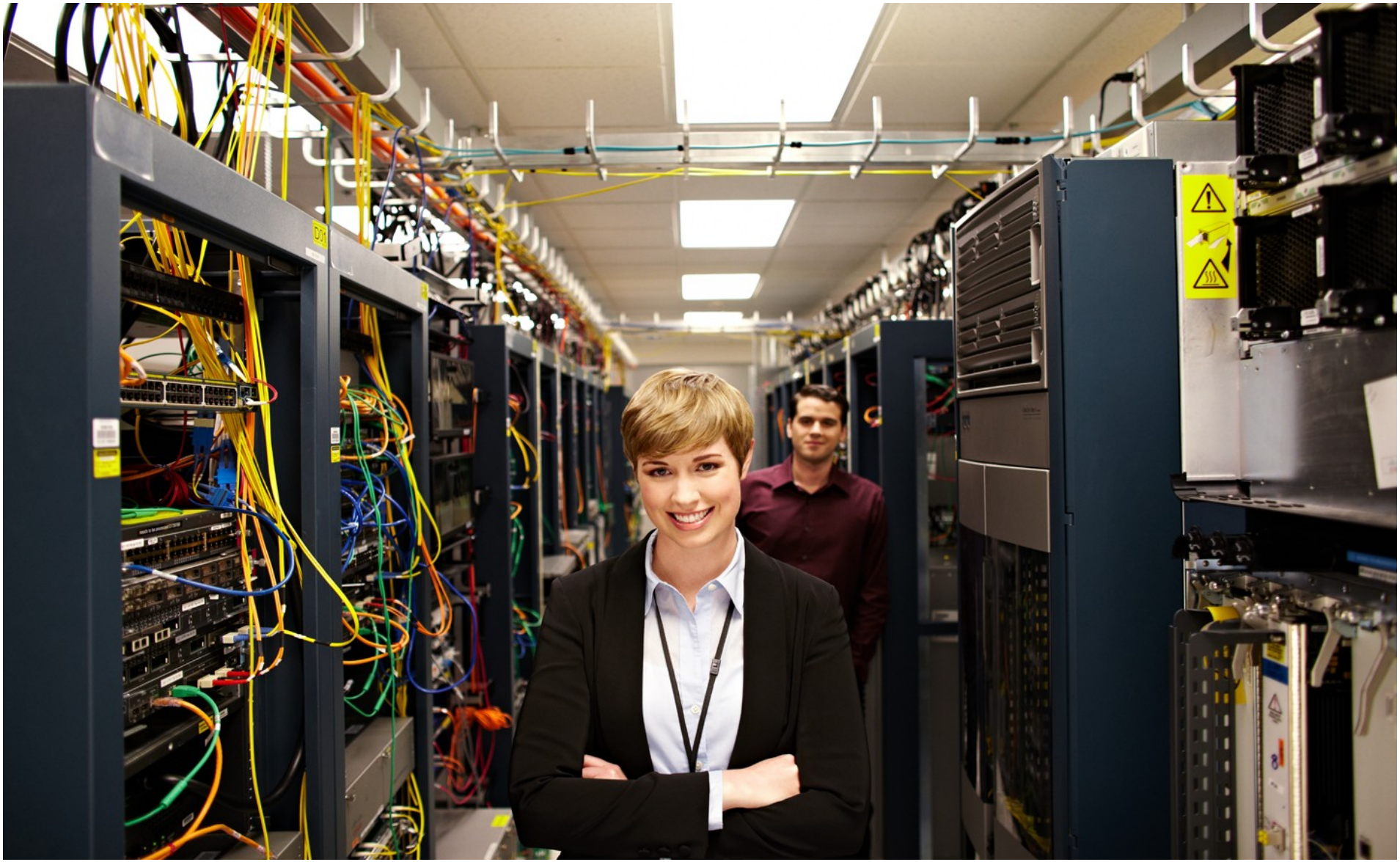
Cisco Control Plane Policing Feature Guide

Cisco Services



Cisco Control Plane Policing Feature Guide

TABLE OF CONTENTS

Contents

Cisco Control Plane Policing Feature Guide

INTRODUCTION

Overview

The performance of the switch is limited by what can be processed in purpose-built hardware application-specific integrated circuits (ASICs) and what can be processed on the switch central processing unit (CPU) by software. Data plane and control plane performance are terms used to describe these performance metrics, respectively. Although the Cisco Catalyst 6500 Supervisor 32 and Supervisor 720 have tremendous capability integrated directly into the hardware, there are specific data types that can only be processed by the switch control plane. Examples of data

that can only be processed by the control plane include routing control protocol, Bridge Protocol data unit (BPDU), Cisco Discovery Protocol, Internet Control Message Protocol (ICMP), or packets with IP options, traffic destined to an IP address of the switch, and management traffic. When too much traffic is redirected to the switch control plane, the CPU can become overwhelmed, resulting in the control plane's inability to perform all required tasks. This condition may not only effect this individual chassis, but other devices within the network. To minimize the effects on control plane performance, a combination of CoPP, hardware rate limiters, and ACLs can be used to reduce the flow of control plane bound traffic, thus keeping the performance of the control plane from being compromised.

Introduction

The switch is typically segmented into three planes of operation, each with a clearly identified objective:

- the data plane allows the ability to forward data packets
- the control plane allows the ability to route data correctly
- the management plane allows the ability to manage network elements.

The vast majority of packets handled by a switch travel through the switch by way of the forwarding plane, or data plane. However, the system's route processor must handle certain packets, such as routing protocols, keepalives, packets destined to the local IP addresses of the switch, and packets from management protocols and other interactive access protocols, such as Telnet and Secure Shell (SSH) Protocol. This type of traffic is often referred to as control plane traffic.

Packet overloads on a switch's control plane can slow down routing processes and, as a result, degrade network service levels and user productivity. One cause for an

overburdened switch control plane is a switch r making inefficient use of shared CPU and memory resources. The same result can occur if reconnaissance or denial-of-service (DoS) attacks appear on the control plane, or if a routing protocol otherwise misbehaves.

For example, if a high volume of rogue packets generated by a virus or worm is presented to the control plane, the switch will spend an excessive amount of time processing and discarding unnecessary traffic. This can eventually overwhelm the route processor, which is responsible for handling switch control plane functions, and possibly bring switch r processes to a halt. Following is an overview of several Cisco IOS Software security features that protect the control plane of networking devices.

- Receive Access Control Lists: Receive Access Controls Lists (rACLs) are designed to protect the route processor on high-end switches from unnecessary traffic that could potentially affect system performance.
- The rACL feature uses standard or extended ACLs that control the traffic sent by the various line cards to the route processor on distributed architectures. An rACL does not apply to transit traffic.
- Control Plane Policing: The control plane policing (CoPP) feature significantly improves upon the rACL feature. Whereas rACLs allow the configuration of basic “permit” and “deny” filters for traffic destined to the switch CPU, the CPP feature extends this by allowing users to configure a quality of service (QoS) filter that can also “rate-limit” this traffic.
- Control Plane Protection: Cisco Control Plane Protection (CPPr) extends the CPP feature by enabling classification of the control plane traffic based on packet destination and information provided by the forwarding plane, allowing appropriate throttling for each category of packet.

  The CoPP feature protects the control plane of Cisco IOS Software-based routers and switches against many attacks, including reconnaissance and denial-of-service (DoS) attacks. In this manner, the control plane can maintain packet forwarding and protocol state despite an attack or heavy load on the router or switch.

Key Benefits

CoPP provides the following benefits:

- Protection against DoS attacks at infrastructure routers and switches
- QoS control for packets that are destined to the

control plane of Cisco routers or switches

- Ease of configuration for control plane policies
- Better platform reliability and availability
- Extends protection against DoS attacks at infrastructure switches 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 switches.
- 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.

Restrictions

- The PFC and DFC provide hardware support for classes that match multicast traffic.
- CoPP is not supported in hardware for broadcast packets. The combination of ACLs, traffic storm control, and CoPP software protection provides protection against broadcast DoS attacks.
- CoPP does not support non-IP classes except for the default non-IP class. ACLs can be used instead of non-IP classes to drop non-IP traffic, and the default non-IP CoPP class can be used to limit to non-IP traffic that reaches the RP CPU.
- Do not use the log keyword in CoPP policy ACLs.
- If you have a large QoS configuration, the system may run out of TCAM space. If this is the case, CoPP may be performed in software.
- When there is a large QoS configuration for other interfaces, you can run out of TCAM space. When this situation occurs, CoPP may be performed entirely in

software and result in performance degradation and CPU cycle consumption.

- You must ensure that the CoPP policy does not filter critical traffic such as routing protocols or interactive access to the switches. Filtering this traffic could prevent remote access to the switch, requiring a console connection.
- The PFC and DFCs support built-in special-case rate limiters, which are useful for situations where an ACL cannot be used (for example, TTL, MTU, and IP options). When you enable the special-case rate limiters, you should be aware that the special-case rate limiters will override the CoPP policy for packets matching the rate-limiter criteria.
- Neither egress CoPP nor silent mode is supported. CoPP is only supported on ingress (service-policy output CoPP cannot be applied to the control plane interface).
- ACE hit counters in hardware are only for ACL logic. You can rely on software ACE hit counters and the show access-list, show policy-map control-plane, and show platform ip qos commands to troubleshoot evaluate CPU traffic.
- CoPP is performed on a per-forwarding-engine basis and software CoPP is performed on an aggregate basis.
- CoPP does not support ACEs with the log keyword.
- CoPP uses hardware QoS TCAM resources. Enter the show tcam utilization command to verify the TCAM utilization.
- To avoid matching the filtering and policing that are configured in a subsequent class, configure policing in each class. CoPP does not apply the filtering in a class that does not contain a police command. A class without a police command matches no traffic.
- The ACLs used for classification are QoS ACLs. The supported QoS ACLs are IP standard, extended, and named.
- These are the only match types supported:
  – ip precedence
  – ip dscp
  – access-group
- Only IP ACLs are supported in hardware.
- MAC-based matching is done in software only.
- You can enter one match command in a single class map only.

- When defining the service policy, the police policy-map action is the only supported action.
- When applying the service policy to the control plane, the input direction is only supported.

Cisco CoPP Planning

CoPP is enabled by default. To disable the default CoPP configuration, enter the no service-policy input policy-default-autocopp control plane configuration mode command.

Cisco CoPP Configuration

CoPP is enabled by default on Catalyst 6500/6800 SUP2T/SUP6T and Catalyst 6880/6840 switches and is based on a preconfigured template. Some class-map configurations do not have corresponding match statements due to the fact that they capture traffic not on the MAC/IP Access Control List (ACL), but rather on internal exceptions that are signalled by the forwarding engine when traffic is received by the switch and a forwarding decision taken.

If a specific class-map needs to be added/modified/

removed from the current CoPP policy, then it must be done from the configuration mode in policy-map mode.

To configure CoPP, perform this task:

	Command or Action	***Purpose***
Step 1	***Router(config)# ip access-list extended access_list_name***	Creates an extended ACL. ***You must configure ACLs in most cases to identify the important or unimportant traffic.***

	Command or Action	*Purpose*
Step 2	***Router(config-ext-nacl)# { permit | deny } protocol source source_wildcard destination destination_wildcard [precedence precedence] [tos tos] [established] [log | log-input] [time-range time_range_name] [fragments]***	Configures filtering in the ACL: • permit sets the conditions under which a packet passes a named IP access list. • deny sets the conditions under which a packet does not pass a named IP access list.
Step 3	Router(config)# **class-map** *traffic_class_name*	Creates a class map.
Step 4	Router(config-cmap)# **match** { **ip precedence** | **ip dscp** | *access_group* }	Configures matching in the class map.

	Command or Action	***Purpose***
Step 5	Router(config)# **policy-map** *service-policy-name*	Defines a service policy map.
Step 6	Router(config-pmap)# **class** *traffic_class_name*	Creates a policy map class.

	Command or Action	*Purpose*
Step 7	Router(config-pmap-c)# **police** *bits_per_second* [*normal_burst_bytes* [*maximum_burst_bytes*]] [**pir** *peak_rate_bps*] [[[**conform-action** *selected_action*] **exceed-action** *selected_action*] **violate-action** *selected_action*] Router(config-pmap-c)# **police rate** *units* **bps** [**burst** *burst_bytes* **bytes**] [peak-rate peak_rate_bps bps] [peak-burst peak_burst_bytes bytes] [**conform-action** *selected_action*] [**exceed-action** *selected_action*] [**violate-action** *selected_action*] Router(config-pmap-c)# **police rate** *units* **pps** [**burst** *burst_packets* **packets**] [**peak-rate** *peak_rate_pps* **pps**] [**peak-burst** *peak_burst_packets* **packets**] [**conform-action** *selected_action*] [**exceed-action** *selected_action*] [**violate-action** *selected_action*] Router(config-pmap-c)# **police flow** [**mask** { **src-only** | **dest-only** | **full-flow** }] *bits_per_second normal_burst_bytes* [[[**conform-action** { **drop** | **set-dscp-transmit** *dscp_value* | **set-prec-transmit** *ip_precedence_value* | **transmit** }] **exceed-action** { **drop** | **policed-dscp** | **transmit** }] **violate-action** { **drop** | **policed-dscp** | **transmit** }]	Configures policing in the service policy map. You can configure any of the following: • Byte-based policing. • Packet-based policing. • Flow-based policing.
Step 8	Router(config)# **control-plane**	Enters the control plane configuration mode.
Step 9	Router(config-cp)# **service-policy input** *service-policy-name*	Applies the QoS service policy to the control plane.

Defining CoPP Traffic Classification

Traffic Classification Overview

You can define any number of classes, but typically traffic is grouped into classes that are based on relative importance. The following provides a sample grouping:

Border Gateway Protocol (BGP)—Traffic that is crucial to maintaining neighbor relationships for BGP routing protocol, for example, BGP keepalives and routing updates. Maintaining BGP routing protocol is crucial to maintaining connectivity within a network or to a service provider. Sites that do not run BGP do not need to use this class.

Interior Gateway Protocol (IGP)—Traffic that is crucial to maintaining IGP routing protocols, for example, open shortest path first OSPF, enhanced interior gateway routing protocol (EIGRP), and routing information protocol (RIP). Maintaining IGP routing protocols is crucial to maintaining connectivity within a network.

Management—Necessary, frequently used traffic that is required during day-to-day operations. For example, traffic used for remote network access, and Cisco IOS image upgrades and management, such as Telnet, secure shell (SSH), network time protocol (NTP), simple network management protocol (SNMP), terminal access controller access control system (TACACS), hypertext transfer protocol (HTTP), trivial file transfer protocol (TFTP), and file transfer protocol (FTP).

Reporting—Traffic used for generating network performance statistics for the purpose of reporting. For example, using Cisco IOS IP service level agreements (SLAs) to generate ICMP with different DSCP settings in order to report on response times within different QoS data classes.

Monitoring—Traffic used for monitoring a switch. Traffic should be permitted but should never be a risk to the switch; with CoPP, this traffic can be permitted but limited to a low rate. For example, ICMP echo request (ping) and traceroute.

Critical Applications—Critical application traffic that is specific and crucial to a particular customer environment. Traffic included in this class should be tailored specifically to the required application requirements of the user (in other words, one customer may use multicast, while another uses

IPsec or generic routing encapsulation (GRE). For example, GRE, hot standby router protocol (HSRP), virtual router redundancy protocol (VRRP), session initiation protocol (SIP), data link switching (DLSw), dynamic host configuration protocol (DHCP), multicast source discovery protocol (MSDP), Internet group management protocol (IGMP), protocol independent multicast (PIM), multicast traffic, and IPsec.

Layer 2 Protocols—Traffic used for address resolution protocol (ARP). Excessive ARP packets can potentially monopolize RP resources, starving other important processes; CoPP can be used to rate limit ARP packets to prevent this situation. ARP is the only Layer 2 protocol that can be specifically classified using the match protocol classification criteria.

Undesirable—Explicitly identifies bad or malicious traffic that should be unconditionally dropped and denied access to the RP. The undesirable classification is particularly useful when known traffic destined for the switch should always be denied and not placed into a default category. If you explicitly deny traffic, then you can enter show commands to collect approximate statistics on the denied traffic and estimate its rate.

Default—All remaining traffic destined for the RP that has not been identified. MQC provides the default class, so the user can specify the treatment to be applied to traffic not explicitly identified in the other user-defined classes. This traffic has a highly reduced rate of access to the RP. With a default classification in place, statistics can be monitored to determine the rate of otherwise unidentified traffic destined for the control plane. After this traffic is identified, further analysis can be performed to classify it and, if needed, the other CoPP policy entries can be updated to accomodate this traffic.

After you have classified the traffic, the ACLs build the classes of traffic that are used to define the policies. For sample basic ACLs for CoPP classification, see the “Sample Basic ACLs for CoPP Traffic Classification” section.

Traffic Classification Restrictions

Before you develop the actual CoPP policy, you must identify and separate the required traffic into different classes. Traffic is grouped into nine classes that are based on relative importance. The actual number of classes needed might differ and should be selected based on your local

requirements and security policies.

You do not have to define policies that match bidirectionally. You only need to identify traffic unidirectionally (from the network to the RP) since the policy is applied on ingress only.

Sample Basic ACLs for CoPP Traffic Classification

This section shows sample basic ACLs for CoPP classification. In the samples, the commonly required traffic is identified with these ACLs:

- ACL 120—Critical traffic
- ACL 121—Important traffic
- ACL 122—Normal traffic
- ACL 123—Explicitly denies unwanted traffic
- ACL 124—All other traffic

This example shows how to define ACL 120 for critical traffic:

```
Router(config)# access-list 120 remark CoPP ACL for critical traffic
```

This example shows how to allow BGP from a known peer to this switch's BGP TCP port:

```
Router(config)# access-list 120 permit tcp host 47.1.1.1 host 10.9.9.9 eq bgp
```

This example shows how to allow BGP from a peer's BGP port to this switch:

```
Router(config)# access-list 120 permit tcp host 47.1.1.1 eq bgp host 10.9.9.9
Router(config)# access-list 120 permit tcp host 10.86.183.120 host 10.9.9.9 eq bgp
Router(config)# access-list 120 permit tcp host 10.86.183.120 eq bgp host 10.9.9.9
```

This example shows how to define ACL 121 for the important class:

```
Router(config)# access-list 121 remark CoPP Important traffic
```

This example shows how to permit return traffic from TACACS host:

```
Router(config)# access-list 121 permit tcp host 1.1.1.1 host 10.9.9.9 established
```

This example shows how to permit SSH access to the switch from a subnet:

```
Router(config)# access-list 121 permit tcp 10.0.0.0 0.0.0.255 host 10.9.9.9 eq 22
```

This example shows how to allow full access for Telnet to the switch from a host in a specific subnet and police the rest of the subnet:

```
Router(config)# access-list 121 deny tcp host 10.86.183.3 any eq telnet
Router(config)# access-list 121 permit tcp 10.86.183.0 0.0.0.255 any eq telnet
```

This example shows how to allow SNMP access from the NMS host to the switch:

```
Router(config)# access-list 121 permit udp host 1.1.1.2 host 10.9.9.9 eq snmp
```

This example shows how to allow the switch to receive NTP packets from a known clock source:

```
Router(config)# access-list 121 permit udp host 1.1.1.3 host 10.9.9.9 eq ntp
```

This example shows how to define ACL 122 for the normal traffic class:

```
Router(config)# access-list 122 remark CoPP normal traffic
```

This example shows how to permit switch-originated traceroute traffic:

```
Router(config)# access-list 122 permit icmp any any ttl-exceeded
Router(config)# access-list 122 permit icmp any any port-unreachable
```

This example shows how to permit receipt of responses to the switch that originated the pings:

```
Router(config)# access-list 122 permit icmp any any echo-reply
```

This example shows how to allow pings to the switch:

```
Router(config)# access-list 122 permit icmp any any echo
```

This example shows how to define ACL 123 for the undesirable class.

```
Router(config)# access-list 123 remark explicitly
```

```
defined "undesirable" traffic
```

In the following example, ACL 123 is a permit entry for classification and monitoring purposes, and traffic is dropped as a result of the CoPP policy.

This example shows how to permit all traffic destined to UDP 1434 for policing:

```
Router(config)# access-list 123 permit udp any any eq
1434
```

This example shows how to define ACL 124 for all other traffic:

```
Router(config)# access-list 124 remark rest of the IP
traffic for CoPP
Router(config)# access-list 124 permit ip any any
```

Monitoring CoPP

You can enter the show policy-map control-plane command for developing site-specific policies, monitoring statistics for the control plane policy, and troubleshooting CoPP. This command displays dynamic information about the actual policy applied, including rate information and the number of bytes (and packets) that conformed or exceeded the configured policies both in hardware and in software.

The output of the show policy-map control-plane command is as follows:

```
Router# show policy-map control-plane
Control Plane Interface
Service policy CoPP-normal
Hardware Counters:
class-map: CoPP-normal (match-all)
Match: access-group 130
police :
96000 bps 3000 limit 3000 extended limit
Earl in slot 3 :
0 bytes
5 minute offered rate 0 bps
aggregate-forwarded 0 bytes action: transmit
exceeded 0 bytes action: drop
aggregate-forward 0 bps exceed 0 bps
Earl in slot 5 :
0 bytes
5 minute offered rate 0 bps
aggregate-forwarded 0 bytes action: transmit
exceeded 0 bytes action: drop
aggregate-forward 0 bps exceed 0 bps

Software Counters:
Class-map: CoPP-normal (match-all) 0 packets, 0 bytes
5 minute offered rate 0 bps, drop rate 0 bps
Match: access-group 130
police:
96000 bps, 3125 limit, 3125 extended limit
conformed 0 packets, 0 bytes; action: transmit
exceeded 0 packets, 0 bytes; action: drop
conformed 0 bps, exceed 0 bps, violate 0 bps
Router#
```

To display the hardware counters for bytes dropped and forwarded by the policy, enter the show platform qos ip command:

```
Router# show platform qos ip
QoS Summary [IP]: (* - shared aggregates, Mod - switch module)
```

```
 Int  Mod  Dir  Class-map  DSCP  Agg  Trust  Fl  AgForward-By
AgPoliced-By
 Id Id
 ------------------------------------------------------
------------------------
 CPP 5 In CoPP-normal 0 1 dscp 0 505408 83822272
 CPP 9 In CoPP-normal 0 4 dscp 0 0 0
 Router#
```

To display the CoPP access list information, enter the show access-lists coppacl-bgp command:

```
 Router# show access-lists coppacl-bgp
 Extended IP access list coppacl-bgp
 10  permit  tcp  host  47.1.1.1  host  10.9.9.9  eq  bgp  (4
matches)
 20 permit tcp host 47.1.1.1 eq bgp host 10.9.9.9
 30  permit  tcp  host  10.86.183.120  host  10.9.9.9  eq  bgp
(1 match)
 40 permit tcp host 10.86.183.120 eq bgp host 10.9.9.9
```

Cisco Control Plane Policing Feature Guide

INTRODUCTION

Troubleshooting

Tip For additional information about Cisco Catalyst 6500 Series Switches (including configuration examples and troubleshooting information), see the documents listed on this page:

http://www.cisco.com/en/US/products/hw/switches/ps708/tsd_products_support_series_home.html

Participate in the Technical Documentation Ideas forum

Resources and Support Information

Obtaining Documentation, Obtaining Support, and Security Guidelines

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service.

TOMORROW
starts here.