specified service policy map
- Display all classes configured for all service policy maps
- Display a specified class of a service policy map
- Display all classes configured for all service policy maps on a specified interface

All Classes for a Specified Service Policy Map

The following example displays the contents of the po1 service policy map:

```
Router# show policy-map po1
Policy Map po1
  Weighted Fair Queueing
    Class class1
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class2
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class3
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class4
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class5
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class6
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class7
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class8
      Bandwidth 937 (kbps) Max thresh 64 (packets)
```

All Classes for All Service Policy Maps

The following example displays the contents of all policy maps on the router:

```
Router# show policy-map
Policy Map poH1
  Weighted Fair Queueing
    Class class1
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class2
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class3
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class4
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class5
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class6
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class7
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class8
      Bandwidth 937 (kbps) Max thresh 64 (packets)
```

```

Policy Map policy2
  Weighted Fair Queueing
    Class class1
      Bandwidth 300 (kbps) Max thresh 64 (packets)
    Class class2
      Bandwidth 300 (kbps) Max thresh 64 (packets)
    Class class3
      Bandwidth 300 (kbps) Max thresh 64 (packets)
    Class class4
      Bandwidth 300 (kbps) Max thresh 64 (packets)
    Class class5
      Bandwidth 300 (kbps) Max thresh 64 (packets)
    Class class6
      Bandwidth 300 (kbps) Max thresh 64 (packets)

```

Specified Class for a Service Policy Map

The following example displays configurations for the class called class7 that belongs to the policy map pol:

```

Router# show policy-map pol class class7
Class class7
  Bandwidth 937 (kbps) Max Thresh 64 (packets)

```

All Classes for All Service Policy Maps on a Specified Interface

The following example displays configurations for classes on the output interface e1/1:

```

Router# show policy-map interface e1/1
Ethernet1/1 output : pol
  Weighted Fair Queueing
    Class class1
      Output Queue: Conversation 264
      Bandwidth 937 (kbps) Max Threshold 64 (packets)
      (total/discards/tail drops) 11548/0/0
    Class class2
      Output Queue: Conversation 265
      Bandwidth 937 (kbps) Max Threshold 64 (packets)
      (total/discards/tail drops) 11546/0/0
    Class class3
      Output Queue: Conversation 266
      Bandwidth 937 (kbps) Max Threshold 64 (packets)
      (total/discards/tail drops) 11546/0/0
    Class class4
      Output Queue: Conversation 267
      Bandwidth 937 (kbps) Max Threshold 64 (packets)
      (total/discards/tail drops) 11702/0/0
    Class class5
      Output Queue: Conversation 268
      Bandwidth 937 (kbps) Max Threshold 64 (packets)
      (total/discards/tail drops) 11701/0/0
    Class class6
      Output Queue: Conversation 269
      Bandwidth 937 (kbps) Max Threshold 64 (packets)
      (total/discards/tail drops) 11702/0/0
    Class class7
      Output Queue: Conversation 270
      Bandwidth 937 (kbps) Max Threshold 64 (packets)
      (total/discards/tail drops) 11857/0/0

```

```
Class class8
  Output Queue: Conversation 271
  Bandwidth 937 (kbps) Max Threshold 64 (packets)
  (total/discards/tail drops) 11858/1/0
```

Command Reference

This section documents new commands that configure the CBWFQ feature. All other commands used with this feature are documented in the Cisco IOS Release 12.0 command references.

- **bandwidth (policy-map class)**
- **class (policy-map)**
- **class-map**
- **fair-queue (class-default)**
- **match access-group**
- **match input-interface**
- **match protocol**
- **policy-map**
- **queue-limit**
- **random-detect exponential-weighting-constant**
- **random-detect precedence**
- **service-policy**
- **show policy-map**
- **show policy-map class**
- **show policy-map interface**

bandwidth (policy-map class)

To specify or modify the bandwidth allocated for a class belonging to a policy map, use the **bandwidth** policy map configuration command. To remove the bandwidth specified for a class, use the **no** form of this command.

```
bandwidth bandwidth-kbps
no bandwidth bandwidth-kbps
```

Syntax Description

bandwidth-kbps Amount of bandwidth in kilobits per second to be assigned to the class.

Defaults

No default behavior.

Command Modes

Policy map configuration

Command History

Release	Modification
12.0(5)T	This command was first introduced.

Usage Guidelines

You use the **bandwidth** command when you configure a policy in a map for a class defined by the **class-map** command. The **bandwidth** command specifies the bandwidth for traffic in that class. Class-based weighted fair queueing (CBWFQ) derives the weight for packets belonging to the class from the bandwidth allocated to the class. CBWFQ then uses the weight to ensure that the queue for the class is serviced fairly.

Note that when the policy map containing class policy configurations is attached to the interface to stipulate the service policy for that interface, available bandwidth is assessed. If a policy map cannot be attached to a particular interface because of insufficient interface bandwidth, then the policy is removed from all interfaces to which it was successfully attached.

Examples

The following example modifies the bandwidth for a class called `acl22`. The default class belongs to a service policy map called `polmap6`. The previous bandwidth for this class was 3000.

```
policy-map polmap6
  class acl22
    bandwidth 2000
    queue-limit 30
```

Related Commands

Command	Description
class	Specifies the class policy whose bandwidth specification is to be configured or modified.
class class-default	Specifies the default class whose bandwidth is to be configured or modified.
policy-map	Specifies the policy map to which the class belongs whose bandwidth is to be configured or modified.
random-detect exponential-weighting-constant	Configures the exponential weight factor used in calculating the average queue length.
random-detect precedence	Configures the parameters for packets with a specific IP Precedence. The minimum threshold for IP Precedence 0 corresponds to half the maximum threshold for the interface. Repeat this command for each precedence.

class (policy-map)

To specify the name of the class whose policy you want to create or change or to specify the default class (commonly known as the class-default class) before you configure its policy, use the **class** policy-map configuration command. To remove a class from the policy map, use the **no class** form of this command.

```
class [class-name | class-default]
no class [class-name | class-default]
```

Syntax Description

<i>class-name</i>	The name of the class for which you want to configure or modify policy.
class-default	Specifies the default class so that you can configure or modify its policy.

Defaults

No default behavior.

Command Modes

Policy-map configuration

Command History

Release	Modification
12.0(5)T	This command was first introduced.

Usage Guidelines

Enter the **policy-map** command to identify the policy map and enter policy map configuration mode before you use the **class** command. After you specify a policy map, you can configure policy for new classes or modify policy for any existing classes in that policy map.

The class name that you specify in the policy map ties the characteristics for that class—that is, its policy—to the class map and its match criteria configured using the **class-map** command.

When you configure policy for a class and specify its bandwidth and attach the policy map to an interface, class-based weighted fair queueing (CBWFQ) determines if the bandwidth requirement of the class can be satisfied. If so, CBWFQ creates the necessary internal data structures to maintain state for the class and allocates a queue for it.

When a class is removed, available bandwidth for the interface is incremented by the amount previously allocated to the class.

The maximum number of classes you can configure for a router—and, therefore, within a policy map—is 64.

You can define a class policy to use either tail drop (by using the **queue-limit** command) or Weighted Random Early Detection (WRED) packet drop (by using the **random-detect** command). You cannot use the **queue-limit** and **random-detect** commands in the same class policy, but they can be used in two class policies in the same policy map.

You can configure the **bandwidth** command when either the **queue-limit** or the **random-detect** command is configured in a class policy. The **bandwidth** command specifies the amount of bandwidth allocated for the class.

For the default class, you can configure the **fair-queue** (class-default) command. The **fair-queue** command specifies the number of dynamic queues for the default class. The **fair-queue** command can be used in the same class policy as either the **queue-limit** or **random-detect** command. It cannot be used with the **bandwidth** command.

Examples

The following example configures policy for a class named `acl136` included in the policy map called `policy1`. Class `acl136` has these characteristics: a minimum of 2000 kilobits per second (kbps) of bandwidth are expected to be delivered to this class in the event of congestion, and the queue reserved for this class can enqueue 40 packets before tail drop is enacted. Note that when the policy map containing this class is attached to the interface to stipulate the service policy for that interface, available bandwidth is assessed, taking into account all class policies and RSVP, if configured.

```
policy-map policy1
class acl136
bandwidth 2000
queue-limit 40
```

The following example configures policy for a class named `int101` included in the policy map called `policy8`. Class `int101` has these characteristics: a minimum of 3000 kbps of bandwidth are expected to be delivered to this class in the event of congestion, and a weight factor of 10 is used to calculate the average queue size. For congestion avoidance, WRED packet drop is used, not tail drop. Note that when the policy map containing this class is attached to the interface to stipulate the service policy for that interface, available bandwidth is assessed.

```
policy-map policy8
class int101
bandwidth 3000
random-detect exponential-weighting-constant 10
```

The following example configures policy for the **class-default** default class included in the policy map called `policy1`. The **class-default** default class has these characteristics: 10 dynamic queues for traffic that does not meet the match criteria of other classes whose policy is defined by the policy map `policy1`, and a maximum of 20 packets per queue before tail drop is enacted to handle additional enqueued packets.

```
policy-map policy1
class class-default
fair-queue 10
queue-limit 20
```

The following example configures policy for the **class-default** default class included in the policy map called policy8. The **class-default** default class has these characteristics: 20 dynamic queues for traffic that doesn't meet the match criteria of other classes whose policy is defined by the policy map policy8 and a weight factor of 20 is used to calculate the average queue size. For congestion avoidance, WRED packet drop is used, not tail drop.

```
policy-map policy8
class class-default
fair-queue 20
  random-detect exponential-weighting-constant 14
```

Related Commands

Command	Description
bandwidth	Specifies or modifies the bandwidth allocated for a class whose policy is specified in the policy map.
policy-map	Specifies the policy map to which the class policy belongs.
queue-limit	Specifies or modifies the maximum number of packets that can accumulate in the queue reserved for the class before tail drop or (if WRED is configured as part of the class policy) packet drop is enacted.
random-detect exponential-weighting-constant	Configures the exponential weight factor used in calculating the average queue length.
random-detect precedence	Configures the parameters for packets with a specific IP Precedence.

class-map

To create a class map to be used for matching packets to the class whose name you specify, use the **class-map** global configuration command. To remove an existing class map from the router, use the **no** form of this command.

```
class-map class-map-name  
no class-map class-map-name
```

Syntax Description

<i>class-map-name</i>	Name of the class for the class map. The class name is used for both the class map and to configure policy for the class in the policy map.
-----------------------	---

Defaults

No default behavior.

Command Modes

Global configuration

Command History

Release	Modification
12.0(5)T	This command was first introduced.

Usage Guidelines

Use this command to specify the name of the class for which you want to create or modify class map match criteria. Use of the **class-map** command enables class-map configuration mode in which you can enter one of the match commands to configure the match criteria for this class. Packets arriving at the output interface are checked against the match criteria configured for a class map to determine if the packet belongs to that class.

You can use one of the following commands in a class map:

- **match access-group**
- **match input-interface**
- **match mpls experimental**
- **match protocol**

If you specify more than one command in a class map, only the last command entered applies. The last command overrides the previously entered commands.

Examples

The following example specifies access101 as the name of a class, and it defines a class map for this class. The **match access-group** command is entered following the **class-map** command to configure the numbered access control list (ACL) 101 whose contents are to be used as match criteria for the class.

```
class-map access101
  match access-group 101
```

Related Commands

Command	Description
class (policy-map)	Specifies a class belonging to a service policy map.
match access-group	Configures the match criteria for a class map based on the specified ACL number.
match input-interface	Configures the match criteria for a class map based on the specified input interface.
match protocol	Configures the match criteria for a class map based on the specified protocol.
policy-map	Specifies the policy map to which the class belongs.

fair-queue (class-default)

To specify the number of dynamic queues to be reserved for use by the class-default class as part of the default class policy, use the **fair-queue** policy map configuration command. To delete the configured number of hash queues from the **class-default** policy, use the **no** form of this command.

fair-queue *number-of-dynamic-queues*
no fair-queue *number-of-dynamic-queues*

Syntax Description

number-of-dynamic-queues (Optional) A power of 2 number in the range of 16 to 4096 specifying the number of dynamic queues.

Defaults

The number of dynamic queues is derived from the interface or ATM permanent virtual circuit (PVC) bandwidth. See Table 1 for the default number of dynamic queues that weighted fair queuing (WFQ) and class-based WFQ (CBWFQ) use when they are enabled on an interface. See Table 2 for the default number of dynamic queues used when WFQ or CBWFQ is enabled on an ATM PVC.

Table 1 Default Number of Dynamic Queues as a Function of Interface Bandwidth

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

Table 2 Default Number of Dynamic Queues as a Function of ATM PVC Bandwidth

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

Command Modes

Policy-map configuration

Command History

Release	Modification
12.0(5)T	This command was first introduced.

Usage Guidelines

This command can be used for the default class (commonly known as the class-default class) only. You can use it in conjunction with either the **queue-limit** command or the **random-detect** command.

The class-default class is the default class to which traffic is directed if that traffic does not satisfy the match criteria of other classes whose policy is defined in the policy map.

Examples

The following example configures policy for the default class included in the policy map called policy9. Packets that do not satisfy match criteria specified for other classes whose policies are configured in the same service policy are directed to the default class, for which 16 dynamic queues have been reserved. Because the **queue-limit** command is configured, tail drop is used for each dynamic queue when the maximum number of packets are enqueued and additional packets arrive.

```
policy-map policy9
  class class-default
    fair-queue 16
    queue-limit 20
```

The following example configures policy for the default class included in the policy map called policy8. The **fair-queue** command reserves 20 dynamic queues to be used for the default class. For congestion avoidance, Weighted Random Early Detection (WRED) packet drop is used, not tail drop.

```
policy-map policy8
  class class-default
    fair-queue 20
    random-detect
```

Related Commands

Command	Description
queue-limit	Specifies or modifies the maximum number of packets the queue can hold for a class policy configured in a policy map.
random-detect (interface)	Enables WRED or DWRED.

match access-group

To configure the match criteria for a class map based on the specified access control list (ACL) number or name, use the **match access-group** class-map configuration command. To remove ACL match criteria from a class map, use the **no** form of this command.

```
match access-group access-group / name access-group-name
no match access-group access-group
```

Syntax Description

<i>access-group</i>	A named ACL whose contents are used as the match criteria against which packets are checked to determine if they belong to this class.
name <i>access-group-name</i>	A named ACL whose contents are used as the match criteria against which packets are checked to determine if they belong to this class.

Defaults

No default behavior.

Command Modes

Class-map configuration

Command History

Release	Modification
12.0(5)T	This command was first introduced.

Usage Guidelines

For class-based weighted fair queueing (CBWFQ), you define traffic classes based on match criteria including ACLs, protocols, input interfaces, QoS labels, and EXP field values. Packets satisfying the match criteria for a class constitute the traffic for that class.

The **match access-group** command specifies a numbered ACL whose contents are used as the match criteria against which packets are checked to determine if they belong to the class specified by the class map.

To use the **match access-group** command, you must first enter the **class-map** command to specify the name of the class whose match criteria you want to establish. After you identify the class, you can use one of the following commands to configure the match criteria.

- **match access-group**
- **match input-interface**
- **match mpls experimental**
- **match protocol**

If you specify more than one command in a class map, only the last command entered applies. The last command overrides the previously entered commands.

Examples

The following example specifies a class map called `acl144` and configures the ACL numbered 144 to be used as the match criteria for this class.

```
class-map acl144
  match access-group 144
```

Related Commands

Command	Description
class-map	Creates a class map to be used for matching packets to the specified class.
match input-interface	Configures a class map to use the specified input interface as a match criterion.
match mpls experimental	Configures a class map to use the specified EXP field value as a match criterion.
match protocol	Configures the match criteria for a class map based on the specified protocol.

match input-interface

To configure a class map to use the specified input interface as match criterion, use the **match input-interface** class-map configuration command. To remove the input interface match criterion from a class map, use the **no** form of this command.

```
match input-interface interface-name
no match input-interface interface-name
```

Syntax Description

interface-name Name of the input interface to be used as match criteria.

Defaults

No default behavior.

Command Modes

Class-map configuration

Command History

Release	Modification
12.0(5)T	This command was first introduced.

Usage Guidelines

For class-based weighted fair queueing (CBWFQ), you define traffic classes based on match criteria including input interfaces, ACLs, protocols, QoS labels, and EXP field values. Packets satisfying the match criteria for a class constitute the traffic for that class.

The **match input-interface** command specifies the name of an input interface to be used as the match criterion against which packets are checked to determine if they belong to the class specified by the class map.

To use the **match input interface** command, you must first enter the **class-map** command to specify the name of the class whose match criteria you want to establish. After you identify the class, you can use the following commands to configure the match criteria.

- **match access-group**
- **match input-interface**
- **match mpls experimental**
- **match protocol**

If you specify more than one command in a class map, only the last command entered applies. The last command overrides the previously entered commands.

Examples

The following example specifies a class map called eth1 and configures the input interface named ethernet1 to be used as the match criterion for this class:

```
class-map eth1
  match input-interface ethernet1
```

Related Commands

Command	Description
class-map	Creates a class map to be used for matching packets to a specified class.
match access-group	Configures the match criteria for a class map based on the specified ACL.
match mpls experimental	Configures a class map to use the specified EXP field value as a match criterion.
match protocol	Configures the match criteria for a class map based on the specified protocol.

match protocol

To configure the match criteria for a class map based on the specified protocol, use the **match protocol** class-map configuration command. To remove protocol-based match criteria from a class map, use the **no match protocol** form of this command.

```
match protocol protocol
no match protocol protocol
```

Syntax Description

<i>protocol</i>	Name of the protocol to match against. The following protocols are supported: aarp —AppleTalk address resolution protocol apollo —Apollo Domain arp —IP ARP bridge —Bridging bstun —Block Serial Tunneling cdp —Cisco Discovery Tunnel clns —ISO Connectionless Network Service clns_es —ISO CLNS End System clns_is —ISO CLNS Intermediate System cmns —ISO Connection-Mode Network Service compressedtcp —Compressed TCP decnet —DECnet decnet_node —DECnet Node decnet_router-I1 —DECnet Router L1 decnet_router-I2 —DECnet Router L2 dls —Data-link switching ip —IP ipx —Novell IPX llc2 —llc2 pad —packet assembler/disassembler links qllc —Qualified Logical Link Control protocol rsrb —Remote Source-Route Bridging snapshot —Snapshot routing support stun —Serial tunnel vines —Banyan VINES xns —Xerox Network Services
-----------------	--

Defaults

No default behavior.

Command Modes

Class-map configuration

Command History

Release	Modification
12.0(5)T	This command was first introduced.

Usage Guidelines

For class-based weighted fair queueing (CBWFQ), you define traffic classes based on match criteria including protocols, ACLs, input interfaces, QoS labels, and EXP field values. Packets satisfying the match criteria for a class constitute the traffic for that class.

The **match protocol** command specifies the name of a protocol to be used as the match criteria against which packets are checked to determine if they belong to the class specified by the class map.

To use the **match protocol** command, you must first enter the **class-map** command to specify the name of the class whose match criteria you want to establish. After you identify the class, you can use the following commands to configure the match criteria.

- **match access-group**
- **match input-interface**
- **match mpls experimental**
- **match protocol**

If you specify more than one command in a class map, only the last command entered applies. The last command overrides the previously entered commands.

Examples

The following example specifies a class map called ipx and configures the ipx protocol as match criteria for it:

```
class-map ipx
  match protocol ipx
```

Related Commands

Command	Description
class-map	Creates a class map to be used for matching packets to a specified class.
match access-group	Configures the match criteria for a class map based on the specified ACL.
match input-interface	Configures a class map to use the specified input interface as a match criterion.
match mpls experimental	Configures a class map to use the specified EXP field value as a match criterion.

policy-map

To create or modify a policy map that can be attached to one or more interfaces to specify a service policy, use the **policy-map** global configuration command. To delete a policy map, use the **no** form of this command.

```
policy-map policy-map
no policy-map policy-map
```

Syntax Description

policy-map Name of the policy map.

Defaults

No default behavior or values.

Command Modes

Global configuration

Command History

Release	Modification
12.0(5)T	This command was first introduced.

Usage Guidelines

Use the **policy-map** command to specify the name of the policy map to be created, added to, or modified before you can configure policies for classes whose match criteria are defined in a class map. Entering the **policy-map** command enables policy map configuration mode in which you can configure or modify the class policies for that policy map.

You can configure class policies in a policy map only if the classes have match criteria defined for them. You use the **class-map** and **match** commands to configure the match criteria for a class. Because you can configure a maximum of 64 class maps, no policy map can contain more than 64 class policies.

A single policy map can be attached to multiple interfaces concurrently. When you attempt to attach a policy map to an interface, the attempt is denied if the available bandwidth on the interface cannot accommodate the total bandwidth requested by class policies comprising the policy map. In this case, if the policy map is already attached to other interfaces, it is removed from them.

Whenever you modify class policy in an attached policy map, CBWFQ is notified and the new classes are installed as part of the policy map in the CBWFQ system.

Examples

The following example creates a policy map called policy1 and configures two class policies included in that policy map. The class policy called Class1 specifies policy for traffic that matches Access Control List 136. The second class is the default class to which packets that do not satisfy configured match criteria are directed.

```
! The following commands create class-map class1 and defines its match criteria:
class-map class1
  match access-group 136

! The following commands create the policy map, which is defined to contain policy
! specification for class1 and the default class:
policy-map policy1

class class1
  bandwidth 2000
  queue-limit 40

class class-default
  fair-queue 16
  queue-limit 20
```

Related Commands

Command	Description
bandwidth (policy-map class)	Specifies or modifies the bandwidth allocated for a class belonging to a policy map.
class (policy-map)	Specifies the name of the class whose policy you want to create or change, and the default class (commonly known as the class-default class) before you configure its policy.
class-map	Creates a class map to be used for matching packets to a specified class.
fair-queue (class-default)	Specifies the number of dynamic queues to be reserved for use by the class-default class as part of the default class policy.
queue-limit	Specifies or modifies the maximum number of packets the queue can hold for a class policy configured in a policy map.
random-detect (interface)	Enables WRED or DWRED.
service-policy	Attaches a policy map to an input interface or VC, or an output interface or VC, to be used as the service policy for that interface or VC.

queue-limit

To specify or modify the maximum number of packets the queue can hold for a class policy configured in a policy map, use the **queue-limit** policy-map configuration command. To remove the queue packet limit from a class, use the **no** form of this command.

queue-limit *number-of-packets*
no queue-limit *number-of-packets*

Syntax Description

number-of-packets A number in the range of 1 to 64 specifying the maximum number of packets that the queue for this class can accumulate.

Defaults

64

Command Modes

Policy-map configuration

Command History

Release	Modification
12.0(5)T	This command was first introduced.

Usage Guidelines

Weighted fair queueing (WFQ) creates a queue for every class for which a class map is defined. Packets satisfying the match criteria for a class accumulate in the queue reserved for the class until they are sent, which occurs when the queue is serviced by the fair queueing process. When the maximum packet threshold you defined for the class is reached, enqueueing of any further packets to the class queue causes tail drop or, if Weighted Random Early Detection (WRED) is configured for the class policy, packet drop to take effect.

Examples

The following example configures a policy map called policy11 to contain policy for a class called acl203. Policy for this class is set so that the queue reserved for it has a maximum packet limit of 40.

```
policy-map policy11
  class acl203
    bandwidth 2000
    queue-limit 40
```

Related Commands

Command	Description
class (policy-map)	Specifies the name of the class whose policy you want to create or change, and the default class (commonly known as the class-default class) before you configure its policy.
policy-map	Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.

random-detect exponential-weighting-constant

To configure the Weighted Random Early Detection (WRED) and VIP-Distributed WRED (DWRED) exponential weight factor for the average queue size calculation for the queue, use the **random-detect exponential-weighting-constant** interface command. To configure the exponential weight factor for the average queue size calculation for the queue reserved for a class, use the **random-detect exponential-weighting-constant** policy map configuration command. To return the value to the default, use the **no** form of this command.

```
random-detect exponential-weighting-constant exponent
no random-detect exponential-weighting-constant
```

Syntax Description

exponent Exponent from 1 to 16 used in the average queue size calculation.

Default

The default exponential weight factor is 9.

Command Mode

Interface configuration when used on an interface.

Policy-map class configuration when used to specify class policy in a policy map.

Command History

Release	Modification
11.1 CC	This command was first introduced.
12.0(5)T	This command was made available as a policy-map class configuration command.

Usage Guidelines

WRED is a congestion avoidance mechanism that slows traffic by randomly dropping packets when congestion exists. WRED and DWRED are most useful with protocols like TCP that respond to dropped packets by decreasing the transmission rate.

Use this command to change the exponent used in the average queue size calculation for the WRED and DWRED services. You can also use this command to configure the exponential weight factor for the average queue size calculation for the queue reserved for a class.

Note The default WRED/DWRED parameter values are based on the best available data. We recommend that you do not change the parameters from their default values unless you have determined that your applications would benefit from the changed values.

The DWRED feature is not supported for class policy.

The DWRED feature is only supported on Cisco 7000 series routers with an RSP7000 card and Cisco 7500 series routers with a VIP2-40 or greater interface processor. A VIP2-50 interface processor is strongly recommended when the aggregate line rate of the port adapters on the VIP is greater than DS3. A VIP2-50 interface processor is required for OC-3 rates.

To use DWRED, distributed Cisco Express Forwarding (dCEF) switching must first be enabled on the interface. For more information on dCEF, refer to the *Cisco IOS Switching Services Configuration Guide* and the *Cisco IOS Switching Services Command Reference*.

Example

The following example configures WRED on an interface with a weight factor of 10:

```
interface Hssi0/0/0
  description 45Mbps to R1
  ip address 10.200.14.250 255.255.255.252
  random-detect
  random-detect exponential-weighting-constant 10
```

The following example configures the policy map policy1 to contain policy specification for the class called class1. During times of congestion, WRED packet drop is used instead of tail drop. The weight factor used for the average queue size calculation for the queue for class1 is 12.

```
! The following commands create the class map called class1:
class-map class1
  match input-interface FE0/1

! The following commands define policy1 to contain policy specification for class1:
policy-map policy1
  class class1
    bandwidth 1000
    random-detect
    random-detect exponential-weighting-constant 12
```

Related Commands

Command	Description
exponential-weighting-constant	Configures the exponential weight factor for the average queue size calculation for a WRED parameter group.
precedence (VC bundle)	Configures precedence levels for a VC member of a bundle, or for a VC class that can be assigned to a VC bundle.
precedence (WRED group)	Configures a WRED group for a particular IP Precedence.
random-detect (interface)	Enables WRED or DWRED.
random-detect (per VC)	Enables per-VC WRED or per-VC DWRED.
random-detect precedence	Configures WRED and DWRED parameters for a particular IP Precedence.
show policy-map	Displays the configuration of all classes comprising the specified service policy map or all classes for all existing policy maps.
show queue	Displays the contents of packets inside a queue for a particular interface or VC.
show queueing	Lists all or selected configured queueing strategies.

random-detect precedence

To configure Weighted Random Early Detection (WRED) and VIP-Distributed WRED (DWRED) parameters for a particular IP Precedence, use the **random-detect precedence** interface configuration command. To return the values to the default for the precedence, use the **no** form of this command.

To configure WRED parameters for a particular IP Precedence for a class policy in a policy map, use the **random-detect precedence** policy-map class configuration command.

```
random-detect precedence precedence min-threshold max-threshold mark-prob-denominator
no random-detect precedence precedence min-threshold max-threshold
mark-prob-denominator
```

Syntax Description

<i>precedence</i>	IP precedence number. The value range is 0 to 7 and RSVP. For Cisco 7000 series routers with an RSP7000 interface processor and Cisco 7500 series routers with a VIP2-40 interface processor (VIP2-50 interface processor strongly recommended), the precedence value range is 0 to 7 only; see Table 3.
<i>min-threshold</i>	Minimum threshold in number of packets. The value range of this argument is 1 to 4096. When the average queue length reaches the minimum threshold, WRED drops some packets with the specified IP Precedence.
<i>max-threshold</i>	Maximum threshold in number of packets. The value range of this argument is the value of the <i>min-threshold</i> argument to 4096. When the average queue length exceeds the maximum threshold, WRED drops all packets with the specified IP Precedence.
<i>mark-prob-denominator</i>	Denominator for the fraction of packets dropped when the average queue depth is at the maximum threshold. For example, if the denominator is 512, one out of every 512 packets is dropped when the average queue is at the maximum threshold. The value range is 1 to 65536. The default is 10; one out of every ten packets is dropped at the maximum threshold.

Default

For all precedences, the *mark-prob-denominator* is 10, and the *max-threshold* is based on the output buffering capacity and the transmission speed for the interface.

The default *min-threshold* depends on the precedence. The *min-threshold* for IP Precedence 0 corresponds to half of the *max-threshold*. The values for the remaining precedences fall between half the *max-threshold* and the *max-threshold* at evenly spaced intervals.

Table 3 lists the default minimum threshold value for each IP Precedence.

Table 3 Default WRED and DWRED Minimum Threshold Values

IP Precedence	Minimum Threshold Value (Fraction of Maximum Threshold Value)	
	WRED	DWRED
0	9/18	8/16
1	10/18	9/16
2	11/18	10/16
3	12/18	11/16
4	13/18	12/16
5	14/18	13/16
6	15/18	14/16
7	16/18	15/16
RSVP	17/18	N/A

Command Mode

Interface configuration when used on an interface.

Policy-map class configuration when used to specify class policy in a policy map.

Command History

Release	Modification
11.1 CC	This command was first introduced.

Usage Guidelines

WRED is a congestion avoidance mechanism that slows traffic by randomly dropping packets when congestion exists. DWRED is similar to WRED but uses the Versatile Interface Processor (VIP) instead of the Route Switch Processor (RSP).

When you configure the **random-detect** command on an interface, packets are given preferential treatment based on the IP Precedence of the packet. Use the **random-detect precedence** command to adjust the treatment for different precedences.

If you want WRED/DWRED to ignore the precedence when determining which packets to drop, enter this command with the same parameters for each precedence. Remember to use reasonable values for the minimum and maximum thresholds.

Note that if you use the **random-detect precedence** command to adjust the treatment for different precedences within class policy, you must ensure that WRED is not configured for the interface to which you attach that service policy.

Note The default WRED/DWRED parameter values are based on the best available data. We recommend that you do not change the parameters from their default values unless you have determined that your applications would benefit from the changed values.

The DWRED feature is only supported on Cisco 7000 series routers with an RSP7000 card and Cisco 7500 series routers with a VIP2-40 or greater interface processor. A VIP2-50 interface processor is strongly recommended when the aggregate line rate of the port adapters on the VIP is greater than DS3. A VIP2-50 interface processor is required for OC-3 rates.

To use DWRED, distributed Cisco Express Forwarding (dCEF) switching must first be enabled on the interface. For more information on dCEF, refer to the *Cisco IOS Switching Services Configuration Guide* and the *Cisco IOS Switching Services Command Reference*.

Note The DWRED feature is not support in class policy.

Example

The following example enables WRED on the interface and specifies parameters for the different IP Precedences:

```
interface Hssi0/0/0
description 45Mbps to R1
ip address 200.200.14.250 255.255.255.252
random-detect
random-detect precedence 0 32 256 100
random-detect precedence 1 64 256 100
random-detect precedence 2 96 256 100
random-detect precedence 3 120 256 100
random-detect precedence 4 140 256 100
random-detect precedence 5 170 256 100
random-detect precedence 6 290 256 100
random-detect precedence 7 210 256 100
random-detect precedence rsvp 230 256 100
```

The following example configures policy for a class named `acl10` included in the policy map called `policy10`. Class `acl10` has these characteristics: a minimum of 2000 kbps of bandwidth are expected to be delivered to this class in the event of congestion and a weight factor of 10 is used to calculate the average queue size. For congestion avoidance, WRED packet drop is used, not tail drop. IP Precedence is reset for levels 0 through 4.

```
policy-map policy10
class acl10
bandwidth 2000
random-detect
random-detect exponential-weighting-constant 10
random-detect precedence 0 32 256 100
random-detect precedence 1 64 256 100
random-detect precedence 2 96 256 100
random-detect precedence 3 120 256 100
random-detect precedence 4 140 256 100
```

Related Commands

Command	Description
bandwidth (policy-map class)	Specifies or modifies the bandwidth allocated for a class belonging to a policy map.
random-detect (interface)	Enables WRED or DWRED.
random-detect (per VC)	Enables per-VC WRED or per-VC DWRED.
random-detect-exponential-weighting-constant	Configures the WRED and DWRED exponential weight factor for the average queue size calculation.
show policy-map interface	Displays the configuration of classes configured for service policies on the specified interface or PVC.
show queue	Displays the contents of packets inside a queue for a particular interface or VC.
show queueing	Lists all or selected configured queueing strategies.

service-policy

To attach a policy map to an input interface or virtual circuit (VC), or an output interface to be used as the service policy for that interface or VC, use the **service-policy** global configuration command. To remove a service policy from an input or output interface or input or output VC, use the **no** form of this command.

```
service-policy {input | output} policy-map
no service-policy {input | output}
```

Syntax Description

input	Attaches the specified policy map to the input interface or input VC.
output	Attaches the specified policy map to the output interface or output VC.
<i>policy-map</i>	The name of a service policy map (created using the policy-map command) to be attached.

Defaults

No service policy is specified.

Command Modes

Global configuration.

VC submode (for a standalone VC).

Bundle-vc configuration (for ATM VC bundle members).

Command History

Release	Modification
12.0(5)T	This command was first introduced.

Usage Guidelines

You can attach a single policy map to one or more interfaces or one or more VCs to specify the service policy for those interfaces or VCs.

Currently a service policy specifies class-based weighted fair queueing (CBWFQ). The class policies comprising the policy map are then applied to packets that satisfy the class map match criteria for the class.

To successfully attach a policy map to an interface or a VC, the aggregate of the configured minimum bandwidths of the classes comprising the policy map must be less than or equal to 75 percent of the interface bandwidth allocated to the VC.

Configuring CBWFQ on a physical interface is only possible if the interface is in the default queueing mode. Serial interfaces at E1 (2.048 Mbps) and below use WFQ by default. Other interfaces use FIFO by default. Enabling CBWFQ on a physical interface overrides the default interface queueing method. Enabling CBWFQ on an ATM permanent virtual circuit (PVC) does not override the default queueing method.

Attaching a service policy and enabling CBWFQ on an interface renders ineffective any commands related to fancy queueing such as commands pertaining to fair queueing, custom queueing, priority queueing, and Weighted Random Early Detection (WRED). You can configure these features only after you remove the policy map from the interface.

You can modify a policy map attached to an interface or VC, changing the bandwidth of any of the classes comprising the map. Bandwidth changes that you make to an attached policy map are effective only if the aggregate of the bandwidth amounts for all classes comprising the policy map, including the modified class bandwidth, is less than or equal to 75 percent of the interface bandwidth. If the new aggregate bandwidth amount exceeds 75 percent of the interface bandwidth, the policy map is not modified.

Examples

The following example attaches the service policy map called policy9 to the input interface Serial1:

```
interface Serial1
 service-policy input policy9
```

The following example attaches the service policy map called policy9 to the input PVC called cisco:

```
pvc cisco 0/34
 service-policy input policy9
 vbr-nt 5000 3000 500
 precedence 4-7
```

The following example attaches the policy called policy9 to the output interface serial1 to specify the service policy for the interface and enable CBWFQ on it:

```
interface serial1
 service-policy output policy9
```

The following example attaches the service policy map called policy9 to the output PVC called cisco:

```
pvc cisco 0/5
 service-policy output policy9
 vbr-nt 4000 2000 500
 precedence 2-3
```

Related Commands

Command	Description
policy-map	Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.
show policy-map	Displays the configuration of all classes comprising the specified service policy map or all classes for all existing policy maps.

show policy-map

To display the configuration of all classes comprising the specified service policy map or all classes for all existing policy maps, use the **show policy** EXEC or privileged EXEC command.

```
show policy [policy-map]
```

Syntax Description

policy-map (Optional) The name of the service policy map whose complete configuration is to be displayed.

Defaults

All existing policy map configurations are displayed.

Command Modes

EXEC or privileged EXEC

Command History

Release	Modification
12.0(5)T	This command was first introduced.

Usage Guidelines

The **show policy-map** command displays the configuration of a service policy map created using the **policy-map** command. You can use the **show policy-map** command to display all class configurations comprising any existing service policy map, whether or not that service policy map has been attached to an interface.

Examples

The following example displays the contents of the pol1 service policy map:

```
Router# show policy-map pol1
Policy Map pol1
  Weighted Fair Queueing
    Class class1
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class2
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class3
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class4
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class5
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class6
      Bandwidth 937 (kbps) Max thresh 64 (packets)
```

```
Class class7
  Bandwidth 937 (kbps) Max thresh 64 (packets)
Class class8
  Bandwidth 937 (kbps) Max thresh 64 (packets)
```

The following example displays the contents of all policy maps on the router:

```
Router# show policy-map
Policy Map poH1
  Weighted Fair Queueing
    Class class1
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class2
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class3
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class4
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class5
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class6
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class7
      Bandwidth 937 (kbps) Max thresh 64 (packets)
    Class class8
      Bandwidth 937 (kbps) Max thresh 64 (packets)
Policy Map policy2
  Weighted Fair Queueing
    Class class1
      Bandwidth 300 (kbps) Max thresh 64 (packets)
    Class class2
      Bandwidth 300 (kbps) Max thresh 64 (packets)
    Class class3
      Bandwidth 300 (kbps) Max thresh 64 (packets)
    Class class4
      Bandwidth 300 (kbps) Max thresh 64 (packets)
    Class class5
      Bandwidth 300 (kbps) Max thresh 64 (packets)
    Class class6
      Bandwidth 300 (kbps) Max thresh 64 (packets)
```

Related Commands

Command	Description
show policy-map class	Displays the configuration for the specified class of the specified policy map.
show policy-map interface	Displays the configuration of classes configured for service policies on the specified interface or PVC.

show policy-map class

To display the configuration for the specified class of the specified policy map, use the **show policy-map class** EXEC or privileged EXEC command.

```
show policy-map policy-map class class-name
```

Syntax Description

<i>policy-map</i>	The name of a policy map that contains the class configuration to be displayed.
<i>class-name</i>	The name of the class whose configuration is to be displayed.

Defaults

No default behavior or values.

Command Modes

EXEC or privileged EXEC

Command History

Release	Modification
12.0(5)T	This command was first introduced.

Usage Guidelines

You can use the **show policy-map class** command to display any single class configuration for any service policy map, whether or not the specified service policy map has been attached to an interface.

Examples

The following example displays configurations for the class called class7 that belongs to the policy map po1.

```
Router# show policy-map po1 class class7
Class class7
  Bandwidth 937 (kbps) Max Thresh 64 (packets)
```

Related Commands

Command	Description
show policy-map	Displays the configuration of all classes comprising the specified service policy map or all classes for all existing policy maps.
show policy-map interface	Displays the configuration of classes configured for service policies on the specified interface or PVC.

show policy-map interface

To display the configuration of all classes configured for all service policies on the specified interface or to display the classes for the service policy for a specific permanent virtual circuit (PVC) on the interface, use the **show policy-map interface** EXEC or privileged EXEC command.

```
show policy-map interface interface-name [vc [vpi/] vci]
```

Syntax Description

<i>interface-name</i>	Name of the interface or subinterface whose policy configuration is to be displayed.
vc	(Optional) For ATM interfaces only, shows the policy configuration for a specified PVC. The name can be up to 16 characters long.
<i>vpi/</i>	(Optional) ATM network virtual path identifier (VPI) for this PVC. The absence of the “/” and a <i>vpi</i> value defaults the <i>vpi</i> value to 0. On the Cisco 7200 and 7500 series routers, this value ranges from 0 to 255. The <i>vpi</i> and <i>vci</i> arguments cannot both be set to 0; if one is 0, the other cannot be 0. If this value is omitted, information for all VCs on the specified ATM interface or subinterface is displayed.
<i>vci</i>	(Optional) ATM network virtual channel identifier (VCI) for this PVC. This value ranges from 0 to 1 less than the maximum value set for this interface by the atm vc-per-vc command. Typically, lower values 0 to 31 are reserved for specific traffic (F4 Operation, Administration, and Maintenance (OAM), switched virtual circuit (SVC) signalling, Integrated Local Management Interface (ILMI), and so on) and should not be used. The VCI is a 16-bit field in the header of the ATM cell. The VCI value is unique only on a single link, not throughout the ATM network, because it has local significance only. The <i>vpi</i> and <i>vci</i> arguments cannot both be set to 0; if one is 0, the other cannot be 0.

Defaults

No default behavior.

Command Modes

EXEC or privileged EXEC.

Command History

Release	Modification
12.0(5)T	This command was first introduced.

Usage Guidelines

This command displays the configuration for classes on the specified interface or the specified PVC only if a service policy has been attached to the interface or the PVC.

You can use the *pvc-name* argument to display output for a PVC only for Enhanced ATM port adapters (PA-A3) that support per-VC queueing.

The counters displayed after the **show policy-map interface** command is entered are updated only if congestion is present on the interface.

Examples

The following example displays configurations for classes on the output interface e1/1:

```
Router# show policy-map interface e1/1
Ethernet1/1 output : pol
Weighted Fair Queueing
Class class1
  Output Queue: Conversation 264
  Bandwidth 937 (kbps) Max Threshold 64 (packets)
  (total/discards/tail drops) 11548/0/0
Class class2
  Output Queue: Conversation 265
  Bandwidth 937 (kbps) Max Threshold 64 (packets)
  (total/discards/tail drops) 11546/0/0
Class class3
  Output Queue: Conversation 266
  Bandwidth 937 (kbps) Max Threshold 64 (packets)
  (total/discards/tail drops) 11546/0/0
Class class4
  Output Queue: Conversation 267
  Bandwidth 937 (kbps) Max Threshold 64 (packets)
  (total/discards/tail drops) 11702/0/0
Class class5
  Output Queue: Conversation 268
  Bandwidth 937 (kbps) Max Threshold 64 (packets)
  (total/discards/tail drops) 11701/0/0
Class class6
  Output Queue: Conversation 269
  Bandwidth 937 (kbps) Max Threshold 64 (packets)
  (total/discards/tail drops) 11702/0/0
Class class7
  Output Queue: Conversation 270
  Bandwidth 937 (kbps) Max Threshold 64 (packets)
  (total/discards/tail drops) 11857/0/0
Class class8
  Output Queue: Conversation 271
  Bandwidth 937 (kbps) Max Threshold 64 (packets)
  (total/discards/tail drops) 11858/1/0
```

show policy-map interface

The following example displays configurations for classes comprising the service policy for the output VC 0/101 on the output interface atm2/0.6:

```
Router# show policy-map interface atm2/0.6
ATM2/0.6: VC 0/101 - output : p1
  Weighted Fair Queueing
    Class c-vcl-c1
      Output Queue: Conversation 264
      Bandwidth 31 (kbps)
      mean queue depth: 1
      drops: class random tail min-th max-th mark-prob
             0      0      0    100   200   1/10
             1      0      0    105   200   1/10
             2      0      0    110   200   1/10
             3      0      0    115   200   1/10
             4      0      0    120   200   1/10
             5      0      0    125   200   1/10
             6      0      0    130   200   1/10
             7      0      0    135   200   1/10
             rsvp  0      0    140   200   1/10
    Class c-vcl-c2
      Output Queue: Conversation 265
      Bandwidth 54 (kbps)
      mean queue depth: 1
      drops: class random tail min-th max-th mark-prob
             0      0      0     60   100   1/10
             1      0      0     65   100   1/10
             2      0      0     70   100   1/10
             3      0      0     75   100   1/10
             4      0      0     80   100   1/10
             5      0      0     83   100   1/10
             6      0      0     85   100   1/10
             7      0      0     87   100   1/10
             rsvp  0      0     90   100   1/10
    Class c-vcl-c3
      Output Queue: Conversation 266
      Bandwidth 77 (kbps)
      mean queue depth: 0
      drops: class random tail min-th max-th mark-prob
             0      0      0      1    10   1/10
             1      0      0      2    10   1/10
             2      0      0      3    10   1/10
             3      0      0      4    10   1/10
             4      0      0      5    10   1/10
             5      0      0      6    10   1/10
             6      0      0      7    10   1/10
             7      0      0      7    10   1/10
             rsvp  0      0      7    10   1/10
    Class c-vcl-c4
      Output Queue: Conversation 267
      Bandwidth 100 (kbps)
      mean queue depth: 9
      drops: class random tail min-th max-th mark-prob
             0      0      0      1    10   1/10
             1      9     220     2    10   1/10
             2     24     645     3    10   1/10
             3     22     844     4    10   1/10
             4      0      0      5    10   1/10
             5     23     351     6    10   1/10
             6     28     213     7    10   1/10
             7     59     540     7    10   1/10
             rsvp  0      0      7    10   1/10
```

```

Class c-vc1-c5
  Output Queue: Conversation 268
    Bandwidth 123 (kbps)
  mean queue depth: 150
  drops: class  random  tail    min-th  max-th  mark-prob
         0      120    1777   50     150    1/50
         1      136    1549   60     150    1/50
         2       88    2354   70     150    1/50
         3      121    1569   80     150    1/50
         4      122    1717   80     150    1/50
         5       0      0      90     150    1/50
         6       0      0     100    150    1/50
         7      105    2058  110     150    1/50
         rsvp    0      0     120    150    1/50

Class c-vc1-c6
  Output Queue: Conversation 269
    Bandwidth 146 (kbps) Max Threshold 64 (packets)
  (total/discards/tail drops) 50216/32696/0

Class c-vc1-c7
  Output Queue: Conversation 270
    Bandwidth 216 (kbps) Max Threshold 64 (packets)
  (total/discards/tail drops) 74577/51994/0

Class class-default
  Flow Based Fair Queueing
  Number of Hashed Queues 256
  drops: class  random  tail    min-th  max-th  mark-prob
         0      101    828   50     150    1/50
         1       87    1154  60     150    1/50
         2      115    476   70     150    1/50
         3      116    444   80     150    1/50
         4      123    338   80     150    1/50
         5       92    1042  90     150    1/50
         6       79    1068  100    150    1/50
         7      110    740   110    150    1/50
         rsvp    0      0     120    150    1/50

```

Related Commands

Command	Description
show policy-map	Displays the configuration of all classes comprising the specified service policy map or all classes for all existing policy maps.
show policy-map class	Displays the configuration for the specified class of the specified policy map.

