# 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

| | |
|---|---|
| *access-group* | A numbered ACL whose contents are used as the match criteria against which packets are checked to determine if they belong to this class. |
| **name** *access-group-name* | 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 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 or named 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 its 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 a 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 a 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 introduced. |

## Usage Guidelines

For class-based weighted fair queueing (CBWFQ), you define traffic classes based on match criteria including input interfaces, Access Control Lists (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 one of the following commands to configure its 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 ethernet1to 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 mpls experimental

To configure a class map to use the specified value of the EXP field as a match criterion, use the **match mpls experimental** class-map configuration command. To remove the EXP field match criterion from a class map, use the **no** form of this command.

**match mpls experimental** *number*

**no match mpls experimental** *number*

## Syntax Description

| | |
|---|---|
| *number* | The EXP field value to be used as match criteria. Any number from 0 to 7. |

## Defaults

No default behavior.

## Command Modes

Class-map configuration

## Command History

| Release | Modification |
|---|---|
| 12.0(7)T | This command was introduced. |

## Usage Guidelines

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

The **match mpls experimental** command specifies the name of an EXP field value 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 mpls experimental** 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 its 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 EXP field value 0 to be used as the match criterion for this class:

```
class-map eth1
 match mpls experimental 1
```

**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 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** form of this command.

**match protocol** *protocol*

**no match protocol** *protocol*

## Syntax Description

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

**Defaults**

No default behavior.

**Command Modes**

Class-map configuration

**Command History**

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

**Usage Guidelines**

For class-based weighted fair queueing (CBWFQ), you define traffic classes based on match criteria including protocols, Access Control Lists (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 one of the following commands to configure its 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 internetwork packet exchange (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. |

# max-reserved-bandwidth

To change the percent of interface bandwidth allocated for class-based weighted fair queueing (CBWFQ), low latency queueing (LLQ), and IP RTP Priority, use the **max-reserved bandwidth** interface configuration command. To restore the default value, use the **no** form of this command.

**max-reserved-bandwidth** *percent*

**no max-reserved-bandwidth**

## Syntax Description

| | |
|---|---|
| *percent* | Percent of interface bandwidth allocated for CBWFQ, LLQ, and IP RTP Priority. |

## Defaults

75 percent

## Command Modes

Interface configuration

## Command History

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

## Usage Guidelines

The sum of all bandwidth allocation on an interface should not exceed 75 percent of the available bandwidth on an interface. The remaining 25 percent of bandwidth is used for overhead, including Layer 2 overhead, control traffic, and best-effort traffic.

If you need to allocate more than 75 percent for CBWFQ, LLQ, and IP RTP Priority, you can use the **max-reserved-bandwidth** command. The *percent* argument specifies the maximum percentage of the total interface bandwidth that can be used by CBWFQ classes, LLQ, and IP RTP Priority.

If you do use the **max-reserved-bandwidth** command, make sure that not too much bandwidth is taken away from best-effort and control traffic.

The **max-reserved-bandwidth** command is intended for use on main interfaces only; it has no effect on virtual circuits (VCs) or ATM permanent virtual circuits (PVCs).

## Examples

In the following example, the policy1 policy map is configured for three classes with a total of 8 Mbps configured bandwidth, as shown in the output from the **show policy-map** command:

```
Router# show policy-map policy1
 Policy Map policy1
  Weighted Fair Queueing
    Class class1
      Bandwidth 2500 (kbps) Max Threshold 64 (packets)
    Class class2
      Bandwidth 2500 (kbps) Max Threshold 64 (packets)
    Class class3
      Bandwidth 3000 (kbps) Max Threshold 64 (packets)
```

When you enter the **service-policy output** command in an attempt to attach the policy map on a 10-Mbps Ethernet interface, an error message such as the following is produced:

```
I/f Ethernet1/1 class class3 requested bandwidth 3000 (kbps) Available only 2500 (kbps)
```

The error message is produced because the default maximum configurable bandwidth is 75 percent of the available interface bandwidth, which in this example is 7.5 Mbps. To change the maximum configurable bandwidth to 80 percent, use the **max-reserved-bandwidth** command in interface configuration mode, as follows:

```
max-reserved-bandwidth 80
service output policy1
end
```

To verify that the policy map was attached, enter the **show policy-map interface** command:

```
Router# show policy-map interface e1/1
 Ethernet1/1  output :policy1
  Weighted Fair Queueing
    Class class1
      Output Queue:Conversation 265
        Bandwidth 2500 (kbps) Packets Matched 0 Max Threshold 64 (packets)
        (discards/tail drops) 0/0
    Class class2
      Output Queue:Conversation 266
        Bandwidth 2500 (kbps) Packets Matched 0 Max Threshold 64 (packets)
        (discards/tail drops) 0/0
    Class class3
      Output Queue:Conversation 267
        Bandwidth 3000 (kbps) Packets Matched 0 Max Threshold 64 (packets)
        (discards/tail drops) 0/0
```

### Virtual Template Configuration Example

The following example configures a strict priority queue in a virtual template configuration with CBWFQ. The **max-reserved-bandwidth** command changes the maximum bandwidth allocated between CBWFQ and IP RTP Priority from the default (75 percent) to 80 percent.

```
multilink virtual-template 1
interface virtual-template 1
 ip address 172.16.1.1 255.255.255.0
 no ip directed-broadcast
 ip rtp priority 16384 16383 25
 service-policy output policy1
 ppp multilink
 ppp multilink fragment-delay 20
 ppp multilink interleave
 max-reserved-bandwidth 80
 end

interface Serial0/1
 bandwidth 64
 ip address 10.1.1.2 255.255.255.0
 no ip directed-broadcast
 encapsulation ppp
 ppp multilink
 end
```

To make the virtual access interface function properly, do not configure the **bandwidth** command on the virtual template. Configure it on the actual interface, as shown in the example.

## Related Commands

| Command | Description |
|---|---|
| **bandwidth (policy-map class)** | Specifies or modifies the bandwidth allocated for a class belonging to a policy map. |
| **ip rtp priority** | Reserves a strict priority queue for a set of RTP packet flows belonging to a range of UDP destination ports. |

# oam-bundle

To enable end-to-end F5 Operation, Administration, and Maintenance (OAM) loopback cell generation and OAM management for a virtual circuit (VC) class that can be applied to a VC bundle, use the **oam-bundle** vc-class configuration command. To remove OAM management from the class configuration, use the **no** form of this command.

To enable end-to-end F5 OAM loopback cell generation and OAM management for all VC members of a bundle, use the **oam-bundle** bundle configuration command. To remove OAM management from the bundle, use the **no** form of this command.

**oam-bundle** [**manage**] [*frequency*]

**no oam-bundle** [**manage**] [*frequency*]

## Syntax Description

| | |
|---|---|
| **manage** | (Optional) Enables OAM management. If omitted, loopback cells are sent but the bundle is not managed. |
| *frequency* | (Optional) Number of seconds between sending OAM loopback cells. Values range from 0 to 600 seconds. |

## Defaults

End-to-end F5 OAM loopback cell generation and OAM management are disabled, but if OAM cells are received, they are looped back. The default value for the *frequency* argument is 10 seconds.

## Command Modes

VC-class configuration (for a VC class).

Bundle configuration (for an ATM VC bundle).

## Command History

| Release | Modification |
|---|---|
| 12.0(3)T | This command was introduced. |

## Usage Guidelines

This command defines whether a VC bundle is OAM-managed. If this command is configured for a bundle, every VC member of the bundle is OAM-managed. If OAM management is enabled, further control of OAM management is configured using the **oam retry** command.

This command has no effect if the VC class that contains the command is attached to a standalone VC, that is, if the VC is not a bundle member. In this case, the attributes are ignored by the VC.

To use this command in bundle configuration mode, enter the **bundle** subinterface configuration command to create the bundle or to specify an existing bundle before you enter this command.

To use this command in vc-class configuration mode, first enter the **vc-class atm** global configuration command.

VCs in a VC bundle are subject to the following configuration inheritance rules (listed in order of next highest precedence):

- VC configuration in bundle-vc mode
- Bundle configuration in bundle mode (with effect of assigned vc-class configuration)

## Examples

The following example enables OAM management for a bundle called chicago:

```
bundle chicago
 oam-bundle manage
```

## Related Commands

| Command | Description |
|---|---|
| **broadcast** | Configures broadcast packet duplication and transmission for an ATM VC class, PVC, SVC, or VC bundle. |
| **class-bundle** | Configures a VC bundle with the bundle-level commands contained in the specified VC class. |
| **encapsulation** | Sets the encapsulation method used by the interface. |
| **inarp** | Configures the Inverse ARP time period for an ATM PVC, VC class, or VC bundle. |
| **oam retry** | Configures parameters related to OAM management for an ATM PVC, SVC, VC class, or VC bundle. |
| **protocol (ATM)** | Configures a static map for an ATM PVC, SVC, VC class, or VC bundle. Enables Inverse ARP or Inverse ARP broadcasts on an ATM PVC by either configuring Inverse ARP directly on the PVC, on the VC bundle, or in a VC class (applies to IP and IPX protocols only). |

# 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. The name can be a maximum of 40 alphanumeric characters. |

## Defaults

No default behavior or values.

## Command Modes

Global configuration

## Command History

| Release | Modification |
|---|---|
| 12.0(5)T | This command was 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. |

# precedence (VC bundle)

To configure precedence levels for a virtual circuit (VC) class that can be assigned to a VC bundle and thus applied to all VC members of that bundle, use the **precedence** vc-class configuration command. To remove the precedence levels from the VC class, use the **no** form of this command.

To configure the precedence levels for a VC member of a bundle, use the **precedence** bundle-vc configuration command. To remove the precedence levels from the VC, use the **no** form of this command.

**precedence** [**other** | *range*]

**no precedence**

## Syntax Description

| | |
|---|---|
| **other** | (Optional) Any precedence levels in the range 0 to 7 that are not explicitly configured. This is the default. |
| *range* | (Optional) A single precedence level specified as a number, or a range of precedence levels, specified as a hyphenated range. |

## Defaults

Defaults to **other**, that is, any precedence levels in the range 0 - 7 that are not explicitly configured.

## Command Modes

VC-class configuration (for a VC class).

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

## Command History

| Release | Modification |
|---|---|
| 11.1(22)CC | This command was introduced. |
| 12.0(3)T | This command was extended to configure precedence levels for a VC member of a bundle. |

## Usage Guidelines

Assignment of precedence levels to VC bundle members allows you to create differentiated service because you can distribute the IP Precedence levels over the different VC bundle members. You can map a single precedence level or a range of levels to each discrete VC in the bundle, thereby enabling VCs in the bundle to carry packets marked with different precedence levels. Alternatively, you can configure a VC with the **precedence other** command to indicate that it can carry traffic marked with precedence levels not specifically configured for it. Only one VC in the bundle can be configured with the **precedence other** command to carry all precedence levels not specified. This VC is considered the default one.

To use this command in vc-class configuration mode, enter the **vc-class atm** global configuration command before you enter this command. This command has no effect if the VC class that contains the command is attached to a standalone VC, that is, if the VC is not a bundle member.

To use this command to configure an individual bundle member in bundle-vc configuration mode, first enter the **bundle** command to enact bundle configuration mode for the bundle to which you want to add or modify the VC member to be configured. Then, use the **pvc-bundle** command to specify the VC to be created or modified and enter bundle-vc configuration mode.

VCs in a VC bundle are subject to the following configuration inheritance rules (listed in order of next highest precedence):

- VC configuration in bundle-vc mode
- Bundle configuration in bundle mode (with effect of assigned vc-class configuration)
- Subinterface configuration in subinterface mode

## Examples

The following example configures a class called control-class that includes a **precedence** command that, when applied to a bundle, configures all VC members of that bundle to carry IP Precedence level 7 traffic. Note, however, that VC members of that bundle can be individually configured with the **precedence** command at the bundle-vc level, which would supervene.

```
vc-class atm control-class
 precedence 7
```

The following example configures permanent virtual circuit (PVC) 401 (with the name of control-class) to carry traffic with IP Precedence levels in the range of 4-2, overriding the precedence level mapping set for the VC through vc-class configuration.

```
pvc-bundle control-class 401
 precedence 4-2
```

## Related Commands

| Command | Description |
|---|---|
| **bump** | Configures the bumping rules for a VC class that can be assigned to a VC bundle. |
| **class-vc** | Assigns a VC class to an ATM PVC, SVC, or VC bundle member. |
| **protect** | Configures a VC class with protected group or protected VC status for application to a VC bundle member. |
| **ubr** | Configures UBR QoS and specifies the output peak cell rate for an ATM PVC, SVC, VC class, or VC bundle member. |
| **ubr+** | Configures UBR QoS and specifies the output peak cell rate and output minimum guaranteed cell rate for an ATM PVC, SVC, VC class, or VC bundle member. |
| **vbr-nrt** | Configures the VBR-NRT QoS and specifies output peak cell rate, output sustainable cell rate, and output maximum burst cell size for an ATM PVC, SVC, VC class, or VC bundle member. |

# precedence (WRED group)

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

**precedence** *precedence min-threshold max-threshold mark-probability-denominator*

**no precedence** *precedence min-threshold max-threshold mark-probability-denominator*

## Syntax Description

| | |
|---|---|
| *precedence* | IP Precedence number. Values range from 0 to 7. |
| *min-threshold* | Minimum threshold in number of packets. Value range from 1 to 4096. When the average queue length reaches this number, WRED/DWRED begins to drop packets with the specified IP Precedence. |
| *max-threshold* | Maximum threshold in number of packets. The value range is *min-threshold* to 4096. When the average queue length exceeds this number, WRED/DWRED drops all packets with the specified IP Precedence. |
| *mark-probability-denominator* | Denominator for the fraction of packets dropped when the average queue depth is *max-threshold*. For example, if the denominator is 512, one out of every 512 packets is dropped when the average queue is at the *max-threshold*. The value is 1 to 65536. The default is 10; one out of every ten packets is dropped at the *max-threshold*. |

## Defaults

For all IP Precedences, the *mark-probability-denominator* argument is 10, and the *max-threshold* argument is based on the output buffering capacity and the transmission speed for the interface.

The default *min-threshold* argument depends on the IP Precedence. The *min-threshold* argument for IP Precedence 0 corresponds to half of the *max-threshold* argument. The values for the remaining IP Precedences fall between half the *max-threshold* argument and the *max-threshold* argument at evenly spaced intervals. Table 7 lists the default minimum value for each IP Precedence.

***Table 7 Default WRED Minimum Threshold Values***

| IP Precedence | Minimum Threshold Value (Fraction of Maximum Threshold Value) |
|---|---|
| 0 | 8/16 |
| 1 | 9/16 |
| 2 | 10/16 |
| 3 | 11/16 |
| 4 | 12/16 |
| 5 | 13/16 |

*Table 7 Default WRED Minimum Threshold Values (continued)*

| IP Precedence | Minimum Threshold Value (Fraction of Maximum Threshold Value) |
|---|---|
| 6 | 14/16 |
| 7 | 15/16 |

**Command Modes**

Random-detect-group configuration

**Command History**

| Release | Modification |
|---|---|
| 11.1(22)CC | This command was 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).

If used, this command is issued after the **random-detect-group** command.

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

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

**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.

**Examples**

The following example specifies parameters for WRED parameter group sanjose for the different IP Precedences:

```
random-detect-group sanjose
  precedence 0 32 256 100
  precedence 1 64 256 100
  precedence 2 96 256 100
  precedence 3 128 256 100
  precedence 4 160 256 100
  precedence 5 192 256 100
  precedence 6 224 256 100
  precedence 7 256 256 100
```

**Related Commands**

| Command | Description |
|---|---|
| **exponential-weighting-constant** | Configures the exponential weight factor for the average queue size calculation for a WRED parameter group. |
| **random-detect (per-VC)** | Enables per-VC WRED or per-VC DWRED. |
| **random-detect-group** | Defines the WRED or DWRED parameter group. |
| **random-detect precedence** | Configures WRED and DWRED parameters for a particular IP Precedence. |
| **show queueing** | Lists all or selected configured queueing strategies. |
| **show queueing interface** | Displays the queueing statistics of an interface or VC. |

# priority

To give priority to a class within a policy map, use the **priority** policy-map class configuration command. To disable the strict priority queue, use the **no** form of this command.

**priority** *bandwidth*

**no priority** [*bandwidth*]

**Syntax Description**

| | |
|---|---|
| *bandwidth* | Guaranteed allowed bandwidth (in kbps) for the priority traffic. Beyond the guaranteed bandwidth, the priority traffic will be dropped in the event of congestion to ensure that the nonpriority traffic is not starved. |

**Defaults**

No default behavior or values.

**Command Modes**

Policy-map class configuration

**Command History**

| Release | Modification |
|---|---|
| 12.0(6)T | This command was introduced. |

**Usage Guidelines**

This command configures low latency queueing (LLQ), providing strict priority queueing for class-based weighted fair queueing (CBWFQ). Strict priority queueing allows delay-sensitive data such as voice to be dequeued and sent first (before packets in other queues are dequeued), giving delay-sensitive data preferential treatment over other traffic.

The **priority** command allows you to set up classes based on a variety of criteria (not just UDP ports) and assign priority to them, and is available for use on serial interfaces and ATM permanent virtual circuits (PVCs). A similar command, **ip rtp priority**, allows you to stipulate priority flows based only on User Datagram Protocol (UDP) port numbers and is not available for ATM PVCs.

The *bandwidth* argument is used to specify the maximum amount of bandwidth allocated for packets belonging to a class configured with the **priority** command. The bandwidth parameter both guarantees bandwidth to the priority class and restrains the flow of packets from the priority class.

When the device is not congested, the priority class traffic is allowed to exceed its allocated bandwidth. When the device is congested, the priority class traffic above the allocated bandwidth is discarded.

Remember the following guidelines when using the **priority** command:

- Layer 2 encapsulations are accounted for in the amount of bandwidth specified with the **priority** command. However, the amount of bandwidth does not include other headers such as ATM cell tax overheads. You must also allow bandwidth for possible jitter introduced by the routers in the voice path.
- The **priority** command can be used for Voice over IP (VoIP) on serial links and ATM PVCs. The **priority** command does not support VoIP over Frame Relay links.

- The **random-detect**, **queue-limit**, and **bandwidth** commands cannot be used while the **priority** command is configured.
- The **priority** command can be configured in multiple classes, but it should only be used for voice-like, constant bit rate (CBR) traffic.

## Examples

The following example configures strict priority queueing with a guaranteed bandwidth of 50 kbps for the policy map called policy1:

```
policy-map policy1
 class voice
 priority 50
```

Configuring the **priority** command in multiple classes provides the ability to police the priority classes individually. For an example, refer to the following configuration:

```
policy-map policy1
 class voice1
  priority 24
 class voice2
  priority 48
 class data
  bandwidth 20
```

In this example, voice1 and voice2 classes of traffic go into the high priority queue and get strict priority queueing over data traffic. However, voice1 traffic will be rate-limited to 24 kbps and voice2 traffic will be rate-limited to 48 kbps. The classes will be individually rate-limited (and given first-in first-out [FIFO] treatment) even if they go into the same queue.

## Related Commands

| Command | Description |
|---|---|
| **ip rtp priority** | Reserves a strict priority queue for a set of RTP packet flows belonging to a range of UDP destination ports. |
| **ip rtp reserve** | Reserves a special queue for a set of RTP packet flows belonging to a range of UDP destination ports. |
| **max-reserved-bandwidth** | Changes the percent of interface bandwidth allocated for CBWFQ, LLQ, and IP RTP Priority. |
| **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. |

# priority-group

To assign the specified priority list to an interface, use the **priority-group** interface configuration command. To remove the specified priority group assignment, use the **no** form of this command.

**priority-group** *list-number*

**no priority-group** *list-number*

**Syntax Description**

| | |
|---|---|
| *list-number* | Priority list number assigned to the interface. Any number from 1 to 16. |

**Defaults**

This command is disabled by default.

**Command Modes**

Interface configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |

**Usage Guidelines**

Only one list can be assigned per interface. Priority output queueing provides a mechanism to prioritize packets sent on an interface.

Use the **show queueing priority** and **show interfaces** commands to display the current status of the output queues.

**Examples**

The following example causes packets for transmission on serial interface 0 to be classified by priority list 1:

```
interface serial 0
 priority-group 1
```

The following example shows how to establish queueing priorities based on the address of the serial link on a STUN connection. Note that you must use the **priority-group** interface configuration command to assign a priority group to an output interface.

```
stun peer-name 131.108.254.6
stun protocol-group 1 sdlc
!
interface serial 0
! Disable the ip address for interface serial 0:
no ip address
! Enable the interface for STUN:
encapsulation stun
!
stun group 2
stun route address 10 tcp 131.108.254.8 local-ack priority
!
! Assign priority group 1 to the input side of interface serial 0:
priority-group 1
! Assign a low priority to priority list 1 on serial link identified
! by group 2 and address A7:
priority-list 1 stun low address 2 A7
```

## Related Commands

| Command | Description |
|---|---|
| **locaddr-priority-list** | Maps LUs to queueing priorities as one of the steps to establishing queueing priorities based on LU addresses. |
| **priority-list default** | Assigns a priority queue for those packets that do not match any other rule in the priority list. |
| **priority-list interface** | Establishes queueing priorities on packets entering from a given interface. |
| **priority-list protocol** | Establishes queueing priorities based on the protocol type. |
| **priority-list protocol ip tcp** | Establishes BSTUN or STUN queueing priorities based on the TCP port. |
| **priority-list protocol stun address** | Establishes STUN queueing priorities based on the address of the serial link. |
| **priority-list queue-limit** | Specifies the maximum number of packets that can be waiting in each of the priority queues. |
| **show interfaces** | Displays statistics for all interfaces configured on the router or access server. |
| **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. |

# priority-list default

To assign a priority queue for those packets that do not match any other rule in the priority list, use the **priority-list default** global configuration command. To return to the default or assign **normal** as the default, use the **no** form of this command.

**priority-list** *list-number* **default** {**high** | **medium** | **normal** | **low**}

**no priority-list** *list-number* **default**

## Syntax Description

| | |
|---|---|
| *list-number* | Any number from 1 to 16 that identifies the priority list. |
| **high** \| **medium** \| **normal** \| **low** | Priority queue level. The **normal** queue is used, if you use the **no** form of this command. |

## Defaults

This command is not enabled by default.

## Command Modes

Global configuration

## Command History

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |

## Usage Guidelines

When you use multiple rules, remember that the system reads the priority settings in order of appearance. When classifying a packet, the system searches the list of rules specified by **priority-list** commands for a matching protocol or interface type. When a match is found, the system assigns the packet to the appropriate queue. The system searches the list in the order specified, and the first matching rule terminates the search.

## Examples

The following example sets the priority queue for those packets that do not match any other rule in the priority list to a low priority:

```
priority-list 1 default low
```

## Related Commands

| Command | Description |
|---|---|
| **priority-group** | Assigns the specified priority list to an interface. |
| **priority-list interface** | Establishes queueing priorities on packets entering from a given interface. |
| **priority-list protocol** | Establishes queueing priorities based on the protocol type. |
| **priority-list queue-limit** | Specifies the maximum number of packets that can be waiting in each of the priority queues. |

| Command | Description |
|---|---|
| **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. |

# priority-list interface

To establish queueing priorities on packets entering from a given interface, use the **priority-list interface** global configuration command. To remove an entry from the list, use the **no** form of this command with the appropriate arguments.

**priority-list** *list-number* **interface** *interface-type interface-number* {**high | medium | normal | low**}

**no priority-list** *list-number* **interface** *interface-type interface-number* {**high | medium | normal | low**}

## Syntax Description

| | |
|---|---|
| *list-number* | Any number from 1 to 16 that identifies the priority list. |
| *interface-type* | The type of the interface. |
| *interface-number* | The number of the interface. |
| **high | medium | normal | low** | Priority queue level. |

## Defaults

No queueing priorities are established by default.

## Command Modes

Global configuration

## Command History

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |

## Usage Guidelines

When you use multiple rules, remember that the system reads the priority settings in order of appearance. When classifying a packet, the system searches the list of rules specified by **priority-list** commands for a matching protocol or interface type. When a match is found, the system assigns the packet to the appropriate queue. The system searches the list in the order specified, and the first matching rule terminates the search.

## Examples

The following example assigns a list entering on serial interface 0 to a medium priority queue level:

```
priority-list 3 interface serial 0 medium
```

**Note** This command defines a rule that determines how packets are attached to an interface. Once the rule is defined, the packet is actually attached to the interface using the **priority-group** command.

## Related Commands

| Command | Description |
|---|---|
| **priority-group** | Assigns the specified priority list to an interface. |
| **priority-list default** | Assigns a priority queue for those packets that do not match any other rule in the priority list. |
| **priority-list protocol** | Establishes queueing priorities based on the protocol type. |
| **priority-list queue-limit** | Specifies the maximum number of packets that can be waiting in each of the priority queues. |
| **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. |

# priority-list protocol

To establish queueing priorities based upon the protocol type, use the **priority-list protocol** global configuration command. To remove a priority list entry assigned by protocol type, use the **no** form of this command with the appropriate arguments.

**priority-list** *list-number* **protocol** *protocol-name* {**high** | **medium** | **normal** | **low**} *queue-keyword keyword-value*

**no priority-list** *list-number* **protocol** [*protocol-name* {**high** | **medium** | **normal** | **low**} *queue-keyword keyword-value*]

## Syntax Description

| | |
|---|---|
| *list-number* | Any number from 1 to 16 that identifies the priority list. |
| *protocol-name* | Protocol type: **aarp**, **apollo**, **appletalk**, **arp**, **bridge** (transparent), **clns**, **clns_es**, **clns_is**, **compressedtcp**, **cmns**, **decnet**, **decnet_node**, **decnet_router-l1**, **decnet_router-l2**, **dlsw**, **ip**, **ipx**, **pad**, **rsrb**, **stun**, **vines**, **xns**, and **x25**. |
| **high** \| **medium** \| **normal** \| **low** | Priority queue level. |
| *queue-keyword keyword-value* | Possible keywords are **fragments**, **gt**, **list**, **lt**, **tcp**, and **udp**. See Table 8. |

## Defaults

No queueing priorities are established.

## Command Modes

Global configuration

## Command History

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |

## Usage Guidelines

When you use multiple rules for a single protocol, remember that the system reads the priority settings in order of appearance. When classifying a packet, the system searches the list of rules specified by **priority-list** commands for a matching protocol type. When a match is found, the system assigns the packet to the appropriate queue. The system searches the list in the order specified, and the first matching rule terminates the search.

The **decnet_router-l1** keyword refers to the multicast address for all level 1 routers, which are intra-area routers, and the **decnet_router-l2** keyword refers to all level 2 routers, which are interarea routers.

The **dlsw**, **rsrb**, and **stun** keywords refer only to direct encapsulation.

Use Table 8, Table 9, and Table 10 to configure the queueing priorities for your system.

*Table 8 Protocol Priority Queue Keywords and Values*

| Option | Description |
|---|---|
| **fragments** | Assigns the priority level defined to fragmented IP packets (for use with IP only). More specifically, this command matches IP packets whose fragment offset field is nonzero. The initial fragment of a fragmented IP packet has a fragment offset of zero, so such packets are not matched by this command.<br><br>**Note** Packets with a nonzero fragment offset do not contain TCP or User Datagram Protocol (UDP) headers, so other instances of this command that use the **tcp** or **udp** keyword will always fail to match such packets. |
| **gt** *byte-count* | Specifies a greater-than count. The priority level assigned goes into effect when a packet size exceeds the value entered for the *byte-count* argument.<br><br>**Note** The size of the packet must also include additional bytes because of MAC encapsulation on the outgoing interface. |
| **list** *list-number* | Assigns traffic priorities according to a specified list when used with AppleTalk, bridging, IP, IPX, VINES, or XNS. The *list-number* argument is the access list number as specified by the **access-list** global configuration command for the specified *protocol-name*. For example, if the protocol is AppleTalk, *list-number* should be a valid AppleTalk access list number. |
| **lt** *byte-count* | Specifies a less-than count. The priority level assigned goes into effect when a packet size is less than the value entered for the *byte-count* argument.<br><br>**Note** The size of the packet must also include additional bytes because of MAC encapsulation on the outgoing interface. |
| **tcp** *port* | Assigns the priority level defined to TCP segments originating from or destined to a specified port (for use with IP only). Table 9 lists common TCP services and their port numbers. |
| **udp** *port* | Assigns the priority level defined to UDP packets originating from or destined to a specified port (for use with IP only). Table 10 lists common UDP services and their port numbers. |

*Table 9 Common TCP Services and Their Port Numbers*

| Service | Port |
|---|---|
| FTP data | 20 |
| FTP | 21 |
| Simple Mail Transfer Protocol (SMTP) | 25 |
| Telnet | 23 |

*Table 10 Common UDP Services and Their Port Numbers*

| Service | Port |
|---|---|
| Domain Name System (DNS) | 53 |
| Network File System (NFS) | 2049 |
| remote-procedure call (RPC) | 111 |
| SNMP | 161 |
| Trivial File Transfer Protocol (TFTP) | 69 |

**Note** Table 9 and Table 10 include some of the more common TCP and UDP port numbers. However, you can specify any port number to be prioritized; you are not limited to those listed.

For some protocols, such as TFTP and FTP, only the initial request uses port 69. Subsequent packets use a randomly chosen port number. For these types of protocols, the use of port numbers fails to be an effective method to manage queued traffic.

## Examples

The following example assigns 1 as the arbitrary priority list number, specifies DECnet as the protocol type, and assigns a high-priority level to the DECnet packets sent on this interface:

```
priority-list 1 protocol decnet high
```

The following example assigns a medium-priority level to every DECnet packet with a size greater than 200 bytes:

```
priority-list 2 protocol decnet medium gt 200
```

The following example assigns a medium-priority level to every DECnet packet with a size less than 200 bytes:

```
priority-list 4 protocol decnet medium lt 200
```

The following example assigns a high-priority level to traffic that matches IP access list 10:

```
priority-list 1 protocol ip high list 10
```

The following example assigns a medium-priority level to Telnet packets:

```
priority-list 4 protocol ip medium tcp 23
```

The following example assigns a medium-priority level to UDP DNS packets:

```
priority-list 4 protocol ip medium udp 53
```

The following example assigns a high-priority level to traffic that matches Ethernet type code access list 201:

```
priority-list 1 protocol bridge high list 201
```

The following example assigns a high-priority level to data-link switching plus (DLSw+) traffic with TCP encapsulation:

```
priority-list 1 protocol ip high tcp 2065
```

The following example assigns a high-priority level to DLSw+ traffic with direct encapsulation:

```
priority-list 1 protocol dlsw high
```

**Note** This command define a rule that determines how packets are attached to an interface. Once the rule is defined, the packet is actually attached to the interface using the **priority-group** command.

## Related Commands

| Command | Description |
|---|---|
| **priority-group** | Assigns the specified priority list to an interface. |
| **priority-list default** | Assigns a priority queue for those packets that do not match any other rule in the priority list. |
| **priority-list interface** | Establishes queueing priorities on packets entering from a given interface. |
| **priority-list queue-limit** | Specifies the maximum number of packets that can be waiting in each of the priority queues. |
| **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. |

# priority-list queue-limit

To specify the maximum number of packets that can be waiting in each of the priority queues, use the **priority-list queue-limit** global configuration command. To select the normal queue, use the **no** form of this command.

**priority-list** *list-number* **queue-limit** [*high-limit* [*medium-limit* [*normal-limit* [*low-limit*]]]]

**no priority-list** *list-number* **queue-limit**

## Syntax Description

| | |
|---|---|
| *list-number* | Any number from 1 to 16 that identifies the priority list. |
| *high-limit*<br>*medium-limit*<br>*normal-limit*<br>*low-limit* | (Optional) Priority queue maximum length. A value of 0 for any of the four arguments means that the queue can be of unlimited size for that particular queue.<br>For default values for these arguments, see Table 11. |

## Defaults

This command is not enabled by default.

## Command Modes

Global configuration

## Command History

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |

## Usage Guidelines

If a priority queue overflows, excess packets are discarded and messages can be sent, if appropriate, for the protocol.

The default queue limit arguments are listed in Table 11.

***Table 11 Default Priority Queue Packet Limits***

| Priority Queue Argument | Packet Limits |
|---|---|
| *high-limit* | 20 |
| *medium-limit* | 40 |
| *normal-limit* | 60 |
| *low-limit* | 80 |

**Note** If priority queueing is enabled and there is an active ISDN (Integrated Services Digital Network) call in the queue, changing the configuration of the **priority-list queue-limit** command drops the call from the queue. For more information about priority queueing, refer to the *Quality of Service Configuration Guide*, Release 12.1.

## Examples

The following example sets the maximum packets in the priority queue to 10:

```
priority-list 2 queue-limit 10 40 60 80
```

## Related Commands

| Command | Description |
|---|---|
| **priority-group** | Assigns the specified priority list to an interface. |
| **priority-list default** | Assigns a priority queue for those packets that do not match any other rule in the priority list. |
| **priority-list interface** | Establishes queueing priorities on packets entering from a given interface. |
| **priority-list protocol** | Establishes queueing priorities based on the protocol type. |
| **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. |

# protect

To configure a virtual circuit (VC) class with protected group or protected VC status for application to a VC bundle member, use the **protect** command in vc-class configuration mode. To remove the protected status from the VC class, use the **no** form of this command.

To configure a specific VC as part of a protected group of the bundle or to configure it as an individually protected VC bundle member, use the **protect** command in bundle-vc configuration mode. To remove the protected status from the VC, use the **no** form of this command.

**protect** {**group** | **vc**}

**no protect** {**group** | **vc**}

## Syntax Description

| | |
|---|---|
| **group** | Configures the VC bundle member as part of the protected group of the bundle. |
| **vc** | Configures the VC member as individually protected. |

## Defaults

The VC neither belongs to the protected group nor is it an individually protected VC.

## Command Modes

VC-class configuration (for a VC class).

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

## Command History

| Release | Modification |
|---|---|
| 12.0(3)T | This command was introduced. |

## Usage Guidelines

Use this command in vc-class configuration mode to configure a VC class to contain protected group or individual protected VC status. When the class is applied to the VC bundle member, that VC is characterized by the protected status. You can also apply this command directly to a VC in bundle-vc configuration mode.

When a protected VC goes down, it takes the bundle down. When all members of a protected group go down, the bundle goes down.

To use this command in vc-class configuration mode, first enter the **vc-class atm** global configuration command.

This command has no effect if the VC class that contains the command is attached to a standalone VC, that is, if the VC is not a bundle member.

To use this command in bundle-vc configuration mode, first enter the **bundle** command to enact bundle configuration mode for the bundle containing the VC member to be configured. Then enter the **pvc-bundle** configuration command to add the VC to the bundle as a member of it.

VCs in a VC bundle are subject to the following configuration inheritance rules (listed in order of next highest precedence):

- VC configuration in bundle-vc mode
- Bundle configuration in bundle mode (with effect of assigned vc-class configuration)
- Subinterface configuration in subinterface mode

## Examples

The following example configures a class called control-class to include a **protect** command, which, when applied to a VC bundle member, configures the VC as an individually protected VC bundle member. When this protected VC goes down, it takes the bundle down.

```
vc-class atm control-class
 protect vc
```

## Related Commands

| Command | Description |
|---|---|
| **bump** | Configures the bumping rules for a VC class that can be assigned to a VC bundle. |
| **class-vc** | Assigns a VC class to an ATM PVC, SVC, or VC bundle member. |
| **precedence (VC bundle)** | Configures precedence levels for a VC class that can be assigned to a VC bundle and thus applied to all VC members of that bundle. |
| **ubr** | Configures UBR QoS and specifies the output peak cell rate for an ATM PVC, SVC, VC class, or VC bundle member. |
| **ubr+** | Configures UBR QoS and specifies the output peak cell rate and output minimum guaranteed cell rate for an ATM PVC, SVC, VC class, or VC bundle member. |
| **vbr-nrt** | Configures the VBR-NRT QoS and specifies output peak cell rate, output sustainable cell rate, and output maximum burst cell size for an ATM PVC, SVC, VC class, or VC bundle member. |

# pvc-bundle

To add a virtual circuit (VC) to a bundle as a member of the bundle and enter bundle-vc configuration mode in order to configure that VC bundle member, use the **pvc-bundle** bundle configuration command. To remove the VC from the bundle, use the **no** form of this command.

**pvc-bundle** *pvc-name* [*vpi*/] [*vci*]

**no pvc-bundle** *pvc-name* [*vpi*/] [*vci*]

## Syntax Description

| | |
|---|---|
| *pvc-name* | The name of the permanent virtual circuit (PVC) bundle. |
| *vpi*/ | (Optional) ATM network virtual path identifier (VPI) for this PVC. The absence of the "/" and a *vpi* value defaults the *vpi* value to 0.<br><br>On the Cisco 7200 and 7500 series routers, the value range is 0 to 255; on the Cisco 4500 and 4700 routers, the value range is 0 to 1 less than the quotient of 8192 divided by the value set by the **atm vc-per-vp** command.<br><br>The *vpi* and *vci* arguments cannot both be set to 0; if one is 0, the other cannot be 0. |
| *vci* | (Optional) ATM network virtual channel identifier (VCI) for this PVC. The value range is 0 to 1 less than the maximum value set for this interface by the **atm vc-per-vp** command. Typically, lower values 0 to 31 are reserved for specific traffic (F4 Operation, Administration, and Maintenance (OAM), SVC signalling, Integrated Local Management Interface (ILMI), and so on) and should not be used.<br><br>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.<br><br>The *vpi* and *vci* arguments cannot both be set to 0; if one is 0, the other cannot be 0. |

## Defaults

None

## Command Modes

Bundle configuration

## Command History

| Release | Modification |
|---|---|
| 12.0(3)T | This command was introduced. |

## Usage Guidelines

Each bundle can contain multiple VCs having different QoS attributes. This command associates a VC with a bundle, making it a member of that bundle. Before you can add a VC to a bundle, the bundle must exist. Use the **bundle** command to create a bundle. You can also use this command to configure a VC that already belongs to a bundle. You enter the command in the same way, giving the name of the VC bundle member.

The **pvc-bundle** command enters into bundle-vc configuration mode, in which you can specify VC-specific and VC class attributes for the VC.

## Examples

The following example specifies an existing bundle named chicago and enters into bundle configuration mode. Then it adds two VCs to the bundle. For each added VC, bundle-vc mode is entered and a VC class is attached to the VC to configure it.

```
bundle chicago
 pvc-bundle chicago-control 207
  class control-class
 pvc-bundle chicago-premium 206
  class premium-class
```

The following example configures the PVC called chicago-control, an existing member of the bundle called chicago, to use class-based weighted fair queueing (CBWFQ). The example configuration attaches the policy map called policy1 to the PVC. Once the policy map is attached, the classes comprising policy1 determine the service policy for the PVC chicago-control.

```
bundle chicago
 pvc-bundle chicago-control 207
  class control-class
  service-policy output policy1
```

## Related Commands

| Command | Description |
|---|---|
| **atm vc-per-vp** | Sets the maximum number of VCIs to support per VPI. |
| **bump** | Configures the bumping rules for a VC class that can be assigned to a VC bundle. |
| **class-bundle** | Configures a VC bundle with the bundle-level commands contained in the specified VC class. |
| **class-vc** | Assigns a VC class to an ATM PVC, SVC, or VC bundle member. |
| **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. |
| **protect** | Configures a VC class with protected group or protected VC status for application to a VC bundle member. |
| **ubr** | Configures UBR QoS and specifies the output peak cell rate for an ATM PVC, SVC, VC class, or VC bundle member. |
| **ubr+** | Configures UBR QoS and specifies the output peak cell rate and output minimum guaranteed cell rate for an ATM PVC, SVC, VC class, or VC bundle member. |
| **vbr-nrt** | Configures the VBR-NRT QoS and specifies output peak cell rate, output sustainable cell rate, and output maximum burst cell size for an ATM PVC, SVC, VC class, or VC bundle member. |