Control Plane Protection

The Control Plane Protection feature is an extension of the policing functionality provided by the existing Control-plane Policing feature. The Control-plane Policing feature allows Quality of Service (QoS) policing of aggregate control-plane traffic destined to the route processor. The Control Plane Protection feature extends this policing functionality by allowing finer policing granularity.

The functionality added with Control Plane Protection includes a traffic classifier, which intercepts traffic and classifies it into three control-plane categories. New port-filtering and queue-thresholding features have also been added. The port-filtering feature provides for policing of packets going to closed or nonlistened TCP/UDP ports, while queue-thresholding limits the number of packets for a specified protocol that will be allowed in the control-plane IP input queue.

History for the Control Plane Protection Feature

Release	Modification
12.4(4)T	This feature was introduced.

Finding Support Information for Platforms and Cisco IOS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS software image support. Access Cisco Feature Navigator at http://www.cisco.com/go/fn. You must have an account on Cisco.com. If you do not have an account or have forgotten your username or password, click **Cancel** at the login dialog box and follow the instructions that appear.

Contents

Corporate Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Prerequisites for Control Plane Protection

- You understand the principles of Control-plane Policing and how to classify control-plane traffic.
- You understand the concepts and general configuration procedure (class map and policy map) for applying QoS policies on a router.

For information about Control-plane Policing and its capabilities, see the *Control-plane Policing* section of the *Cisco IOS Quality of Service Solutions Configuration Guide*, Release 12.4

For information about Cisco IOS QoS and the procedure for configuring QoS in your network using the modular QoS command-line interface (MQC), see the *Cisco IOS Quality of Service Solutions Configuration Guide*, Release 12.4.

Restrictions for Control Plane Protection

Control Plane Protection for IPv4

Control Plane Protection is restricted to IPv4 input path only.

No Support for Direct ACL Configuration

The current release of Control Plane Protection does not support direct access control list (ACL) configuration in the control-plane subinterfaces, but rather can be configured using Modular QoS CLI (MQC) policies.

Requires CEF

Control Plane Protection depends on Cisco Express Forwarding (CEF) for IP packet redirection. If you disable CEF globally, this will remove all active protect and policing policies configured on the control-plane subinterfaces. Aggregate control-plane interface policies will continue to function as normal.

Control-plane Feature Policy Restriction

Policies applicable on the control-plane host subinterface are subject to the following restrictions:

- The port-filter feature policy supports only TCP/UDP-based protocols.
- The queue-thresholding feature policy supports only TCP/UDP-based protocols.

No Support for Distributed or Hardware Switching Platforms

This release does not provide support for distributed or hardware switching platforms.

Control-plane IP Traffic Classification Restrictions

The control-plane host subinterface only supports TCP/UDP-based host traffic. All IP packets entering the control-plane matching any of the following conditions are not classified any further and are redirected to the cef-exception subinterface:

- IP Packets with IP options.
- IP Packets with TTL less than or equal to 1.

Protocols Auto-detected by the Port-filter

Some Cisco IOS TCP/UDP-based services, when configured, may not be auto-detected by the port-filter. That is, they do not get listed under the **show control-plane host open ports** output and they are not classified as an open port. This type of port must be manually added to the active port-filter class-map to be unblocked.

Control-plane Policing Subinterface Restrictions

There are no restrictions on existing aggregate control-plane policing policies. New control-plane policing policies that are configured on host subinterface will not process ARP traffic since ARP traffic is processed at the cef-exception and aggregate interfaces.

Information About Control Plane Protection

To configure the Control-plane Policing feature, you should understand the following concepts:

- Benefits of Control Plane Protection, page 3
- Control Plane Protection Architecture, page 3
- Control-plane Interface and Subinterfaces, page 4

Benefits of Control Plane Protection

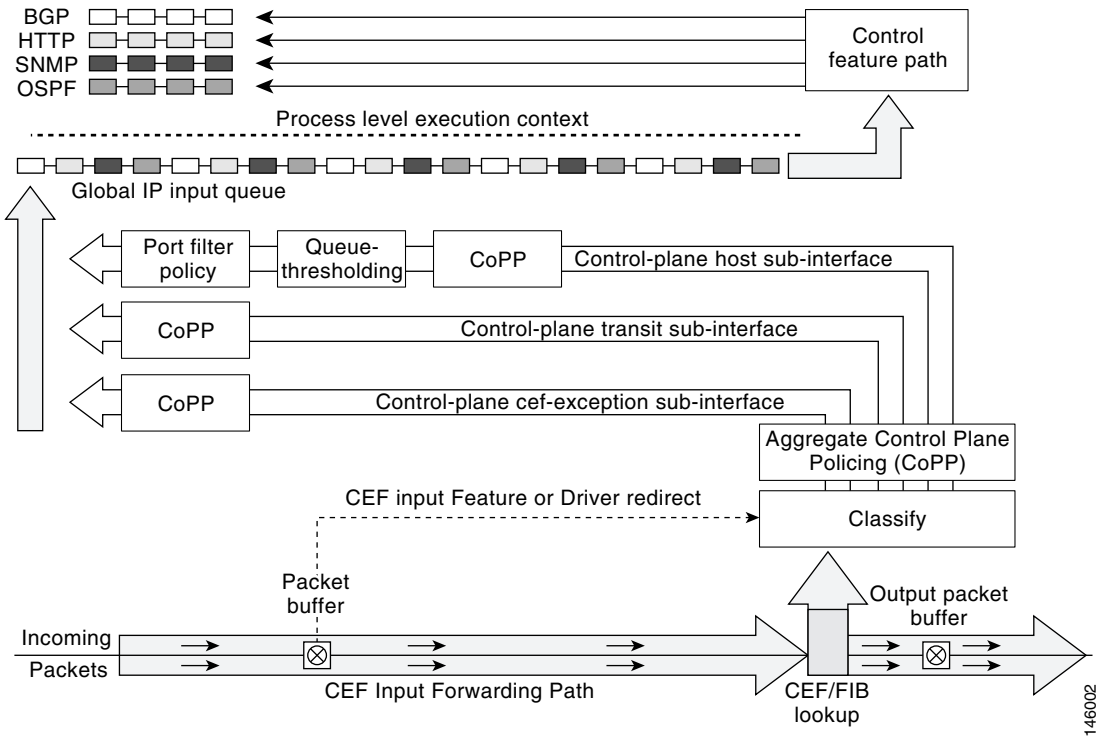
Configuring the Control Plane Protection feature on your Cisco router provides the following benefits:

- Extends protection against DoS attacks at infrastructure routers by providing mechanism for finer policing granularity for control-plane traffic that allows you to rate-limit each type individually.
- Provides a mechanism for early dropping of packets that are directed to closed or nonlistened IOS TCP/UDP ports.
- Provides ability to limit protocol queue usage such that no single protocol flood can overwhelm the input interface.
- Provides QoS control for packets that are destined to the control-plane of Cisco routers.
- Provides ease of configuration for control plane policies using MQC Infrastructure.
- Provides better platform reliability, security and availability.
- Provides dedicated control-plane subinterface for aggregate, host, transit and cef-exception control-plane traffic processing.
- Is highly flexible: permit, deny, rate-limit.
- Provides CPU protection so it can be used for important jobs, such as routing.

Control Plane Protection Architecture

Figure 1 shows control-plane architecture with the Control Plane Protection feature.

Figure 1 Control-plane Architecture with Control Plane Protection



The following sections describe the components of the Control Plane Protections feature.

Control-plane Interface and Subinterfaces

Control Plane Policing (CoPP) introduced the concept of early rate-limiting protocol specific traffic destined to the processor by applying QoS policies to the aggregate control-plane interface. Control Plane Protection extends this control plane functionality by providing three additional control-plane subinterfaces under the top-level (aggregate) control-plane interface. Each subinterface receives and processes a specific type of control-plane traffic. The three subinterfaces are:

- **Control-plane host subinterface**. This interface receives all control-plane IP traffic that is directly destined for one of the router interfaces. Examples of control-plane host IP traffic include tunnel termination traffic, management traffic or routing protocols such as SSH, SNMP, internal BGP (iBGP), and EIGRP. All host traffic terminates on and is processed by the router. Most control plane protection features and policies operate strictly on the control-plane host subinterface. Since most critical router control plane services, such as routing protocols and management traffic, is received on the control-plane host subinterface, it is critical to protect this traffic through policing and protection policies. CoPP, port-filtering and per-protocol queue thresholding protection features can be applied on the control-plane host subinterface.

Note Non-IP based Layer 2 protocol packets such as ARP or CDP do not fall within the control-plane host subinterface. These packets are currently classified in the control-plane CEF-exception subinterface traffic.

- **Control-plane transit subinterface**. This subinterface receives all control-plane IP traffic that is software switched by the route processor. This means packets that are not directly destined to the router itself but rather traffic traversing through the router. Nonterminating tunnels handled by the router is an example of this type of control-plane traffic. Control plane protection allows specific aggregate policing of all traffic received at this subinterface.
- **Control-plane CEF-exception subinterface**. This control-plane subinterface receives all traffic that is either redirected as a result of a configured input feature in the CEF packet forwarding path for process switching or directly enqueued in the control plane input queue by the interface driver (that is, ARP, external BGP (eBGP), OSPF, LDP, Layer2 Keepalives, and all non-IP host traffic). Control plane protection allows specific aggregate policing of this type of control plane traffic.

At process level, packets with time-to-live (TTL) values less than or equal to one are processed before they get to the control plane protection code. Therefore, any policy applied to the control plane protection host path does not affect these packets at interrupt level or at process level. Any policy that needs to be run on OSPF hellos has to be attached to the CEF-exception path or the aggregate path. Also, at process level, the only control plane protection policies that run are the queue-thresholding, port-filtering, and logging policies. CoPP policies run only at interrupt level.

QoS policies attached on any of the control-plane interfaces or subinterfaces execute at interrupt level prior to packets being enqueued to the IP input queue and sent to the processor.

The transit and CEF-exception control plane subinterfaces exist in parallel to the control plane host subinterface. This release of control plane protection allows for rate-limiting policies to be configured on these paths as control plane policing extensions. The port-filtering and per-protocol queue thresholding features are not available on these control-plane subinterfaces.

All protection features in the control plane are implemented as MQC policies that operate using the control plane class-maps and policy-maps. New class-map and policy-map types have been created for the control plane port-filter and per-protocol queue-threshold features.

Port-filtering

The control-plane port-filtering feature enhances control plane protection by providing for early dropping of packets directed toward closed or nonlistened IOS TCP/UDP ports on the router. The port-filter feature policy can be applied only to the control-plane host subinterface.

The port-filter maintains a global database of all open TCP and UDP ports on the router, including random ephemeral ports created by applications. The port database is dynamically populated with entries provided by the registered applications as they start listening on their advertised ports either by configuration of an application (that is SNMP) or initiation of an application (that is, TFTP transfer). An MQC class-map using the list of open ports can be configured and a simple drop policy can be applied to drop all packets destined to closed or nonlistened ports. Port-filter class-maps also support direct match of any user configured TCP/UDP port numbers.

Queue-thresholding

Control-plane protocol queue-thresholding feature provides a mechanism for limiting the number of unprocessed packets a protocol can have at process-level. This feature can only be applied to the control-plane host subinterface. The intent of this feature is to prevent the input queue from being overwhelmed by any single protocol traffic. Per-protocol thresholding follows a protocol charge model. Each protocol's queue usage is limited such that no single mis-behaving protocol process can jam the interface hold queue. In this release, only a subset of TCP/UDP protocols can be configured for thresholding. Non-IP and Layer 2 protocols such as ARP and CDP cannot be configured. You can set queue limits for the following protocols:

- bgp—Border Gateway Protocol
- dns—Domain Name Server lookup
- ftp—File Transfer Protocol
- http—World Wide Web traffic
- igmp— Internet Group Management Protocol
- snmp—Simple Network Management Protocol
- ssh—Secure Shell Protocol
- syslog—Syslog Server
- telnet—Telnet
- tftp—Trivial File Transfer Protocol
- host-protocols—A wild card for all TCP/UDP protocol ports open on the router not specifically matched/configured

Aggregate Control-plane Services

Control-plane Policing is an existing Cisco IOS feature that allows QoS policing of aggregate control-plane traffic destined to the route processor. The Control Plane Protection feature enhances protection for the router's control-plane by providing finer granularity of policing of traffic destined to the router's processor entering through any of the three control-plane subinterfaces. The CoPP feature is intended to be the first Control Plane Protection feature encountered by packets before any other features/policies. Existing (aggregate) control-plane policing policies will not be affected when the control plane protection functionality is enabled. The aggregate control-plane policing policy will be applied on all control-plane traffic types. However, control plane protection allows for additional and/or separate Control-plane policing policies to be configured and applied on the different types of control-plane subinterfaces (host, transit, CEF-exception).

How to Configure Control Plane Protection

The CLI for control-plane (introduced with the Control Plane Policing feature) has been extended to allow for CoPP policies to be applied to individual control-plane subinterfaces (host, transit, CEF-exception). The command syntax for creating CoPP service policies remains the same. In addition, the MQC class-map and policy-map CLI was modified to allow for additional types. The port-filter and queue-threshold policy features available in the host subinterface uses these new class-map and policy-map "types".

CoPP leverages MQC to define traffic classification criteria and to specify configurable policy actions for the classified traffic. Traffic of interest must first be identified via class-maps, which are used to define packets for a particular traffic class. Once classified, enforceable policy actions for the identified traffic are created with policy-maps. The **control-plane** global command allows the control-plane service policies to be attached to the aggregate control-plane interface itself.

The CLI for configuring control-plane policing policies on the new control-plane subinterfaces remains basically the same as the CLI introduced for Control-plane Policing. The only difference is in how you apply or attach the CoPP policy to the different control-plane subinterfaces.

- Defining Packet Classification Criteria for CoPP, page 7 (required)
- Defining a CoPP Service Policy, page 8 (required)

Defining Packet Classification Criteria for CoPP

Perform this task to define the packet classification criteria for CoPP.

Prerequisites

Before you attach an existing QoS policy to the control-plane subinterface, you must first create the policy using MQC to define a class map and policy map for control-plane traffic.

For information about how to classify traffic and create a QoS policy, see the "Modular Quality of Service Command-Line Interface" chapter in the *Cisco IOS Quality of Service Solutions Configuration Guide*.

Restrictions

- The Control-plane Policing feature requires the MQC to configure packet classification and policing. Thus, restrictions that apply to MQC also apply to control-plane policing.
- Only the following classification (match) criteria are supported: standard and extended IP access lists (named or numbered) and the **match ip dscp** command, the **match ip precedence** command, and the **match protocol arp** command.
- The control-plane policing CLI does not support "type" extensions available with other protection features. This is to preserve backward-compatibility.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map** [**match-any** | **match-all**]
4. **match** {**access-group** | **name** *access-group-name*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**class-map** [**match-any** | **match-all**] *class-map-name* **Example:** `Router(config)# class-map match-any control-plane-class`	Enables class map global configuration command mode used to create a traffic class. • **match-any**—Specifies that one of the match criterion must be met for traffic entering the traffic class to be classified as part of the traffic class. • **match-all**—Specifies that all match criterion must be met for traffic entering the traffic class to be classified as part of the traffic class. • *class-map-name*—Specifies the user-defined name of the traffic class. Names can be a maximum of 40 alphanumeric characters.
Step 4	**match** {**access-group** | **name** *access-group-name*} **Example:** `Router(config-cmap)# match access-group name cpp-igp-acl`	Specifies the match criteria for the class-map.

Defining a CoPP Service Policy

To define a service policy, use the **policy-map** global configuration command to specify the service policy name, and use the configuration commands to associate a traffic class that was configured with the **class-map** command, with the QoS action. The traffic class is associated with the service policy when the **class** command is used. You must issue the **class** command after entering policy-map configuration mode. After entering the **class** command, you are automatically in policy-map class configuration mode.

For information about how to classify traffic and create a QoS policy, see the "Modular Quality of Service Command-Line Interface" chapter in the *Cisco IOS Quality of Service Solutions Configuration Guide*.

Restrictions

- Platform-specific restrictions, if any, are checked when the service policy is applied to the control-plane interface.
- The Control-plane Policing feature requires the modular QoS command-line interface (CLI) (MQC) to configure packet classification and policing. Thus, restrictions that apply to MQC also apply to control-plane policing. Also, only two MQC actions are supported in policy maps - police and drop.
- Support for output policing is available only in Cisco IOS Release 12.3(4)T and later T-train releases, and only on the aggregate control-plane interface. Only input policing is available on the new control-plane host, transit and CEF-exception subinterfaces. (Note that output policing does not provide any performance benefits. It simply controls the information that is leaving the device.)

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map** *policy-map-name*
4. **class** *class-name*
5. **police rate** [*burst-normal*] [*burst-max*] **conform-action** *action* **exceed-action** *action* [**violate-action** *action*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**policy-map** *policy-map-name* **Example:** `Router(config)# policy-map control-plane-policy`	Enters policy map configuration mode to define a policy. • ***policy-map-name***—Name of a service policy map. The name can be a maximum of 40 alphanumeric characters.

	Command or Action	Purpose
Step 4	**class** *class-name* **Example:** `Router(config-pmap)# class control-plane-class`	Enters class map configuration mode, which is used to associate a service policy with a class. • *class-name* —Name of a service policy class. The name can be a maximum of 40 alphanumeric characters.

Step 5	`police rate [burst-normal] [burst-max] conform-action action exceed-action action [violate-action action]`
	
	Example:
	`Router(config-pmap-c)# police rate 50000 pps conform-action transmit exceed-action drop`
		To configure traffic policing, use the police command in policy-map class configuration mode or policy-map class police configuration mode.
		
		• **rate**—Specifies the police rate. If the police rate is specified in pps, the valid value range is 1 to 2000000. If the police rate is specified in bps, the valid range of values is 8000 to 10000000000.
		• **pps**—(Optional) Packets per second (pps) will be used to determine the rate at which traffic is policed.
		• **conform-action action**—Action to take on packets that conform to the rate limit.
		• **exceed-action action**—Action to take on packets that exceed the rate limit.

Entering Control Plane Configuration Mode

After you have created a class of traffic and defined the service policy for the control-plane, apply the policy to either the aggregate control-plane interface or one of the subinterfaces.

Restrictions

- Platform-specific restrictions, if any, are checked when the service policy is applied to the control-plane interface.
- Support for output policing is available only in Cisco IOS Release 12.3(4)T and later T-train releases, and only on the aggregate control-plane interface. Only input policing is available on the new control-plane host, transit and CEF-exception subinterfaces. (Note that output policing does not provide any performance benefits. It simply controls the information that is leaving the device.)

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **control-plane** [**host** |**transit** | **cef-exception**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**control-plane** [**host**|**transit**|**cef-exception**] **Example:** `Router(config)# control-plane`	Enters control-plane configuration mode to attach a QoS policy that manages CP traffic to specified control-plane subinterface: • **host**—Enters control-plane host subinterface configuration mode. • **transit**—Enters control-plane transit subinterface configuration mode. • **cef-exception**—Enters control-plane cef-exception subinterface configuration mode.

Applying CoPP Service Policy

Perform this task to apply CoPP service policies to a control-plane interface.

Prerequisites

Before you attach an existing QoS policy to the control-plane, you must first create the policy by using MQC to define a class map and policy map for control-plane traffic.

For information about how to classify traffic and create a QoS policy, see the "Modular Quality of Service Command-Line Interface" chapter in the *Cisco IOS Quality of Service Solutions Configuration Guide*.

Restrictions

- Platform-specific restrictions, if any, are checked when the service policy is applied to the control-plane interface.
- Support for output policing is available only in Cisco IOS Release 12.3(4)T and later T-train releases, and only on the aggregate control-plane interface. Only input policing is available on the new control-plane host, transit and CEF-exception subinterfaces. (Note that output policing does not provide any performance benefits. It simply controls the information that is leaving the device.)

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **control-plane** [**host** | **transit** | **cef-exception**]
4. **service-policy** {**input** | **output**} *policy-map-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**control-plane** [**host** | **transit** | **cef-exception**] **Example:** `Router(config)# control-plane host`	Attaches a QoS policy that manages CP traffic to a specified subinterface, and enters the control-plane configuration mode. • **host**—Applies policies to host control-plane traffic. • **transit**—Applies policies to transit control-plane traffic. • **cef-exception**—Applies policies to CEF-exception control-plane traffic.
Step 4	**service-policy** {**input** | **output**} *policy-map-name* **Example:** `Router(config-cp)# service-policy input control-plane-policy`	Attaches a QoS service policy to the control-plane. • **input**—Applies the specified service policy to packets received on the control-plane. • **output**—Applies the specified service policy to packets transmitted from the control-plane and enables the router to silently discard packets. • *policy-map-name*—Name of a service policy map (created by using the **policy-map** command) to be attached. The name can be a maximum of 40 alphanumeric characters.

Configuring Port-filter Policy

You can apply the port-filter policy feature to the control-plane host subinterface to block traffic destined to closed or nonlistened TCP/UDP ports. New class-map and service-policy types have been created to accommodate the port-filter configuration. The classification and match criteria for the new port-filter

class-maps supports only a constrained subset of the overall global MQC match criteria. Also, the actions supported by the new port-filter service policy is limited as well. that is only the drop action is supported

Restrictions

- The classification and match criteria for the new port-filter class-maps support only a constrained subset of the overall global MQC match criteria.
- The actions supported by the new port-filter service policy is limited. Only the drop action is supported.
- The port-filter feature policy can only be attached on the control-plane host subinterface.
- Some IOS TCP/UDP-based services, when configured, may not be auto-detected by the port filter. That is, they do not get listed under the “show control plane host open ports” output and are not classified as an open port. This type of port must be manually added to the active port filter class-map to be unblocked when using the ‘closed-port’ match criteria.

There are three required steps to configure a port-filter policy:

- Defining Port-filter Packet Classification Criteria, page 13
- Defining Port-filter Service Policy, page 14
- Applying Port-filter Service Policy to the Host Subinterface, page 15.

Defining Port-filter Packet Classification Criteria

Before you can attach a port-filter service policy to the control-plane host subinterface, you must first create the policy using the modified MQC to define a port-filter class-map and policy-map type for control-plane traffic.

A new MQC class-map type called *port-filter* was created for the port-filter feature. You must first create one or more port-filter class-map(s) before you can create your port-filter service policy. Your port-filter class-maps will separate your traffic into “classes” of traffic in which your service policy will define actions on.

Restrictions

- The classification and match criteria for the new port-filter class-maps supports only a constrained subset of the overall global MQC match criteria. That is, only a subset of match protocol criteria is supported.
- Some IOS TCP/UDP-based services, when configured, may not be auto-detected by the port filter. That is, they do not get listed under the "show control plane host open ports" output and are not classified as an open port. This type of port must be manually added to the active port filter class-map to be unblocked when using the ‘closed-port’ match criteria.

SUMMARY STEPS

1. **enable**
2. **class-map type port-filter** [**match-all** | **match-any**] *class-name*
3. **match** {**closed-ports** | **not** | **ports**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**class-map type port-filter** [**match-all** | **match-any**] *class name* **Example:** `Router(config)# class-map type port-filter match-all pf-class`	Creates a class map used to match packets to a specified class and enables the port-filter class-map configuration mode. • **match-all** —Performs a logical AND on the match criteria. • **match-any**—Performs a logical OR on the match criteria. • *class-name*—Name of a service policy class. The name can be a maximum of 40 alphanumeric characters.
Step 3	**match** {**closed-ports** | **not** | **port**} {**TCP** | **UDP**} **Example:** `Router(config-cmap)# match closed-ports`	Specifies the TCP/UDP match criteria for the class-map • **Closed-ports**—Matches automatically on all closed-ports on the router. • **Port**—Allows you to manually specify a TCP/UDP port to match on. • **TCP**—Specifies a TCP port to match on. • **UDP**—Specifies an UDP port to match on.

Defining Port-filter Service Policy

You can define a port-filter service policy that provides additional control-plane protection. Defining this policy supports early dropping of packets that are directed toward closed on nonlistened TCP/UDP ports on the router.

To configure a Port-filter service policy, use the new policy-map type port-filter global configuration command to specify the port-filter service policy name, and use the following configuration commands to associate a port-filter traffic class that was configured with the class-map type port-filter command, with the port-filter drop action command. The port-filter traffic class is associated with the service policy when the class command is used. The class command must be issued after entering policy-map configuration mode. After entering the class command, you are automatically in policy-map class configuration mode.

Restrictions

The actions supported by the new port-filter service policy is limited. Only the drop action is supported.

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **policy-map type port-filter** *policy-map-name*
4. **class** *class-name*
5. **drop**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**policy-map type port-filter** *policy-map-name* **Example:** `Router(config-pcmap)# policy-map type port-filter cppr-pf-policy`	Creates the port-filter service policy and enters the policy-map configuration mode. • *policy-map-name*—Name of a service policy map. The name can be a maximum of 40 alphanumeric characters.
Step 4	**class** *class name* **Example:** `Router (config-cmap)# class pf-class`	Associates a service policy with a class and enters class map configuration mode. • *class-name*—Name of a service policy class. The name can be a maximum of 40 alphanumeric characters.
Step 5	**drop** **Example:** `Router (config-cmap)# drop`	Applies the port-filter service policy action on the class.

Applying Port-filter Service Policy to the Host Subinterface

Perform this task to apply port-filter service policies to a subinterface.

Prerequisites

Before you attach a port-filter service policy to the control-plane host subinterface, you must first create the policy using MQC to define a class map and policy map for the required control-plane traffic.

Restrictions

The port-filter feature can only be applied on the control-plane host subinterface and only as input policy.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **control-plane** [**host** | **transit** | **cef-exception**]]
4. **service-policy type port-filter** {**input**} *port-filter-policy-map-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.
Step 3	**control-plane** [**host** | **transit** | **cef-exception**] **Example:** `Router(config)# control-plane host`	Attaches a QoS policy that manages traffic to the control-plane host subinterface and enters control-plane configuration mode. **Note** Port-filter can only be applied to the host subinterface. • **host**—Enters control-plane host subinterface configuration mode.
Step 4	**service-policy type port-filter** {**input**} *port-filter-policy-map-name* **Example:** `Router(config-cp)# service-policy input cppr-pf-policy`	Attaches a QoS service policy to the control-plane host subinterface. • **input**— Applies the specified service policy to packets received on the control-plane. • **port-filter-policy-map-name**—Name of a port-filter service policy map (created using the policy-map type port-filter command) to be attached. The name can be a maximum of 40 alphanumeric characters.

Examples

The following example shows how to configure a port-filter policy to drop all traffic destined to closed or “nonlistened” TCP/UDP ports:

```
Router(config)# class-map type port-filter pf-class
Router(config-cmap)# match closed-ports
Router(config-cmap)# exit
Router(config)# policy-map type port-filter pf-policy
Router(config-pmap)# class pf-class
Router(config-pmap-c)# drop
Router(config-pmap-c)# end
Router#
```

The following example shows how to configure a port-filter policy to drop all traffic destined to closed or "nonlistened" ports except NTP.

```
Router(config)# class-map type port-filter pf-class
Router(config-cmap)# match not port udp 123
Router(config-cmap)# match closed-ports
Router(config-cmap)# exit
Router(config)# policy-map type port-filter pf-policy
Router(config-pmap)# class pf-class
Router(config-pmap-c)# drop
Router(config-pmap-c)# end
Router#
```

Configuring Queue-threshold Policy

The Control Plane Protection feature includes a new queue-threshold policy feature that can be applied to the control-plane host subinterface. The queue-threshold feature allows you to limit the number of packets for a given higher level protocol allowed in the control-plane IP input queue. Much like the port-filter feature, new class-map and policy-map types have been created to accommodate the queue-threshold feature. As with the port-filter feature, the queue-threshold feature supports a very specific class-map and policy-map capabilities.

Restrictions

- The classification and match criteria for the new queue-threshold class-maps supports only a constrained subset of the overall global MQC match criteria. That is, only a subset of match protocol option.
- The actions supported by the new queue-threshold service policy is limited. Only the queue-limit action is supported.
- The queue-threshold feature is supported only on the control-plane host subinterface as an input policy.

There are three steps required to configure a queue-threshold policy:

- Defining Queue-threshold Packet Classification Criteria, page 17
- Defining a Queue-threshold Service Policy, page 19
- Applying a Queue-threshold policy to the Host Subinterface, page 20

Defining Queue-threshold Packet Classification Criteria

You can define a queue-threshold service policy when you want to limit the number of unprocessed packets that a protocol can have at process level.

Before you can attach a queue-threshold service policy to the control-plane host subinterface, you must first create the policy using the modified MQC to define a queue-threshold class-map and policy-map type for control-plane traffic.

A new MQC class-map type called *queue-threshold* was created for the queue-threshold feature. You must first create one or more queue-threshold class-map(s) before you can create your queue-threshold service policy. Your queue-threshold class-maps will separate your traffic into "classes" of traffic in which your service policy will define actions on.

Restrictions

The classification and match criteria for the new queue-threshold class-map supports only a constrained subset of the overall global MQC match criteria. That is, only a subset of the match protocol criteria is supported.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map type** *queue-threshold* [**match-all | match-any**] *class name*
4. **match protocol** [**bgp | dns | ftp | http | igmp | snmp | ssh | syslog | telnet | tftp**] [**cr**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose
Step 3	**class-map type** *queue-threshold* [**match-all** | **match-any**] *class name* **Example:** `Router(config)#class-map type queue-threshold match-all cppr-pf`	Applies a class map for the queue-threshold and enables the queue-threshold class-map configuration mode. • **match-all**—Performs a logical AND on the match criteria. • **match-any**—Performs a logical OR on the match criteria. • *class-name*—Name of a service policy class. The name can be a maximum of 40 alphanumeric characters.
Step 4	**match protocol** [**bgp** | **dns** | **ftp** | **http** | **igmp** | **snmp** | **ssh** | **syslog** | **telnet** | **tftp** | **host-protocols**] **Example:** `Router(config-cmap)# match protocol bgp`	Specifies the upper layer protocol match criteria for the class-map. • **bgp—**Border Gateway Protocol • **dns**—Domain Name Server lookup • **ftp**—File Transfer Protocol • **http**—World Wide Web traffic • **igmp**—Internet Group Management Protocol • **snmp**—Simple Network Management Protocol • **ssh**—Secure Shell Protocol • **syslog**—Syslog Server • **telne**t—Telnet • **tftp**—Trivial File Transfer Protocol • **host-protocols**—any open TCP/UDP port on the router.

Defining a Queue-threshold Service Policy

To configure a queue-threshold service policy, use the new policy-map type called queue-threshold global configuration command to specify the queue-threshold service policy name, and use the following configuration commands to associate a queue-threshold traffic class that was configured with the class-map type queue-threshold command, with the queue-threshold queue-limit action command. The queue-threshold traffic class is associated with the service policy when the class command is used. The class command must be issued after entering policy-map configuration mode. After entering the class command, you are automatically in policy-map class configuration mode.

Restrictions

The actions supported by the new queue-threshold service policy is limited. Only the queue-limit action is supported.

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **policy-map type queue-threshold** *policy-name*
4. **class** *class name*
5. **queue-limit** *number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.
Step 3	**policy-map type queue-threshold** ***policy name*** **Example:** `Router(config)# policy-map type queue-threshold cppr-qt-policy`	Enables the queue-threshold service policy configuration mode. • *policy-name*—Name of a service policy map. The name can be a maximum of 40 alphanumeric characters.
Step 4	**class** *class name* **Example:** `Router(config-pcmap)# class qt-class`	Enters class map configuration mode used to associate a service policy with a class. • *class-name*—Name of a service policy class. The name can be a maximum of 40 alphanumeric characters.
Step 5	**queue-limit** *number* **Example:** `Router(config-cmap)# queue-limit 75`	Applies the queue-threshold service policy action on the class. **Note** Queue limit range is 0 to 255.

Applying a Queue-threshold policy to the Host Subinterface

Perform this task to apply queue-threshold service policies to the control-plane host subinterface.

Prerequisites

Before you attach a queue-threshold service policy to the control-plane host subinterface, you must first create the policy by using MQC to define a class map and policy map for the required control-plane traffic.

Restrictions

The queue-threshold feature can only be applied on the control-plane host subinterface as an input policy.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **control-plane** [**host** | **transit** | **cef-exception**]
4. **service-policy type queue-threshold** {**input**} *queue-threshold-policy-map-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.
Step 3	**control-plane** [**host** | **transit** | **cef-exception**] **Example:** `Router(config)# control-plane host`	Attaches a QoS queue-threshold policy that manages traffic to the host subinterface and enters control-plane configuration mode. **Note** queue-threshold can only be applied to the host subinterface. • **host**—Enters control-plane host subinterface configuration mode.
Step 4	**service-policy type queue-threshold {input}** *queue-threshold-policy-map-name* **Example:** `Router(config-cp)# service-policy input cppr-qt-policy`	Attaches a QoS service policy to the control-plane. • **input**—Applies the specified service policy to packets received on the control-plane. • *queue-threshold-policy-map-name*—Name of a queue-threshold service policy map (created using the policy-map type queue-threshold command) to be attached. The name can be a maximum of 40 alphanumeric characters.

Examples

The following example shows how to configure a queue-threshold policy to set the queue limit for SNMP protocol traffic to 50, telnet traffic to 50, and all other protocols to 150.

```
Router(config)# class-map type queue-threshold qt-snmp-class
Router(config-cmap)# match protocol snmp
Router(config-cmap)# class-map type queue-threshold qt-telnet-class
Router(config-cmap)# match protocol telnet
Router(config-cmap)# class-map type queue-threshold qt-other-class
Router(config-cmap)# match host-protocols
Router(config-cmap)# exit
Router(config)# policy-map type queue-threshold qt-policy
Router(config-pmap)# class qt-snmp-class
```

```
Router(config-pmap-c)# queue-limit 50
Router(config-pmap-c)# class qt-telnet-class
Router(config-pmap-c)# queue-limit 50
Router(config-pmap-c)# class qt-other-class
Router(config-pmap-c)# queue-limit 150
Router(config-pmap-c)# end
Router#
```

Verifying Control Plane Protection

Use the **show policy-map control-plane** command to verify Control Plane Protection configurations and to view statistics for control-plane service policies.

To display information about the service policy attached to the control-plane, perform the following optional steps.

SUMMARY STEPS

1. **enable**
2. **show policy-map** [**type** *policy-type*] **control-plane** [**pfx | slot** *slot number*] [**all**] [**host | transit | cef-exception**] [{**input | output**} [**class** *class-name*]]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**show policy-map** [**type** *policy-type*] control-plane [**pfx** | **slot** *slot number*] [**all**] [**host** | **transit** | **cef-exception**] [{**input** | **output**} [**class** *class-name*]] **Example:** `Router# show policy-map control-plane all`	Displays information about the control-plane. • **policy-type**—Specifies policy-map type that you want statistics for (i.e., port-filter or queue-threshold) • **pfx**—Does not apply to Control Plane Protection feature. • **slot**—Does not apply to Control Plane Protection feature • **all**—Information for all control plane interfaces. • **host**—Policy-map and class-map statistics for the host path. • **transit**—Policy-map and class-map statistics for transit path. • **cef-exception**—Policy-map and class-map statistics for CEF-exception path. • **input**—Statistics for the attached input policy will be displayed. • **output**—Statistics for the attached output policy will be displayed. • **class** *class name*—Name of class whose configuration and statistics are to be displayed.

Examples

The following example shows that the aggregate CoPP policy map named “copp-transit-policy” is associated with the control-plane transit subinterface and displays the statistics for that policy:

```
Router# show policy-map control-plane transit
 control-plane Transit

  Service-policy input: copp-transit-policy

    Class-map: copp-transit-class (match-all)
      8 packets, 592 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: any
      police:
          rate 2000 pps, burst 488 packets
        conformed 8 packets; actions:
          transmit
        exceeded 0 packets; actions:
          drop
        conformed 0 pps, exceed 0 pps
```

```
    Class-map: class-default (match-any)
      0 packets, 0 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: any
```

The following example shows that the policy map "TEST" is associated with the aggregate control-plane interface. This policy map polices traffic that matches the class map "TEST," while allowing all other traffic (that matches the class map "class-default") to go through as is.

```
Router# show policy-map control-plane

control-plane

Service-policy input:TEST

Class-map:TEST (match-all)
      20 packets, 11280 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match:access-group 101
      police:
        8000 bps, 1500 limit, 1500 extended limit
        conformed 15 packets, 6210 bytes; action:transmit
        exceeded 5 packets, 5070 bytes; action:drop
        violated 0 packets, 0 bytes; action:drop
        conformed 0 bps, exceed 0 bps, violate 0 bps

Class-map:class-default (match-any)
      105325 packets, 11415151 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match:any
```

Additional References

The following sections provide references related to the Control Plane Protection feature.

Related Documents

Related Topic	Document Title
QoS information and configuration tasks	*Cisco IOS Quality of Service Solutions Configuration Guide*
Additional QoS commands	*Cisco IOS Quality of Service Solutions Command Reference*, Release 12.4T

Standards

Standard	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIB	MIBs Link
• CISCO-CLASS-BASED-QOS-MIB **Note** Supported only in Cisco IOS Release 12.3(7)T.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator, found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
None	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Command Reference

This section documents only modified commands.

- **class-map**
- **control-plane**
- **show policy-map control-plane**

class-map

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

class-map [**type** {**stack** | **access-control** | **port-filter** | **queue-threshold**}][**match-all** | **match-any**] *class-map-name*

no class-map [**type** {**stack** | **access-control** | **port-filter** | **queue-threshold**}] [**match-all** | **match-any**] *class-map-name*

Syntax Description

type stack	(Optional) Enables the flexible packet matching (FPM) functionality to determine the correct protocol stack in which to examine. If the appropriate protocol header description files (PHDFs) have been loaded onto the router (via the **load protocol** command), a stack of protocol headers can be defined so the filter can determine which headers are present and in what order.
type access-control	(Optional) Determines the exact pattern to look for in the protocol stack of interest. **Note** You must specify a stack class map (via the **type stack** keywords) before you can specify an access-control class map (via the **type access-control** keywords).
type port-filter	(Optional) Creates a port-filter class-map that enables the TCP/UDP port policing of control plane packets. When enabled it provides filtering of traffic destined to specific ports on the Control Plane host subinterface.
type queue-threshold	(Optional) Enables queue thresholding that limits the total number of packets for a specified protocol that is allowed in the control plane IP input queue. This feature applies only to control plane host subinterface.
match-all | **match-any**	(Optional) Determines how packets are evaluated when multiple match criteria exist. Packets must either meet all of the match criteria (**match-all**) or one of the match criteria (**match-any**) in order to be considered a member of the class.
class-map-name	Name of the class for the class map. The name can be a maximum of 40 alphanumeric characters. 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 or values

Command Modes

Global configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.
12.0(5)XE	This command was integrated into Cisco IOS Release 12.0(5)XE.
12.0(7)S	This command was integrated into Cisco IOS Release 12.0(7)S.
12.1(1)E	This command was integrated into Cisco IOS Release 12.1(1)E.
12.4(4)T	The **type**, **stack**, and **access-control** keywords were added to support FPM. The **type, port-filter** and **queue-threshold** keywords were added to support Control Plane Protection.

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 either the input or output interface (determined by how the **service-policy** command is configured) are checked against the match criteria configured for a class map to determine if the packet belongs to that class.

When configuring a class map, you can use one or more **match** commands to specify match criteria. For example, you can use the **match access-group** command, the **match protocol** command, or the **match input-interface** command. The **match** commands vary according to the Cisco IOS release. For more information about match criteria and **match** commands, see the "Modular Quality of Service Command-Line Interface (CLI)" chapter of the *Cisco IOS Quality of Service Solutions Configuration Guide*.

Examples

The following example specifies class101 as the name of a class, and it defines a class map for this class. The class called class101 specifies policy for traffic that matches access control list 101.

```
class-map class101
  match access-group 101
```

The following example shows how to define FPM traffic classes for slammer and UDP packets. The match criteria defined within the class maps is for slammer and UDP packets with an IP length not to exceed 404 bytes, UDP port 1434, and pattern 0x4011010 at 224 bytes from start of IP header.

```
load protocol disk2:ip.phdf
load protocol disk2:udp.phdf

class-map type stack match-all ip_udp
 description "match UDP over IP packets"
 match field ip protocol eq 0x11 next udp

class-map type access-control match-all slammer
 description "match on slammer packets"
 match field udp dest_port eq 0x59A
 match field ip length eq 0x194
 match start network-start offset 224 size 4 eq 0x4011010
```

The following example shows how to configure a port-filter policy to drop all traffic destined to closed or "nonlistened" ports except SNMP.

```
Router(config)# class-map type port-filter pf-class
Router(config-cmap)# match not port udp 123
Router(config-cmap)# match closed-ports
Router(config-cmap)# exit
Router(config)# policy-map type port-filter pf-policy
```

```
Router(config-pmap)# class pf-class
Router(config-pmap-c)# drop
Router(config-pmap-c)# end
Router#
```

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.
class class-default	Specifies the default class for a service policy map.
match (class-map)	Configures the match critera for class map of the basis of port filter and/or protocol queue policies.
match access-group	Configures the match criteria for a class map on the basis of 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.
match protocol	Configures the match criteria for a class map on the basis of the specified protocol.
policy-map	Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.
service-policy	Attaches a policy map to an input interface or virtual circuit (VC), or an output interface or VC, to be used as the service policy for that interface or VC.

control-plane

To enter control-plane configuration mode and apply a CoPP, port-filter or queue-threshold policy to police traffic destined for the control plane, use the **control-plane** command in global configuration mode. To remove an existing control-plane configuration from the router, use the **no** form of this command.

control-plane [**host | transit | cef-exception**]

no control-plane [**host | transit | cef-exception**]

Syntax Description

host	(Optional) Applies policies to host control plane traffic
transit	(Optional) Applies policies to transit control plane traffic.
cef-exception	(Optional) Applies policies to CEF-exception control plane traffic.

Defaults

No control plane service policies are defined.

Command Modes

Global configuration

Command History

Release	Modification
12.2(18)S	This command was introduced.
12.3(4)T	This command was integrated into Cisco IOS Release 12.3(4)T.
12.0(29)S	This command was integrated into Cisco IOS Release 12.0(29)S.
12.2(18)SXD1	This command was integrated into Cisco IOS Release 12.2(18)SXD1.
12.4(4)T	The **host**, **transit** and **cef-exception** keywords were added.

Usage Guidelines

After you enter the **control-plane** command, you can define aggregate control-plane policing policies for the route processor (RP). You can configure a service policy to police all traffic destined to the control plane:

- From all line cards on the router (aggregate CP services)
- From all interfaces on a line card (distributed CP services)

Aggregate CP services manage traffic destined for the control plane and received on the central switch engine from all line cards in the router.

Distributed CP services manage CP traffic from interfaces on a specified line card before CP packets are forwarded to the central switch engine where aggregate CP services are applied.

Control Plane Policing in this version includes enhanced control plane functionality. It provides a mechanism for early dropping of packets directed toward closed or nonlistened Cisco IOS TCP/UPD ports on the router. It also provides the ability to limit protocol queue usage such that no single misbehaving protocol process can wedge the control plane interface hold queue.

With this enhancement, you can classify control plane traffic into different categories of traffic. These categories are:

- **Control-plane host subinterface**—Subinterface that receives all control-plane IP traffic that is directly destined for one of the router interfaces. Examples of control-plane host IP traffic include tunnel termination traffic, management traffic or routing protocols such as SSH, SNMP, BGP, OSPF, and EIGRP. All host traffic terminates on and is processed by the router. Most control plane protection features and policies operate strictly on the control-plane host subinterface. Since most critical router control plane services, such as routing protocols and management traffic, is received on the control-plane host subinterface, it is critical to protect this traffic through policing and protection policies. CoPP, port-filtering and per-protocol queue thresholding protection features can be applied on the control-plane host subinterface.
- **Control-plane transit subinterface**— Subinterface that receives all control-plane IP traffic that is software switched by the route processor. This means packets not directly destined to the router itself but rather traffic traversing through the router. Non-terminating tunnels handled by the router is an example of this type of control-plane traffic. Control-plane Protection allows specific aggregate policing of all traffic received at this subinterface.
- **Control-plane CEF-exception subinterface**— Subinterface that receives all traffic that is either redirected as a result of a configured input feature in the CEF packet forwarding path for process switching or directly enqueued in the control plane input queue by the interface driver (for example, ARP, L2 Keepalives and all non-IP host traffic). Control-plane Protection allows specific aggregate policing of this specific type of control-plane traffic.

Examples

The following example shows how to configure trusted hosts with source addresses 10.1.1.1 and 10.1.1.2 to forward Telnet packets to the control plane without constraint, while allowing all remaining Telnet packets to be policed at the specified rate. The QoS policy is then applied for aggregate CP services to all packets that are entering the control plane from all line cards in the router.

```
! Allow 10.1.1.1 trusted host traffic.
Router(config)# access-list 140 deny tcp host 10.1.1.1 any eq telnet
! Allow 10.1.1.2 trusted host traffic.
Router(config)# access-list 140 deny tcp host 10.1.1.2 any eq telnet
! Rate limit all other Telnet traffic.
Router(config)# access-list 140 permit tcp any any eq telnet
! Define class-map "telnet-class."
Router(config)# class-map telnet-class
Router(config-cmap)# match access-group 140
Router(config-cmap)# exit
Router(config)# policy-map control-plane-in
Router(config-pmap)# class telnet-class
Router(config-pmap-c)# police 80000 conform transmit exceed drop
Router(config-pmap-c)# exit
Router(config-pmap)# exit
! Define aggregate control plane service for the active Route Processor.
Router(config)# control-plane
Router(config-cp)# service-policy input control-plane-in
Router(config-cp)# exit
```

The next example also shows how to configure trusted hosts with source addresses 10.1.1.1 and 10.1.1.2 to forward Telnet packets to the control plane without constraint, while allowing all remaining Telnet packets that enter through slot 1 to be policed at the specified rate. The QoS policy is applied for distributed CP services to all packets that enter through the interfaces on the line card in slot 1 and are destined for the control plane.

```
! Allow 10.1.1.1 trusted host traffic.
Router(config)# access-list 140 deny tcp host 10.1.1.1 any eq telnet
```

```
! Allow 10.1.1.2 trusted host traffic.
Router(config)# access-list 140 deny tcp host 10.1.1.2 any eq telnet
! Rate limit all other Telnet traffic.
Router(config)# access-list 140 permit tcp any any eq telnet
! Define class-map "telnet-class."
Router(config)# class-map telnet-class
Router(config-cmap)# match access-group 140
Router(config-cmap)# exit
Router(config)# policy-map control-plane-in
Router(config-pmap)# class telnet-class
Router(config-pmap-c)# police 80000 conform transmit exceed drop
Router(config-pmap-c)# exit
Router(config-pmap)# exit
! Define aggregate control plane service for the active Route Processor.
Router(config)# control-plane slot 1
Router(config-cp)# service-policy input control-plane-in
Router(config-cp)# exit
```

The following shows how to apply an aggregate CoPP policy to the host control plane traffic by applying it to the host control plane feature path:

```
Router(config)# control-plane host
Router(config-cp)# service-policy input cpp-policy-host
```

The following shows how to apply an aggregate CoPP policy to the transit control plane traffic by applying it to the control plane transit feature path:

```
Router(config)# control-plane transit
Router(config-cp)# service-policy input cpp-policy-transit
```

The following shows how to apply an aggregate CoPP policy to the CEF-exception control plane traffic by applying it to the control plane CEF-exception feature path:

```
Router(config)# control-plane unidentified
Router(config-cp)# service-policy input cpp-policy-unidentified
```

Related Commands

Command	Description
service-policy (control-plane)	Attaches a policy map to the control plane for aggregate or distributed control-plane services.
show policy-map control-plane	Displays the configuration of a class or all classes for the policy map of a control plane.

show policy-map control-plane

To display the configuration either of a class or of all classes for the policy map of a control-plane, use the **show policy-map control-plane** command in privileged EXEC mode.

show policy-map [**type** *policy-type*] **control-plane** [**pfx | slot** *slot number*] [**all**] [**host | transit | cef-exception**] [{**input | output**} [**class** *class-name*]]

Syntax Description

type *policy-type*	(Optional) Specifies policy-map type for which you want statistics (for example, port-filter or queue-threshold).
pfx	(Optional) Information for all control plane interfaces.
slot *slot number*	(Optional) Policy type and class-map statistics for slot-level aggregate.
all	(Optional) Information for all control plane interfaces.
host	(Optional) Policy type and class-map statistics for the host subinterface.
transit	(Optional) Policy type and class-map statistics for the transit subinterface.
cef-exception	(Optional) Policy type and class-map statistics for the cef-exception subinterface.
input	(Optional) Displays statistics for the attached input policy.
output	(Optional) Displays statistics for the attached output policy.
class *class-name*	(Optional) Name of the class whose configuration and statistics are to be displayed.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(18)S	This command was introduced.
12.3(4)T	This command was integrated into Cisco IOS Release 12.3(4)T, and support for the **output** keyword was added.
12.4(4)T	Support was added for the **type** *policy-type* keyword and argument combination, and the **host**, **transit**, and **cef-exception** keywords.

Usage Guidelines

The **show policy-map control-plane** command displays information for control-plane policing services, which control the number or rate of packets that are going to the process level.

The **type** keyword supports the *policy-type* argument. The options are either port-filter or queue-threshold.

Examples

The following example shows that the policy map called “TEST” is associated with the control plane. This policy map polices traffic that matches the class-map called “TEST,” while allowing all other traffic (that matches the class-map called “class-default”) to go through as is.

```
Router# show policy-map control-plane
```

```
Control Plane

Service-policy input:TEST

Class-map:TEST (match-all)
      20 packets, 11280 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match:access-group 101
      police:
        8000 bps, 1500 limit, 1500 extended limit
        conformed 15 packets, 6210 bytes; action:transmit
        exceeded 5 packets, 5070 bytes; action:drop
        violated 0 packets, 0 bytes; action:drop
        conformed 0 bps, exceed 0 bps, violate 0 bps

Class-map:class-default (match-any)
      105325 packets, 11415151 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match:any
```

Table 1 describes the significant fields shown in the display.

Table 1 show policy-map control-plane Field Descriptions

Field	Description
Fields Associated with Classes or Service Policies	
Service-policy input	Name of the input service policy that is applied to the control plane. (If configured, this field will also show the output service policy.)
Class-map	Class of traffic being displayed. Traffic is displayed for each configured class. The choice for implementing class matches (for example, match-all or match-any) can also appear next to the traffic class.
offered rate	Rate, in kbps, at which packets are coming into the class.
drop rate	Rate, in kbps, at which packets are dropped from the class. The drop rate is calculated by subtracting the number of successfully transmitted packets from the offered rate.
Match	Match criteria for the specified class of traffic. For more information about the variety of match criteria options available, see the chapter "Configuring the Modular Quality of Service Command-Line Interface" in the *Cisco IOS Quality of Service Solutions Configuration Guide*.
Fields Associated with Traffic Policing	
police	Indicates that the police command has been configured to enable traffic policing.
conformed	Displays the action to be taken on packets conforming to a specified rate. Displays the number of packets and bytes on which the action was taken.

Table 1 show policy-map control-plane Field Descriptions (continued)

Field	Description
exceeded	Displays the action to be taken on packets exceeding a specified rate. Displays the number of packets and bytes on which the action was taken.
violated	Displays the action to be taken on packets violating a specified rate. Displays the number of packets and bytes on which the action was taken.

Related Commands

Command	Description
control-plane	Enters control-plane configuration mode, which allows users to associate or modify attributes or parameters that are associated with the control plane of the device.
service-policy (control-plane)	Attaches a policy map to a control plane for aggregate control plane services.