



Configuring Denial of Service Protection

This chapter contains information on how to protect your Cisco 7600 series router against Denial of Service (DoS) attacks. The information covered in this chapter is unique to the Cisco 7600 series routers, and it supplements the network security information and procedures in the “[Configuring Network Security](#)” chapter in this publication as well as the network security information and procedures in these publications:

- *Cisco IOS Security Configuration Guide*, Release 12.2, at this URL:
http://www.cisco.com/en/US/docs/ios/12_2/security/configuration/guide/fsecur_c.html
- *Cisco IOS Security Command Reference*, Release 12.2, at this URL:
http://www.cisco.com/en/US/docs/ios/12_2/security/command/reference/fsecur_r.html



Note

For complete syntax and usage information for the commands used in this chapter, refer to these publications:

- The *Cisco IOS Master Command List*, Release 12.2SX at this URL:
http://www.cisco.com/en/US/docs/ios/mcl/122sxmcl/12_2sx_mcl_book.html
- The Release 12.2 publications at this URL:
http://www.cisco.com/en/US/products/sw/iosswrel/ps1835/products_installation_and_configuration_guides_list.html

This chapter consists of these sections:

- [Understanding How DoS Protection Works](#), page 36-2
- [DoS Protection Default Configuration](#), page 36-21
- [DoS Protection Configuration Guidelines and Restrictions](#), page 36-22
- [Understanding How Control Plane Policing Works](#), page 36-28
- [CoPP Default Configuration](#), page 36-28
- [CoPP Configuration Guidelines and Restrictions](#), page 36-28
- [Configuring CoPP](#), page 36-29
- [Monitoring CoPP](#), page 36-31
- [Defining Traffic Classification](#), page 36-32

**Tip**

For additional information (including configuration examples and troubleshooting information), see the documents listed on this page:

http://www.cisco.com/en/US/products/hw/routers/ps368/tsd_products_support_series_home.html

Understanding How DoS Protection Works

The following sections contain an overview of the DoS protection on the Cisco 7600 series router and describe some types of DoS attack scenarios:

- [DoS Protection with a PFC2, page 36-2](#)
- [DoS Protection with a PFC3, page 36-10](#)

DoS Protection with a PFC2

This section contains information about the available methods to counteract DoS attacks with a PFC2 and includes configuration examples. The following sections describe these protection methods:

- [Security ACLs, page 36-2](#)
- [Security ACLs, page 36-2](#)
- [QoS ACLs, page 36-3](#)
- [FIB Rate Limiting, page 36-4](#)
- [ARP Throttling, page 36-5](#)
- [uRPF Check, page 36-5](#)
- [TCP Intercept, page 36-6](#)

Security ACLs

The Cisco 7600 series router can deny DoS packets in hardware using security access control lists (ACLs). Security ACLs are applied in hardware using the TCAM to traffic that can be easily identified using Layer 3 or Layer 4 data. You can apply security ACLs preventively before a DoS attack occurs or after an attack has been identified.

This example shows how a security ACL is used to drop DoS packets:

```
Router# clear mls ip mod 9
Router# show mls ip mod 9
Displaying Netflow entries in module 9
-----
DstIP          SrcIP          Prot:SrcPort:DstPort  Src i/f:AdjPtr
-----
Pkts          Bytes          Age   LastSeen  Attributes
-----
192.168.0.0   192.168.1.0   0    :0        :0        0    : 0
1843
84778
192.168.1.0   192.168.0.0   0    :0        :0        0    : 0
→ 2742416     126151136     2    02:30:17  L3 - Dynamic  <== Note: traffic flow identified
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# no access-list 199
```

```

Router(config)# access-list 199 deny ip host 192.168.0.0 any
Router(config)# access-list 199 permit ip any any
Router(config)# interface g9/1
→ Router(config-if)# ip access 199 in <===== Note: security ACL applied
Router(config-if)# end
Router#
1w6d: %SYS-5-CONFIG_I: Configured from console by console
Router# clear mls ip mod 9
Router# show mls ip mod 9
Displaying Netflow entries in module 9
DstIP          SrcIP          Prot:SrcPort:DstPort  Src i/f:AdjPtr
-----
Pkts           Bytes          Age   LastSeen  Attributes
-----
192.168.0.0    192.168.1.0   0    :0        :0        0    : 0
1542           70932         2    02:31:56  L3 - Dynamic
192.168.1.0    192.168.0.0   0    :0        :0        0    : 0
→ 0            0             2    02:31:56  L3 - Dynamic <===== Note: hardware-forwarded
→ <===== Note: traffic stopped

Extended IP access list 199
  deny ip host 192.168.0.0 any (100 matches)
  permit ip any any
Router# show access-list 199
Extended IP access list 199
→  deny ip host 192.168.0.0 any (103 matches)
  permit ip any any
Router #

```

Security VACLs

Security virtual access lists (VACLs) are security-enforcement tools based on Layer 2, Layer 3, and Layer 4 information. The result of a security VACL lookup against a packet can be a permit, a deny, a permit and capture, or a redirect. When you associate a security VACL with a particular VLAN, all traffic must be permitted by the security VACL before the traffic is allowed into the VLAN. Security VACLs are enforced in hardware, so there is no performance penalty for applying security VACLs to a VLAN on the Cisco 7600 series routers.

QoS ACLs

Unlike security ACLs, QoS ACLs can be used to limit the rate of traffic without denying access to all the traffic in a flow.

This example shows how to use a QoS ACL to prevent a ping attack on a router. A QoS ACL is configured and applied on all interfaces to limit the rate of incoming ICMP echo packets.

```

Router# show ip ospf neighbors

Neighbor ID      Pri  State           Dead Time   Address        Interface
6.6.6.122       1    FULL/BDR       00:00:30   6.6.6.122     Vlan46
Router# show ip eigrp neighbors
IP-EIGRP neighbors for process 200
H   Address          Interface      Hold Uptime   SRTT   RTO   Q   Seq Type
                               (sec)          (ms)          Cnt Num
0   4.4.4.122         V144          11 00:06:07   4     200  0  6555
→ Router# <===== Note: ping attack starts
Router# show proc cpu | include CPU utilization
CPU utilization for five seconds: 99%/90%; one minute: 48%; five minutes: 25%
Router#
2w0d: %OSPF-5-ADJCHG: Process 100, Nbr 6.6.6.122 on Vlan46 from FULL to DOWN, Neighbor
Down: Dead timer expired

```

```

Router# show ip eigrp neighbors
IP-EIGRP neighbors for process 200
Router#
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# access-list 199 permit icmp any any echo
Router(config)# class-map match-any icmp
Router(config-cmap)# match access-group 199
Router(config-cmap)# exit
Router(config)# policy-map icmp
Router(config-pmap)# class icmp
Router(config-pmap-c)# police 96000 16000 16000 conform-action transmit exceed-action drop
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# interface range g4/1 - 9
→ Router(config-if-range)# service-policy input icmp          <===== Note: policy applied
Router(config-if-range)# end
2w0d: %SYS-5-CONFIG_I: Configured from console by console
2w0d: %OSPF-5-ADJCHG: Process 100, Nbr 6.6.6.122 on Vlan46 from LOADING to FULL, Loading
Done
Router# show ip eigrp neighbors
IP-EIGRP neighbors for process 200
H   Address                Interface    Hold Uptime    SRTT   RTO  Q  Seq Type
                               (sec)      (ms)          Cnt  Num
0   4.4.4.122              V144        13 00:00:48    8     200 0  6565
Router#

```

FIB Rate Limiting



Note

The PFC2 CPU rate limiters are off by default.

The forwarding information base (FIB) rate-limiting feature allows all packets that require software processing to be rate limited.

This example shows traffic destined for a nonexistent host address on a locally connected subnet. Normally, the ARP request would result in an ARP reply and the installation of a FIB adjacency for this traffic. However, the adjacency in the FIB for the destination subnet would continue to receive traffic that would be forwarded for software processing. By applying rate-limiting to this traffic, the rate of traffic forwarded for software processing can be limited to a manageable amount.

```

Router# show ip eigrp neighbors
IP-EIGRP neighbors for process 200
H   Address                Interface    Hold Uptime    SRTT   RTO  Q  Seq Type
                               (sec)      (ms)          Cnt  Num
0   4.4.4.122              V144        11 00:00:26    8     200 0  6534
Router# show ip ospf neighbors

Neighbor ID    Pri   State           Dead Time   Address        Interface
6.6.6.122     1    FULL/BDR        00:00:36   6.6.6.122    Vlan46
→Router#                                           <===== Note: attack starts
Router# show arp | include 199.2.250.250
Internet 199.2.250.250    0    Incomplete     ARPA
Router#
1w6d: %OSPF-5-ADJCHG: Process 100, Nbr 6.6.6.122 on Vlan46 from FULL to DOWN, Neighbor Down: Dead
timer expired
Router# show ip eigrp neighbors
IP-EIGRP neighbors for process 200
Router#
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.

```

```

→Router(config)# mls rate-limit unicast cef receive 1000 <===== Note: traffic rate limited to 1000 pps
Router(config)# end
Router#
1w6d: %SYS-5-CONFIG_I: Configured from console by console
Router#
1w6d: %OSPF-5-ADJCHG: Process 100, Nbr 6.6.6.122 on Vlan46 from LOADING to FULL, Loading Done
Router# show ip eigrp neighbors
IP-EIGRP neighbors for process 200
H   Address                Interface    Hold Uptime   SRTT   RTO  Q  Seq Type
   (sec)                    (ms)                Cnt Num
0   4.4.4.122                V144        12 00:00:07   12    200  0  6536
Router#

```

Traffic Storm Control

A traffic storm occurs when packets flood the LAN, which creates excessive traffic and degrades network performance. The traffic storm control feature prevents LAN ports from being disrupted by a broadcast, multicast, or unicast traffic storm on physical interfaces from either mistakes in network configurations or from users issuing a DoS attack. Traffic storm control (also called traffic suppression) monitors incoming traffic levels over a 1-second traffic storm control interval. During the interval, traffic storm control compares the traffic level with the configured traffic storm control level. The traffic storm control level is a percentage of the total available bandwidth of the port. Each port has a single traffic storm control level that is used for all types of traffic (broadcast, multicast, and unicast).

Traffic storm control is configured on an interface and is disabled by default. The configuration example here enables broadcast address storm control on interface FastEthernet 2/3 to a level of 20 percent. When the broadcast traffic exceeds the configured level of 20 percent of the total available bandwidth of the port within a 1-second traffic-storm-control interval, traffic storm control will drop all broadcast traffic until the end of the traffic-storm-control interval.

```
Router(config-if)# storm-control broadcast level 20
```

The Cisco 7600 series router supports broadcast storm control on all LAN ports and multicast and unicast storm control on Gigabit Ethernet ports.

When two or three suppression modes are configured simultaneously, they share the same level settings. If broadcast suppression is enabled, and if multicast suppression is also enabled and configured at a 70-percent threshold, the broadcast suppression will also have a setting for 70 percent.

For more information about configuring traffic storm control see [Chapter 39, “Configuring Traffic Storm Control.”](#)

ARP Throttling

ARP throttling can be used to automatically install hardware-based FIB and adjacency entries to drop packets during ARP resolution. Most of these packets are dropped, but a small number are sent to the MSFC (rate limited).

uRPF Check

When you enable the unicast reverse path forwarding (uRPF) check, packets that lack a verifiable source IP address, such as spoofed IP source addresses, are discarded. Cisco Express Forwarding (CEF) tables are used to verify that the source addresses and the interfaces on which they were received are consistent with the FIB tables on the supervisor engine.

After you enable uRPF check on an interface (per-VLAN basis), the incoming packet is compared to the CEF tables through a reverse lookup. If the packet is received from one of the reverse path routes, the packet is forwarded. If there is no reverse path route on the interface on which the packet was received, the packet fails the uRPF check and is either dropped or forwarded, depending on whether an ACL is applied to the uRPF check fail traffic. If no ACL is specified in the CEF tables, then the forged packets are immediately dropped.

You can only specify an ACL for the uRPF check for packets that fail the uRPF check. The ACL checks whether the packet should immediately be dropped or forwarded. The uRPF check with ACL is not supported in any PFC3 in hardware. Packets that are denied in the uRPF ACL are forwarded in hardware. Packets that are permitted are sent to the CPU.

The uRPF check with a PFC2 is supported in hardware but with only one return path. However, all packets that fail the uRPF check, and are forwarded because of an applied ACL, can be sent and rate limited to the MSFC to generate ICMP unreachable messages; these actions are all software driven. The uRPF check in hardware is supported for routes with up to two return paths (interfaces) and up to six return paths with interface groups configured (two from the FIB table and four from the interface groups)

TCP Intercept

TCP intercept protects recipients of TCP traffic from TCP SYN-flooding DoS attacks. A normal TCP connection starts with a three-way handshake. Host A sends a SYN request to Host B requesting the start of a new TCP session. Host B responds with a SYN ACK acknowledging the receipt of the SYN request. Host A then returns an ACK for Host B's SYN ACK, and the session commences. A SYN-flooding attack occurs when hackers flood servers with requests for connections that have unreachable return addresses. The three-way handshake is never completed, and the connections cannot be established. The amount of session requests to which the server host is responding can overwhelm the server host and prevent legitimate users from connecting to legitimate services, such as web sites and email servers.

TCP intercept prevents the SYN flooding by intercepting and validating TCP requests. TCP intercept supports the following modes:

- **Intercept mode**—The TCP intercept software intercepts TCP synchronization (SYN) packets from clients to servers that match an extended access list. The software establishes a connection with the client on behalf of the destination server, and if successful, establishes the connection with the server on behalf of the client and connects the two half-connections together transparently. Connection attempts from unreachable hosts will never reach the server. The software continues to intercept and forward packets throughout the duration of the connection.

In the case of illegitimate requests from potential hackers, the software's aggressive timeouts on half-open connections and its thresholds on TCP connection requests protect destination servers while still allowing valid requests. When establishing the network security policy using TCP intercept, you can choose to intercept all requests or only those coming from specific networks or destined for specific servers. You also can configure the connection rate and threshold of outstanding connections.

- **Watch mode**—The software passively watches the connection requests flowing through the router. If a connection fails to get established in a configurable interval, the software intervenes and terminates the connection attempt.

Because TCP intercept can operate in either active intercept mode or passive watch mode, it is important to decide which mode is suitable for the network, and to configure your network accordingly. TCP intercept is hardware-assisted on the PFC2 and PFC3 (all types). Configuring many sources and destinations for active intercept mode may overrun the CPU, so it is recommended that only critical servers be protected with active intercept mode.

The default mode of operation is intercept mode. In intercept mode, the software actively intercepts each incoming connection request (SYN) and responds on behalf of the server with the SYN-ACK, and then waits for the ACK from the client. After the preparation is complete, the original SYN is sent to the server, and the software performs the three-way handshake with the server. The two halves are connected together.

In Watch mode, the connection requests pass through the router to the server, but are watched until they become established. If they fail to become established within 30 seconds (this value is configurable), the software sends a reset to the server to clear up its state. Configuring routers for watch mode has less CPU impact than intercept mode. In watch mode, the CPU is not performing checks and connects on both halves of the connection. The CPU is passively monitoring the connection and acting on failed connections after the fact.

TCP intercept is configured globally by first creating the extended access list for the traffic to be intercepted, and then creating the TCP intercept list. The type of traffic to be intercepted must be one of the following:

- All requests
- Only the requests that come from specific networks
- Only the requests that are destined for specific servers

This example defines the source in the access list as any; it does not attempt to filter the source address because it is difficult to know exactly who to intercept packets from. The destination, is specified to protect the destination servers from the TCP SYN-flood attack. If an access list match is not found, traffic is permitted to pass without further action.

```
Router(config)# access-list 101 permit tcp any 10.1.1.1 0.0.0.255
Router(config)# ip tcp intercept list 101
```

Table 36-1 lists the command used to configure the TCP intercept.

Table 36-1 TCP Intercept Configuration

Command	Purpose
Router(config)# access-list <i>access-list-number</i> {deny permit} tcp any <i>destination destination-wildcard</i>	Defines an IP extended access list.
Router(config)# ip tcp intercept list <i>access-list-number</i>	Enables TCP intercept.
Router(config)# ip tcp intercept mode {intercept watch}	Sets the TCP intercept mode.
Router(config)# ip tcp intercept drop-mode {oldest random}	Sets the drop mode.
Router(config)# ip tcp intercept watch-timeout <i>seconds</i>	Changes the time allowed to reach established state; valid values are from 1 to 2147483 seconds.
Router(config)# ip tcp intercept finrst-timeout <i>seconds</i>	Changes the time between receipt of a reset or FIN-exchange and dropping the connection; valid values are from 1 to 2147483 seconds.
Router(config)# ip tcp intercept connection-timeout <i>seconds</i>	Changes the time the software will manage a connection after no activity; valid values are from 1 to 2147483 seconds.

Table 36-1 TCP Intercept Configuration (continued)

Command	Purpose
Router(config)# ip tcp intercept max-incomplete low <i>number</i>	Defines the number of incomplete connections below which the software leaves aggressive mode; valid values are from 1 to 2147483647 connections.
Router(config)# ip tcp intercept max-incomplete high <i>number</i>	Defines the maximum number of incomplete connections allowed before the software enters aggressive mode; valid values are from 1 to 2147483647 connections.
Router(config)# ip tcp intercept one-minute low <i>number</i>	Defines the number of connection requests below which the software leaves aggressive mode; valid values are from 1 to 2147483647 connections.
Router(config)# ip tcp intercept one-minute high <i>number</i>	Defines the number of connection requests received in the last one-minute sample period before the software enters aggressive mode; valid values are from 1 to 2147483647 connections.
Router# show tcp intercept connections	Displays incomplete connections and established connections.
Router# show tcp intercept statistics	Displays TCP intercept statistics.

Hardware-Based Rate Limiters on the PFC2

The PFC2 supports additional hardware-based rate limiters. The PFC2 provides four rate-limiter registers for the new rate limiters, which are configured globally on the router. These rate-limiter registers are present in the Layer 3 forwarding engine (PFC) and are responsible for containing rate-limiting information for result packets that match the various available configured rate limiters.

Because four rate-limiter registers are present on the Layer 3 forwarding engine only, these registers can force different rate-limiting scenarios to share the same register. The registers are assigned on a first-come, first-serve basis. If all registers are being utilized, the only way to configure another rate limiter is to free one register.

The hardware-based rate limiters available on the PFC2 are as follows:

- Ingress and egress ACL bridged packets
- FIB receive and FIB glean cases
- VACL log
- Layer 3 features

Ingress-Egress ACL Bridged Packets (Unicast Only)

This rate limiter rate limits packets sent to the MSFC because of an ingress/egress ACL bridge result. The router accomplishes this by altering existing and new ACL TCAM entries with a TCAM bridge result to a Layer 3 redirect result pointing to the MSFC. Packets hitting the TCAM entries with the altered Layer 3 redirect rate limit result will be rate limited according to the instructions set in CLI by the network administrator. Both the ingress and egress values will be the same, as they both share the same rate-limiter register. If the ACL bridge ingress/egress rate limiting is disabled, the Layer 3 redirect rate limit results are converted to the bridge result.

Ingress or egress ACL-bridged packet cases share a single rate-limiter register. If the feature is turned on, ingress and egress ACLs use the same rate-limiter value.

This example shows how to rate limit the unicast packets from an ingress ACL bridge result to 50000 packets per second, and 50 packets in burst:

```
Router(config)# mls rate-limit unicast acl input 50000 50
```

This example shows how to rate limit the unicast packets from an ingress ACL bridge result to the same rate (50000 pps and 50 packets in burst) for egress ACL bridge results:

```
Router(config)# mls rate-limit unicast acl output 50000 50
```

If the values of the rate limiter are altered on either the ingress or the egress when both are enabled, both values are changed to that new value. In the following example, the output rate is changed to 40000 pps:

```
Router(config)# mls rate-limit unicast acl output 40000 50
```

When you enter the **show mls rate-limit** command, both the ACL bridged in and the ACL bridged out display the new value of 40000 pps:

```
Router# sh mls rate-limit
Load for five secs: 0%/0%; one minute: 0%; five minutes: 0% Time source is NTP,
10:32:15.584 PDT Fri Aug 5 2005
```

Rate Limiter Type	Status	Packets/s
ACL BRIDGE IN	Off	-
ACL BRIDGE OUT	Off	-
L3_SEC_FEATURES	Off	-
VACL LOG	Off	-
FIB RECEIVE	Off	-
FIB GLEAN	Off	-

FIB (CEF) Receive and FIB Glean Cases (Unicast Only)

The FIB receive rate limiter provides the capability to rate limit all packets that contain the MSFC IP address as the destination address. The rate limiters do not discriminate between good frames and bad frames.



Note

Do not enable the FIB receive rate limiter if you are using CoPP. The FIB receive rate limiter overrides the CoPP policies.

This example shows how to rate limit the traffic to 25000 pps with a burst of 60:

```
Router(config)# mls rate-limit unicast cef receive 25000 60
```

The FIB glean rate limiter does not limit ARP traffic, but provides the capability to rate limit traffic that requires address resolution (ARP) and requires that it be sent to the MSFC. This situation occurs when traffic enters a port and contains the destination of a host on a subnet that is locally connected to the MSFC, but no ARP entry exists for that destination host. In this case, because the MAC address of the destination host will not be answered by any host on the directly connected subnet that is unknown, the “glean” adjacency is hit and the traffic is sent directly to the MSFC for ARP resolution. This rate limiter limits the possibility of an attacker overloading the CPU with such ARP requests.

This example shows how to rate limit the rate at which this traffic is sent to the MSFC to 20000 pps and a burst of 60:

```
Router(config)# mls rate-limit unicast cef glean 20000 60
```

VACL Log (Unicast Only)

Packets that are sent to the MSFC because of VLAN-ACL logging can be rate limited to ensure that the CPU is not overwhelmed with logging tasks. VACLs are processed in hardware, but the MSFC does the logging. When VACL logging is configured on the router, IP packets that are denied in the VACL generate log messages.

This example shows how to rate limit logging requests to 5000 pps (the range for this rate limiter is from 10 to 5000 pps):

```
Router(config)# mls rate-limit unicast acl vACL-log 5000
```

Layer 3 Security Features (Unicast Only)

Some security features are processed by first being sent to the MSFC. For these security features, you need to rate limit the number of these packets being sent to the MSFC to reduce any potential overloading. The security features include authentication proxy (auth-proxy), IPSEC, and inspection.

Authentication proxy is used to authenticate inbound or outbound users or both. These users are normally blocked by an access list, but with auth-proxy, the users can bring up a browser to go through the firewall and authenticate on a terminal access controller access control system plus (TACACS+) or RADIUS server (based on the IP address). The server passes additional access list entries down to the router to allow the users through after authentication. These ACLs are stored and processed in software, and if there are many users utilizing auth-proxy, the MSFC may be overwhelmed. Rate limiting would be advantageous in this situation.

IPSec and inspection are also done by the MSFC and may require rate limiting. When the Layer 3 security feature rate limiter is enabled, all Layer 3 rate limiters for auth-proxy, IPSec and inspection are enabled at the same rate.

This example shows how to rate limit the security features to the MSFC to 100000 pps with a burst of 10 packets:

```
Router(config)# mls rate-limit unicast ip features 100000 10
```

DoS Protection with a PFC3

This section contains information about the available methods to counteract DoS attacks with a PFC3 and includes configuration examples. The PFC3 provides a layered defense against DoS attacks using the following methods:

- CPU rate limiters—Controls traffic types.
- Control plane policing (CoPP)—Filters and rate limits control plane traffic. For information about CoPP, see the [“Understanding How Control Plane Policing Works”](#) section on page 36-28.

These sections describe DoS protection with a PFC3:

- [Security ACLs and VACLs, page 36-11](#)
- [QoS Rate Limiting, page 36-12](#)
- [uRPF Check, page 36-12](#)
- [Traffic Storm Control, page 36-13](#)
- [Network Under SYN Attack, page 36-13](#)
- [ARP Policing, page 36-14](#)
- [Recommended Rate-Limiter Configuration, page 36-14](#)

- [Hardware-Based Rate Limiters on the PFC3, page 36-15](#)
 - [Ingress-Egress ACL Bridged Packets \(Unicast Only\), page 36-15](#)
 - [uRPF Check Failure, page 36-16](#)
 - [TTL Failure, page 36-16](#)
 - [ICMP Unreachable \(Unicast Only\), page 36-17](#)
 - [FIB \(CEF\) Receive Cases \(Unicast Only\), page 36-17](#)
 - [FIB Glean \(Unicast Only\), page 36-17](#)
 - [Layer 3 Security Features \(Unicast Only\), page 36-18](#)
 - [ICMP Redirect \(Unicast Only\), page 36-18](#)
 - [VACL Log \(Unicast Only\), page 36-18](#)
 - [MTU Failure, page 36-18](#)
 - [Layer 2 PDU, page 36-19](#)
 - [Layer 2 Protocol Tunneling, page 36-19](#)
 - [IP Errors, page 36-19](#)
 - [Layer 2 Multicast IGMP Snooping, page 36-19](#)
 - [IPv4 Multicast, page 36-19](#)
 - [IPv6 Multicast, page 36-20](#)

Security ACLs and VACLs

If the network is under a DoS attack, ACLs can be an efficient method for dropping the DoS packets before they reach the intended target. Use security ACLs if an attack is detected from a particular host. In this example, the host 10.1.1.10 and all traffic from that host is denied:

```
Router(config)# access-list 101 deny ip host 10.1.1.10 any
Router(config)# access-list 101 permit ip any any
```

Security ACLs also protect against the spoofing of addresses. For example, assume that a source address A is on the inside of a network and a router interface that is pointing to the Internet. You can apply an inbound ACL on the router Internet interface that denies all addresses with a source of A (the inside address). This action stops attacks where the attackers spoof inside source addresses. When the packet arrives at the router interface, it matches on that ACL and drops the packet before it causes damage.

When the Cisco 7600 series router is used with a Cisco Intrusion Detection Module (CIDM), you can dynamically install the security ACL as a response to the detection of the attack by the sensing engine.

VACLs are a security enforcement tool based on Layer 2, Layer 3, and Layer 4 information. The result of a VACL lookup against a packet can be a permit, a deny, a permit and capture, or a redirect. When you associate a VACL with a particular VLAN, all traffic must be permitted by the VACL before the traffic is allowed into the VLAN. VACLs are enforced in hardware, so there is no performance penalty for applying VACLs to a VLAN on the Cisco 7600 series routers.

QoS Rate Limiting

QoS ACLs limit the amount of a particular type of traffic that is processed by the MSFC3. If a DoS attack is initiated against the MSFC, QoS ACLs can prevent the DoS traffic from reaching the MSFC data path and congesting it. The PFC3 performs QoS in hardware, which offers an efficient means of limiting DoS traffic (once that traffic has been identified) to protect the router from impacting the MSFC.

For example, if the network is experiencing ping-of-death or smurf attacks, the administrator should rate limit the ICMP traffic to counteract the DoS attack and still allow legitimate traffic through the processor, or allow it to be forwarded to the MSFC or host. This rate limiting configuration must be done for each flow that should be rate limited and the rate-limiting policy action should be applied to the interface.

In the following example, the access-list 101 permits and identifies ping (echo) ICMP messages from any source to any destination as traffic. Within the policy map, a policing rule defines a specified committed information rate (CIR) and burst value (96000 bps and 16000 bps) to rate limit the ping (ICMP) traffic through the chassis. The policy map then is applied to an interface or VLAN. If the ping traffic exceeds the specified rate on the VLAN or interface where the policy map is applied, it is dropped as specified in the markdown map (the markdown map for the normal burst configurations is not shown in the example).

```
Router(config)# access-list 101 permit icmp any any echo
Router(config)# class-map match-any icmp_class
Router(config-cmap)# match access-group 101
Router(config-cmap)# exit
Router(config)# policy-map icmp_policer
Router(config-pmap)# class icmp_class
Router(config-pmap-c)# police 96000 16000 conform-action transmit exceed-action
policed-dscp-transmit drop
Router(config-pmap-c)# exit
Router(config-pmap)# exit
```

uRPF Check

When you enable the unicast reverse path forwarding (uRPF) check, packets that lack a verifiable source IP address, such as spoofed IP source addresses, are discarded. Cisco Express Forwarding (CEF) tables are used to verify that the source addresses and the interfaces on which they were received are consistent with the FIB tables on the supervisor engine.

After you enable uRPF check on an interface (per-VLAN basis), the incoming packet is compared to the CEF tables through a reverse lookup. If the packet is received from one of the reverse path routes, the packet is forwarded. If there is no reverse path route on the interface on which the packet was received, the packet fails the uRPF check and is either dropped or forwarded, depending on whether an ACL is applied to the uRPF check fail traffic. If no ACL is specified in the CEF tables, then the forged packets are immediately dropped.

You can only specify an ACL for the uRPF check for packets that fail the uRPF check. The ACL checks whether the packet should immediately be dropped or forwarded. The uRPF check with ACL is not supported in any PFC3 in hardware. Packets that are denied in the uRPF ACL are forwarded in hardware. Packets that are permitted are sent to the CPU.

The uRPF check with a PFC3 is supported in hardware; it is also supported in hardware with a PFC2, but with only one return path. However, all packets that fail the uRPF check, and are forwarded because of an applied ACL, can be sent and rate limited to the MSFC to generate ICMP unreachable messages; these actions are all software driven. The uRPF check in hardware is supported for routes with up to two return paths (interfaces) and up to six return paths with interface groups configured (two from the FIB table and four from the interface groups).

Traffic Storm Control

A traffic storm occurs when packets flood the LAN, which creates excessive traffic and degrades network performance. The traffic storm control feature prevents LAN ports from being disrupted by a broadcast, multicast, or unicast traffic storm on physical interfaces from either mistakes in network configurations or from users issuing a DoS attack. Traffic storm control (also called traffic suppression) monitors incoming traffic levels over a 1-second traffic storm control interval. During the interval, traffic storm control compares the traffic level with the configured traffic storm control level. The traffic storm control level is a percentage of the total available bandwidth of the port. Each port has a single traffic storm control level that is used for all types of traffic (broadcast, multicast, and unicast).

Traffic storm control is configured on an interface and is disabled by default. The configuration example here enables broadcast address storm control on interface FastEthernet 2/3 to a level of 20 percent. When the broadcast traffic exceeds the configured level of 20 percent of the total available bandwidth of the port within a 1-second traffic-storm-control interval, traffic storm control will drop all broadcast traffic until the end of the traffic-storm-control interval.

```
Router(config-if)# storm-control broadcast level 20
```

The Cisco 7600 series router supports broadcast storm control on all LAN ports and multicast and unicast storm control on Gigabit Ethernet ports.

When two or three suppression modes are configured simultaneously, they share the same level settings. If broadcast suppression is enabled, and if multicast suppression is also enabled and configured at a 70-percent threshold, the broadcast suppression will also have a setting for 70 percent.

Network Under SYN Attack

A network under a SYN attack is easily recognized. The target host becomes unusually slow, crashes, or suspends operation. Traffic returned from the target host can also cause trouble on the MSFC because return traffic goes to randomized source addresses of the original packets, lacks the locality of “real” IP traffic, and may overflow route caches, or CEF tables.

When the network is under a SYN attack, the TCP intercept feature becomes aggressively defensive. Two factors determine when aggressive behavior on the router begins and ends:

- The total incomplete connections
- Connection requests during the last one-minute sample period

Both factors are configured with low and high values.

If the number of incomplete connections exceed 1,100, or the number of connections arriving in the last one-minute period exceed 1,100, each new arriving connection causes the oldest partial connection (or a random connection) to be deleted. These are the default values, which can be altered. When either of the thresholds is exceeded, the TCP intercept assumes the server is under attack and goes into aggressive mode with the following reactions:

- Each new arriving connection causes the oldest partial (or random partial) to be deleted.
- The initial retransmission timeout is reduced by half to 0.5 seconds, and so the total time trying to establish the connection is cut in half.
- In watch mode, the watch timeout is reduced by half.



Note When both thresholds fall below the configured low value, the aggressive behavior ceases (default value is 900 in both factors). See [Table 36-1](#) for information about TCP intercept configuration.

TCP flows are hardware assisted on both the PFC2 and PFC3 (all PFC3 types).

ARP Policing

During an attack, malicious users may try to overwhelm the MSFC CPU with control packets such as routing protocol or ARP packets. These special control packets can be hardware rate limited using a specific routing protocol and an ARP policing mechanism configurable with the **mls qos protocol** command. The routing protocols supported include RIP, BGP, LDP, OSPF, IS-IS, IGRP, and EIGRP. For example, the command **mls qos protocol arp police 32000** rate limits ARP packets in hardware at 32,000 bps. Although this policing mechanism effectively protects the MSFC CPU against attacks such as line-rate ARP attacks, it does not only police routing protocols and ARP packets to the router but also polices traffic through the box with less granularity than CoPP.

The policing mechanism shares the root configuration with a policing-avoidance mechanism. The policing-avoidance mechanism lets the routing protocol and ARP packets flow through the network when they reach a QoS policer. This mechanism can be configured using the **mls qos protocol protocol pass-through** command.

This example shows how to display the available protocols to use with ARP policing.

```
Router(config)# mls qos protocol ?
  isis
  eigrp
  ldp
  ospf
  rip
  bgp
  ospfv3
  bgpv2
  ripng
  neigh-discover
  wlccp
  arp
```

This example shows how to display the available keywords to use with the **mls qos protocol arp** command:

```
Router(config)# mls qos protocol arp ?
  pass-through  pass-through keyword
  police        police keyword
  precedence    change ip-precedence(used to map the dscp to cos value)
```

Recommended Rate-Limiter Configuration

The recommended rate-limiter configuration is as follows:

- Enable the rate limiters for the traffic types most likely to be used in a DoS attack.
- Do not use a rate limiter on VACL logging unless you configure VACL logging.
- Disable redirects because a platform that supports hardware forwarding, such as the Cisco 7600 series router, reduces the need for redirects.

- Disable unreachable because a platform that supports hardware unreachable, such as the Cisco 7600 series router, reduces the need for unreachable.
- Do not enable the MTU rate limiter if all interfaces have the same MTU.
- When configuring the Layer 2 PDU rate limiter, note the following information:
 - Calculate the expected or possible number of valid PDUs and double or triple the number.
 - PDUs include BPDUs, DTP, VTP, PAgP, LACP, UDLD, etc.
 - Rate limiters do not discriminate between good frames or bad frames.

Hardware-Based Rate Limiters on the PFC3

The PFC3 supports additional hardware-based rate limiters. The PFC3 provides eight rate-limiter registers for the new rate limiters, which are configured globally on the router. These rate-limiter registers are present in the Layer 3 forwarding engine (PFC) and are responsible for containing rate-limiting information for result packets that match the various available configured rate limiters.

Because eight rate-limiter registers are present on the PFC3, these registers can force different rate-limiting scenarios to share the same register. The registers are assigned on a first-come, first-serve basis. If all registers are being utilized, the only way to configure another rate limiter is to free one register.

The hardware-based rate limiters available on the PFC3 are as follows:

- Ingress and egress ACL bridged packets
- uRPF check failures
- FIB receive cases
- FIB glean cases
- Layer 3 security features
- ICMP redirects
- ICMP unreachable (ACL drop)
- No-route (FIB miss)
- VACL log
- TTL failure
- MTU failure
- Multicast IPv4
- Multicast IPv6

Ingress-Egress ACL Bridged Packets (Unicast Only)

This rate limiter rate limits packets sent to the MSFC because of an ingress/egress ACL bridge result. The router accomplishes this by altering existing and new ACL TCAM entries with a TCAM bridge result to a Layer 3 redirect result pointing to the MSFC. Packets hitting the TCAM entries with the altered Layer 3 redirect rate limit result will be rate limited according to the instructions set in CLI by the network administrator. Both the ingress and egress values will be the same, as they both share the same rate-limiter register. If the ACL bridge ingress/egress rate limiting is disabled, the Layer 3 redirect rate limit results are converted to the bridge result.

Ingress or egress ACL-bridged packet cases share a single rate-limiter register. If the feature is turned on, ingress and egress ACLs use the same rate-limiter value.

Burst values regulate how many packets can be allowed in a burst. Each allowed packet consumes a token and a token must be available for a packet to be allowed. One token is generated per millisecond. When packets are not coming in, tokens can be accumulated up to the burst value. For example, if the burst value is set to 50, the router can accumulate up to 50 tokens and absorb a burst of 50 packets.

This example shows how to rate limit the unicast packets from an ingress ACL bridge result to 50000 packets per second, and 50 packets in burst:

```
Router(config)# mls rate-limit unicast acl input 50000 50
```

This example shows how to rate limit the unicast packets from an ingress ACL bridge result to the same rate (50000 pps and 50 packets in burst) for egress ACL bridge results:

```
Router(config)# mls rate-limit unicast acl output 50000 50
```

If the values of the rate limiter are altered on either the ingress or the egress when both are enabled, both values are changed to that new value. In the following example, the output rate is changed to 40000 pps:

```
Router(config)# mls rate-limit unicast acl output 40000 50
```

When you enter the **show mls rate-limit** command, both the ACL bridged in and the ACL bridged out display the new value of 40000 pps:

```
Router# show mls rate-limit
  Rate Limiter Type          Status      Packets/s  Burst
-----
MCAST NON RPF              Off         -           -
MCAST DFLT ADJ             On          100000     100
MCAST DIRECT CON           Off         -           -
ACL BRIDGED IN             On          40000      50
ACL BRIDGED OUT           On          40000      50
IP FEATURES                 Off
...
```

uRPF Check Failure

The uRPF check failure rate limiter allows you to configure a rate for the packets that need to be sent to the MSFC because they failed the uRPF check. The uRPF checks validate that incoming packets on an interface are from a valid source, which minimizes the potential threat of DoS attacks from users using spoofed addresses. When spoofed packets fail the uRPF check, those failures can be sent to the MSFC. The uRPF check rate limiters allow you to rate limit the packets per second that are bridged to the MSFC CPU when a uRPF check failure occurs.

This example shows how to rate limit the uRPF check failure packets sent to the MSFC to 100000 pps with a burst of 100 packets:

```
Router(config)# mls rate-limit unicast ip rpf-failure 100000 100
```

TTL Failure

This rate limiter rate limits packets sent to the MSFC because of a time-to-live (TTL) check failure. As indicated by the **all** keyword in the following example, this rate limiter applies to both multicast and unicast traffic.



Note

The TTL failure rate limiter is not supported for IPv6 multicast.

This example shows how to rate limit the TTL failures to 70000 pps with a burst of 150:

```
Router(config)# mls rate-limit all ttl-failure 70000 150
```


ICMP Unreachable (Unicast Only)

In an ICMP unreachable attack, a device is flooded with a large number of packets that contain a destination address that is unreachable from the flooded device (in this case, the MSFC). The ICMP unreachable rate limiter allows you to rate limit the packets that are sent to the MSFC containing unreachable addresses.

This example shows how to rate limit the packets that are sent to the MSFC because of an ACL drop to 10000 pps and a burst of 100:

```
Router(config)# mls rate-limit unicast ip icmp unreachable acl-drop 10000 100
```

This example shows how to rate limit the packets that require generation of ICMP-unreachable messages because of a FIB miss to 80000 pps and burst to 70:

```
Router(config)# mls rate-limit unicast ip icmp unreachable no-route 80000 70
```

The four rate limiters, ICMP unreachable no route, ICMP unreachable ACL drop, IP errors, and IP RPF failure, share a single rate-limiter register. If any of these limiters are enabled, all of the limiters in this group will share the same value and sometimes the same state (for example, ON/ON/ON). When verifying the rate limiters, if the members of this register are enabled through another feature, an ON-Sharing status (instead of an ON status) is displayed. The exception is the TTL failure rate limiter: its value shares the same value as the other members in the register if you have manually enabled the feature.

FIB (CEF) Receive Cases (Unicast Only)

The FIB receive rate limiter provides the capability to rate limit all packets that contain the MSFC IP address as the destination address. The rate limiters do not discriminate between good frames and bad frames.

**Note**

Do not enable the FIB receive rate limiter if you are using CoPP. The FIB receive rate limiter overrides the CoPP policies.

This example shows how to rate limit the traffic to 25000 pps with a burst of 60:

```
Router(config)# mls rate-limit unicast cef receive 25000 60
```

FIB Glean (Unicast Only)

The FIB glean rate limiter does not limit ARP traffic, but provides the capability to rate limit traffic that requires address resolution (ARP) and requires that it be sent to the MSFC. This situation occurs when traffic enters a port and contains the destination of a host on a subnet that is locally connected to the MSFC, but no ARP entry exists for that destination host. In this case, because the MAC address of the destination host will not be answered by any host on the directly connected subnet that is unknown, the “glean” adjacency is hit and the traffic is sent directly to the MSFC for ARP resolution. This rate limiter limits the possibility of an attacker overloading the CPU with such ARP requests.

This example shows how to rate limit the rate at which this traffic is sent to the MSFC to 20000 pps and a burst of 60:

```
Router(config)# mls rate-limit unicast cef glean 20000 60
```

Layer 3 Security Features (Unicast Only)

Some security features are processed by first being sent to the MSFC. For these security features, you need to rate limit the number of these packets being sent to the MSFC to reduce any potential overloading. The security features include authentication proxy (auth-proxy), IPSEC, and inspection.

Authentication proxy is used to authenticate inbound or outbound users or both. These users are normally blocked by an access list, but with auth-proxy, the users can bring up a browser to go through the firewall and authenticate on a terminal access controller access control system plus (TACACS+) or RADIUS server (based on the IP address). The server passes additional access list entries down to the router to allow the users through after authentication. These ACLs are stored and processed in software, and if there are many users utilizing auth-proxy, the MSFC may be overwhelmed. Rate limiting would be advantageous in this situation.

IPSec and inspection are also done by the MSFC and may require rate limiting. When the Layer 3 security feature rate limiter is enabled, all Layer 3 rate limiters for auth-proxy, IPSec and inspection are enabled at the same rate.

This example shows how to rate limit the security features to the MSFC to 100000 pps with a burst of 10 packets:

```
Router(config)# mls rate-limit unicast ip features 100000 10
```

ICMP Redirect (Unicast Only)

The ICMP-redirect rate limiter allows you to rate limit ICMP traffic. For example, when a host sends packets through a nonoptimal router, the MSFC sends ICMP-redirect messages to the host to correct its sending path. If this traffic occurs continuously, and is not rate limited, the MSFC will continuously generate ICMP-redirect messages.

This example shows how to rate limit the ICMP redirects to 20000 pps, with a burst of 20 packets:

```
Router(config)# mls rate-limit unicast ip icmp redirect 20000 20
```

VACL Log (Unicast Only)

Packets that are sent to the MSFC because of VLAN-ACL logging can be rate limited to ensure that the CPU is not overwhelmed with logging tasks. VACLs are processed in hardware, but the MSFC does the logging. When VACL logging is configured on the router, IP packets that are denied in the VACL generate log messages.

This example shows how to rate limit logging requests to 5000 pps (the range for this rate limiter is from 10 to 5000 pps):

```
Router(config)# mls rate-limit unicast acl vac1-log 5000
```

MTU Failure

Similar to the TTL failure rate limiter, the rate limiter for MTU failures is supported for both unicast and multicast traffic. Packets that fail an MTU check are sent to the MSFC CPU. This might cause the MSFC to be overwhelmed.

This example shows how to rate limit packets failing the MTU failures from being sent to the MSFC to 10000 pps with a burst of 10:

```
Router(config)# mls rate-limit all mtu 10000 10
```

Layer 2 Multicast IGMP Snooping

The IGMP snooping rate limiter limits the number of Layer 2 IGMP packets destined for the supervisor engine. IGMP snooping listens to IGMP messages between the hosts and the supervisor engine. You cannot enable the Layer 2 PDU rate limiter if the Cisco 7600 series router is operating in truncated mode. The router uses truncated mode for traffic between fabric-enabled modules when there are both fabric-enabled and nonfabric-enabled modules installed. In this mode, the router sends a truncated version of the traffic (the first 64 bytes of the frame) over the switch fabric channel.

This example shows how to rate limit IGMP-snooping traffic:

```
Router(config)# mls rate-limit multicast ipv4 igmp 20000 40
```

Layer 2 PDU

The Layer 2 protocol data unit (PDU) rate limiter allows you to limit the number of Layer 2 PDU protocol packets (including BPDUs, DTP, PAgP, CDP, STP, and VTP packets) destined for the supervisor engine and not the MSFC CPU. You cannot enable the Layer 2 PDU rate limiter if the Cisco 7600 series router is operating in truncated mode. The router uses truncated mode for traffic between fabric-enabled modules when there are both fabric-enabled and nonfabric-enabled modules installed. In this mode, the router sends a truncated version of the traffic (the first 64 bytes of the frame) over the switch fabric channel.

This example shows how to rate limit Layer 2 PDUs to 20000 pps with a burst of 20 packets.

```
Router(config)# mls rate-limit layer2 pdu 20000 20
```

Layer 2 Protocol Tunneling

This rate limiter limits the Layer 2 protocol tunneling packets, which include control PDUs, CDP, STP, and VTP packets destined for the supervisor engine. These packets are encapsulated in software (rewriting the destination MAC address in the PDU), and then forwarded to a proprietary multicast address (01-00-0c-cd-cd-d0). You cannot enable the Layer 2 PDU rate limiter if the Cisco 7600 series router is operating in truncated mode. The router uses truncated mode for traffic between fabric-enabled modules when there are both fabric-enabled and nonfabric-enabled modules installed. In this mode, the router sends a truncated version of the traffic (the first 64 bytes of the frame) over the switch fabric channel.

This example shows how to rate limit Layer 2 protocol tunneling packets to 10000 pps with a burst of 10 packets:

```
Router(config)# mls rate-limit layer2 12pt 10000 10
```

IP Errors

This rate limiter limits the packets with IP checksum and length errors. When a packet reaches the PFC3 with an IP checksum error or a length inconsistency error, it must be sent to the MSFC for further processing. An attacker might use the malformed packets to carry out a DoS attack, but the network administrator can configure a rate for these types of packets to protect the control path.

This example shows how to rate limit IP errors sent to the MSFC to 1000 pps with a burst of 20 packets:

```
Router(config)# mls rate-limit unicast ip errors 1000 20
```

IPv4 Multicast

This rate limiter limits the IPv4 multicast packets. The rate limiters can rate limit the packets that are sent from the data path in the hardware up to the data path in the software. The rate limiters protect the control path in the software from congestion and drop the traffic that exceeds the configured rate. Within

the IPv4 multicast rate limiter, there are three rate limiters that you can also configure: the FIB-miss rate limiter, the multicast partially switched flows rate limiter, and the multicast directly connected rate limiter.

The FIB-miss rate limiter allows you to rate limit the multicast traffic that does not match an entry in the mroute table.

The partially switched flow rate limiter allows you to rate limit the flows destined to the MSFC3 for forwarding and replication. For a given multicast traffic flow, if at least one outgoing Layer 3 interface is multilayer switched, and at least one outgoing interface is not multilayer switched (no H-bit set for hardware switching), the particular flow is considered partially switched, or partial-SC (partial shortcut). The outgoing interfaces that have the H-bit flag are switched in hardware and the remaining traffic is switched in software through the MSFC3. For this reason, it may be desirable to rate limit the flow destined to the MSFC3 for forwarding and replication, which might otherwise increase CPU utilization.

The multicast directly connected rate limiter limits the multicast packets from directly connected sources.

This example shows how to rate limit the multicast packets to 30000 pps with a burst of 30:

```
Router(config)# mls rate-limit multicast ipv4 connected 30000 30
```

The **ip-option** keyword and the ip-option rate limiter are supported in PFC3B or PFC3BXL mode only.

This example shows how to set the rate limiters for the IPv4 multicast packets failing the uRPF check:

```
Router(config)# mls rate-limit multicast ipv4 non-rpf 100
```

This example shows how to rate limit the multicast FIB miss packets to 10000 pps with a burst of 10:

```
Router(config)# mls rate-limit multicast ipv4 fib-miss 10000 10
```

This example shows how to rate limit the partial shortcut flows to 20000 pps with a burst of 20 packets:

```
Router(config)# mls rate-limit multicast ipv4 partial 20000 20
```

This example shows how to rate limit the multicast packets to 30000 pps with a burst of 20:

```
Router(config)# mls rate-limit multicast ipv4 connected 30000 20
```

This example shows how to rate limit IGMP-snooping traffic:

```
Router(config)# mls rate-limit multicast ipv4 igmp 20000 40
```

IPv6 Multicast

This rate limiter limits the IPv6 multicast packets. [Table 36-2](#) lists the IPv6 rate limiters and the class of traffic that each rate limiter serves.

Table 36-2 IPv6 Rate Limiters

Rate Limiter	Traffic Classes to be Rate Limited
Connected	Directly connected source traffic
Default-drop	* (*, G/m) SSM * (*, G/m) SSM non-rpf
Route-control	* (*, FF02::X/128)

Table 36-2 IPv6 Rate Limiters (continued)

Rate Limiter	Traffic Classes to be Rate Limited
Starg-bridge	* (*, G/128) SM * SM non-rpf traffic when (*, G) exists
Starg-M-bridge	* (*, G/m) SM * (*, FF/8) * SM non-rpf traffic when (*, G) doesn't exist

You can configure rate limiters for IPv6 multicast traffic using one of the following methods:

- Direct association of the rate limiters for a traffic class—Select a rate and associate the rate with a rate limiter. This example shows how to pick a rate of 1000 pps and 20 packets per burst and associate the rate with the **default-drop** rate limiter:

```
Router(config)# mls rate-limit multicast ipv6 default-drop 1000 20
```

- Static sharing of a rate limiter with another preconfigured rate limiter—When there are not enough adjacency-based rate limiters available, you can share a rate limiter with an already configured rate limiter (target rate limiter). This example shows how to share the **route-cntl** rate limiter with the **default-drop** target rate limiter:

```
Router(config)# mls rate-limit multicast ipv6 route-cntl share default-drop
```

If the target rate limiter is not configured, a message is displayed that indicates that the target rate limiter must be configured for it to be shared with other rate limiters.

- Dynamic sharing of rate limiters—If you are not sure about which rate limiter to share with, use the **share auto** keywords to enable dynamic sharing. When you enable dynamic sharing, the system selects a preconfigured rate limiter and shares the given rate limiter with the preconfigured rate limiter. This example shows how to choose dynamic sharing for the route-cntl rate limiter:

```
Router(config)# mls rate-limit multicast ipv6 route-cntl share auto
```

This example shows how to set the rate limiters for the IPv6 multicast packets from a directly connected source:

```
Router(config)# mls rate-limit multicast ipv6 connected 1500 20
```

This example shows how to configure a direct association of the rate limiters for a traffic class:

```
Router(config)# mls rate-limit multicast ipv6 default-drop 1000 20
```

This example shows how to configure the static sharing of a rate limiter with another preconfigured rate limiter:

```
Router(config)# mls rate-limit multicast ipv6 route-cntl share default-drop
```

This example shows how to enable dynamic sharing for the route control rate limiter:

```
Router(config)# mls rate-limit multicast ipv6 route-cntl share auto
```

DoS Protection Default Configuration

Table 36-3 shows the DoS protection default configuration for the PFC3 hardware-based rate limiters.

Table 36-3 PFC3 Hardware-based Rate Limiter Default Setting

Rate Limiter	Default Status (ON/OFF)	Default Value
Ingress/Egress ACL Bridged Packets	OFF	
RPF Failures	ON	100 pps, burst of 10 packets
FIB Receive cases	OFF	
FIB Glean Cases	OFF	
Layer 3 Security features	OFF	
ICMP Redirect	OFF	
ICMP Unreachable	ON	100 pps, burst of 10 packets
VACL Log	ON	2000 pps, burst of 10 packets
TTL Failure	OFF	
MTU Failure	OFF	
Layer 2 PDU	OFF	
Layer 2 Protocol Tunneling	OFF	
IP Errors	ON	100 pps, burst of 10 packets
Multicast IGMP	OFF	
Multicast FIB-Miss	ON	100000 pps, burst of 100 packets
Multicast Partial-SC	ON	100000 pps, burst of 100 packets
Multicast Directly Connected	OFF	
Multicast Non-RPF	OFF	
Multicast IPv6	ON	If the <i>packets-in-burst</i> is not set, a default of 100 is programmed for multicast cases.

DoS Protection Configuration Guidelines and Restrictions

The section contains these configuration guidelines and restrictions:

- [PFC2, page 36-22](#)
- [PFC3, page 36-23](#)

PFC2

When configuring DoS protection on systems configured with a PFC2, follow these guidelines and restrictions:

- When using security ACLs to drop DoS packets, note the following information:
 - The security ACL must specify the traffic flow to be dropped.
 - Security ACLs need to be configured on all external interfaces that require protection. Use the **interface range** command to configure a security ACL on multiple interfaces.

- When using QoS ACLs to limit the rate of packets, note the following information:
 - The QoS ACL must specify the traffic flow to be rate limited.
 - When adding a QoS ACL to limit the rate of packets to an interface that already has a QoS ACL configured, you can perform one of the following:
 - * Merge the rate-limiting ACL with the existing QoS ACL.
 - * Define a separate class that matches the DoS ACL and tie the class to the policy map.
 - QoS ACLs need to be configured on all external interfaces that require protection. Use the interface range command to configure an ACL on multiple interfaces.
- The CPU rate limiters limit the traffic in aggregate only and do not distinguish between good and bad packets.
- The following FIB rate-limiting usage guidelines apply:
 - FIB rate limiting does not limit the rate of broadcast or some multicast traffic in hardware. The PFC3 has separate multicast rate limiters. The Supervisor Engine 2 does not have separate multicast rate limiters.
 - FIB rate limiting does not differentiate between legitimate and illegitimate traffic (for example, tunnels, Telnet).
 - FIB rate limiting applies aggregate rate limiting and not per-flow rate limiting.

PFC3

When configuring DoS protection on systems configured with a PFC3, follow these CPU rate limiter guidelines and restrictions:



Note

For the CoPP guidelines and restrictions, see the [“CoPP Configuration Guidelines and Restrictions” section on page 36-28](#).

- Do not use these rate limiters if multicast is enabled in systems configured with a PFC3A:
 - TTL failure
 - MTU failure
- These rate limiters are supported only in PFC3B or PFC3BXL mode:
 - Unicast IP options
 - Multicast IP options
- These are Layer 2 rate limiters:
 - Layer 2 PDUs
 - Layer 2 protocol tunneling
 - Layer 2 Multicast IGMP
- There are eight Layer 3 registers and two Layer 2 registers that can be used as CPU rate limiters.
- Do not use the CEF receive limiter if CoPP is being used. The CEF receive limiter will override the CoPP traffic.
- Rate limiters override the CoPP traffic.

- Configured rate limits is applied to each forwarding engine (except for the Layer 2 hardware rate limiter which is applied globally).
- Layer 2 rate limiters are not supported in truncated mode.
- The following restrictions apply when using the ingress and egress ACL-bridged packet rate limiters:
 - The ingress and egress ACL-bridged packet rate limiter is available for unicast traffic only.
 - The ingress and egress ACL-bridged packet rate limiters share a single rate-limiter register. If you enable the ACL-bridge ingress and egress rate limiters, both the ingress and the egress ACLs must share the same rate-limiter value.
- Use the **mls rate-limit unicast** command to rate limit unicast traffic.
- Use the **mls rate-limit multicast** command to rate limit multicast traffic.
- Use the **mls rate-limit multicast layer 2** command to rate limit Layer 2 multicast traffic.

Monitoring Packet Drop Statistics

You can capture the incoming or outgoing traffic on an interface and send a copy of this traffic to an external interface for monitoring by a traffic analyzer. To capture traffic and forward it to an external interface, use the **monitor session** command.

When capturing traffic, these restrictions apply:

- The incoming captured traffic is not filtered.
- The incoming captured traffic is not rate limited to the capture destination.

Monitoring Dropped Packets Using Monitor Session Commands

This example shows how to use the **monitor session** command to capture and forward traffic to an external interface:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# monitor session 1 source vlan 44 both
Router(config)# monitor session 1 destination interface g9/1
Router(config)# end
Router#
2w0d: %SYS-5-CONFIG_I: Configured from console by console
```

This example shows how to use the **show monitor session** command to display the destination port location:

```
Router# show monitor session 1
Session 1
-----
Source Ports:
  RX Only:      None
  TX Only:      None
  Both:         None
Source VLANs:
  RX Only:      None
  TX Only:      None
  Both:         44
Destination Ports: Gi9/1
Filter VLANs:    None
```


Monitoring Dropped Packets Using show tcam interface Command

PFC3B and PFC3BXL mode supports ACL hit counters in hardware. You can use the **show tcam interface** command to display each entry in the ACL TCAM.

This example shows how to use the **show tcam interface** command to display the number of times the entry was hit:

```
Router# show tcam interface fa5/2 acl in ip detail
```

```
-----
DPort - Destination Port      SPort - Source Port          TCP-F - U -URG Pro - Protocol
I      - Inverted LOU         TOS   - TOS Value              - A -ACK rtr - Router
MRFM  - M -MPLS Packet       TN     - T -Tcp Control           - P -PSH COD - C -Bank Care Flag
      - R -Recirc. Flag       - N -Non-cachable         - R -RST     - I -OrdIndep. Flag
      - F -Fragment Flag     CAP   - Capture Flag            - S -SYN     - D -Dynamic Flag
      - M -More Fragments    F-P   - FlowMask-Prior.        - F -FIN T   - V(Value)/M(Mask)/R(Result)
X      - XTAG                 (*)  - Bank Priority
-----
```

```
Interface: 1018 label: 1 lookup_type: 0
protocol: IP packet-type: 0
```

```
+++++-----+-----+-----+-----+-----+-----+-----+-----+
|T|Index| Dest Ip Addr | Source Ip Addr| DPort | SPort | TCP-F|Pro|MRFM|X|TOS|TN|COD|F-P|
+++++-----+-----+-----+-----+-----+-----+-----+-----+
V 18396      0.0.0.0      0.0.0.0      P=0          P=0          ----- 0 ---- 0 0 -- --- 0-0
M 18404      0.0.0.0      0.0.0.0      0            0            0 ---- 0 0
R rslt: L3_DENY_RESULT          rtr_rslt: L3_DENY_RESULT

V 36828      0.0.0.0      0.0.0.0      P=0          P=0          ----- 0 ---- 0 0 -- --- 0-0
M 36836      0.0.0.0      0.0.0.0      0            0            0 ---- 0 0
R rslt: L3_DENY_RESULT (*)      rtr_rslt: L3_DENY_RESULT (*)
Router#
```

You can also use the TTL and IP options counters to monitor the performance of the Layer 3 forwarding engine.

This example shows how to use the **show mls statistics** command to display packet statistics and errors associated with the Layer 3 forwarding engine:

```
Router# show mls statistics
```

```
Statistics for Earl in Module 6
```

```
L2 Forwarding Engine
  Total packets Switched          : 25583421

L3 Forwarding Engine
  Total packets L3 Switched       : 25433414 @ 24 pps

Total Packets Bridged             : 937860
Total Packets FIB Switched        : 23287640
Total Packets ACL Routed          : 0
Total Packets Netflow Switched    : 0
Total Mcast Packets Switched/Routed : 96727
Total ip packets with TOS changed : 2
Total ip packets with COS changed : 2
Total non ip packets COS changed  : 0
Total packets dropped by ACL      : 33
Total packets dropped by Policing  : 0
```

```

Errors
  MAC/IP length inconsistencies      : 0
  Short IP packets received         : 0
  IP header checksum errors         : 0
  TTL failures                      : 0
<----- TTL counters
  MTU failures                      : 0
<-----MTU failure counters

Total packets L3 Switched by all Modules: 25433414 @ 24 pps

```

Monitoring Dropped Packets Using VACL Capture

The VACL capture feature allows you to direct traffic to ports configured to forward captured traffic. The capture action sets the capture bit for the forwarded packets so that ports with the capture function enabled can receive the packets. Only forwarded packets can be captured.

You can use VACL capture to assign traffic from each VLAN to a different interface.

VACL capture does not allow you to send one type of traffic, such as HTTP, to one interface and another type of traffic, such as DNS, to another interface. Also, VACL capture granularity is only applicable to traffic switched locally; you cannot preserve the granularity if you direct traffic to a remote router.

This example shows how to use VACL capture to capture and forward traffic to a local interface:

```

Router(config-if)# switchport capture
Router(config-if)# switchport capture allowed vlan add 100

```

Displaying Rate-Limiter Information

The **show mls rate-limit** command displays information about the configured rate limiters.

The **show mls rate-limit usage** command displays the hardware register that is used by a rate-limiter type. If the register is not used by any rate-limiter type, Free is displayed in the output. If the register is used by a rate-limiter type, Used and the rate-limiter type are displayed.

In the command output, the rate-limit status could be one of the following:

- On indicates that a rate for that particular case has been set.
- Off indicates that the rate-limiter type has not been configured, and the packets for that case are not rate limited.
- On/Sharing indicates that a particular case (not manually configured) is affected by the configuration of another rate limiter belonging to the same sharing group.
- A hyphen indicates that the multicast partial-SC rate limiter is disabled.

In the command output, the rate-limit sharing indicates the following information:

- Whether sharing is static or dynamic
- Group dynamic sharing codes

To display the configured rate limiters, use the **show mls rate-limit** command:

```

Router# show mls rate-limit
Sharing Codes: S - static, D - dynamic
Codes dynamic sharing: H - owner (head) of the group, g - guest of the group

Rate Limiter Type      Status   Packets/s  Burst  Sharing
-----
MCAST NON RPF         Off      -          -      -
MCAST DFLT ADJ        On       100000    100    Not sharing

```

```

MCAST DIRECT CON Off - -
ACL BRIDGED IN Off - -
ACL BRIDGED OUT Off - -
IP FEATURES Off - -
ACL VACL LOG On 2000 1 Not sharing
CEF RECEIVE Off - -
CEF GLEAN Off - -
MCAST PARTIAL SC On 100000 100 Not sharing
IP RPF FAILURE On 100 10 Group:0 S
TTL FAILURE Off - -
ICMP UNREAC. NO-ROUTE On 100 10 Group:0 S
ICMP UNREAC. ACL-DROP On 100 10 Group:0 S
ICMP REDIRECT Off - -
MTU FAILURE Off - -
MCAST IP OPTION Off - -
UCAST IP OPTION Off - -
LAYER_2 PDU Off - -
LAYER_2 PT Off - -
IP ERRORS On 100 10 Group:0 S
CAPTURE PKT Off - -
MCAST IGMP Off - -
MCAST IPv6 DIRECT CON Off - -
MCAST IPv6 *G M BRIDG Off - -
MCAST IPv6 *G BRIDGE Off - -
MCAST IPv6 SG BRIDGE Off - -
MCAST IPv6 ROUTE CNTL Off - -
MCAST IPv6 DFLT DROP Off - -
MCAST IPv6 SECOND. DR Off - -
Router#

```

To display the usage of the hardware rate limiters, use the **show mls rate-limit usage** command:

```

Router# show mls rate-limit usage
          Rate Limiter Type      Packets/s      Burst
          -----
Layer3 Rate Limiters:
RL# 0: Free - - -
RL# 1: Free - - -
RL# 2: Free - - -
RL# 3: Used
          MCAST DFLT ADJ      100000      100
RL# 4: Free - - -
RL# 5: Free - - -
RL# 6: Used
          IP RPF FAILURE      100      10
          ICMP UNREAC. NO-ROUTE      100      10
          ICMP UNREAC. ACL-DROP      100      10
          IP ERRORS      100      10
RL# 7: Used
          ACL VACL LOG      2000      1
RL# 8: Rsvd for capture - - -

Layer2 Rate Limiters:
RL# 9: Reserved
RL#10: Reserved
RL#11: Free - - -
RL#12: Free - - -
Router#

```

Understanding How Control Plane Policing Works

The control plane policing (CoPP) feature increases security on the Cisco 7600 series router by protecting the MSFC from unnecessary or DoS traffic and giving priority to important control plane and management traffic. The PFC3 and DFC3 provide hardware support for CoPP. CoPP works with the PFC3 rate limiters.



Note

The Supervisor Engine 2 does not support CoPP.

The PFC3 supports the built-in “special case” rate limiters that can be used when an ACL cannot classify particular scenarios, such as IP options cases, TTL and MTU failure cases, packets with errors, and multicast packets. When enabling the special-case rate limiters, the special-case rate limiters override the CoPP policy for packets matching the rate-limiter criteria.

The traffic managed by the MSFC is divided into three functional components or *planes*:

- Data plane
- Management plane
- Control plane

The majority of traffic managed by the MSFC is handled by way of the control and management planes. You can use CoPP to protect the control and management planes, and ensure routing stability, reachability, and packet delivery. CoPP uses a dedicated control plane configuration through the modular QoS CLI (MQC) to provide filtering and rate-limiting capabilities for the control plane packets.

CoPP Default Configuration

CoPP is disabled by default.

CoPP Configuration Guidelines and Restrictions

When configuring CoPP, follow these guidelines and restrictions:

- In releases earlier than Release 12.2(18)SXE, the PFC3 does not support the MQC class-default in hardware; the class default is replaced with a normal class map. If you define a catch-all map, the MQC class-default is supported in hardware.
- Classes that match multicast are not applied in hardware but are applied in software.
- CPP is not supported in hardware for broadcast packets. The combination of ACLs, traffic storm control, and CPP software protection provides protection against broadcast DoS attacks.
- CoPP does not support ARP policies. ARP policing mechanisms provide protection against ARP storms.
- 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.

- With PFC3A, egress QoS and CoPP cannot be configured at the same time. In this situation, CoPP is performed in the software. A warning message is displayed to inform you that egress QoS and CoPP cannot be configured at the same time.
- 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 routers. Filtering this traffic could prevent remote access to the router, requiring a console connection.
- PFC3 supports 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.
- CoPP is not enabled in hardware unless MMLS QoS is enabled globally with the **mls qos** command. If the **mls qos** command is not entered, CoPP will only work in software and will not provide any benefit to the hardware.
- 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 mls 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 is not supported in hardware for multicast packets. The combination of ACLs, multicast CPU rate limiters and CoPP software protection provides protection against multicast DoS attacks.
- 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.
- CoPP does not support MAC ACLs.

Configuring CoPP

CoPP uses MQC to define traffic classification criteria and to specify the configurable policy actions for the classified traffic. You must first identify the traffic to be classified by defining a class map. The class map defines packets for a particular traffic class. After you have classified the traffic, you can create policy maps to enforce policy actions for the identified traffic. The **control-plane** global configuration command allows the CoPP service policies to be directly attached to the control plane.

For information on how to define the traffic classification criteria, refer to the [“Defining Traffic Classification”](#) section on page 36-32.

To configure CoPP, perform this task:

	Command	Purpose
Step 1	Router(config)# mls qos	Enables MLS QoS globally.
Step 2	Router(config)# ip access-list extended <i>access-list-name</i> Router(config-ext-nacl)# { permit deny } <i>protocol source source-wildcard</i> <i>destination destination-wildcard</i> [precedence precedence] [tos tos] [established] [log log-input] [time-range <i>time-range-name</i>] [fragments]	Defines ACLs to match traffic: <ul style="list-style-type: none"> • 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. Note You must configure ACLs in most cases to identify the important or unimportant traffic.
Step 3	Router(config)# class-map <i>traffic-class-name</i> Router(config-cmap)# match { ip precedence } { ip dscp } <i>access-group</i>	Defines the packet classification criteria. Use the match statements to identify the traffic associated with the class.
Step 4	Router(config)# policy-map <i>service-policy-name</i> Router(config-pmap)# class <i>traffic-class-name</i> Router(config-pmap-c)# police { <i>bits-per-second</i> [<i>normal-burst-bytes</i>]} [<i>maximum-burst-bytes</i>] [pir peak-rate-bps] [conform-action action] [exceed-action <i>action</i>] [violate-action action]	Defines a service policy map. Use the class <i>traffic-class-name</i> command to associate classes to the service policy map. Use the police statements to associate actions to the service policy map.
Step 5	Router(config)# control-plane Router(config-cp)#	Enters the control plane configuration mode.
Step 6	Router(config-cp)# service-policy input <i>service-policy-name</i>	Applies the QoS service policy to the control plane.

When defining the packet classification criteria, follow these guidelines and restrictions:

- 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. QoS ACLs supported 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.
- In releases earlier than Release 12.2(18)SXE, the MQC class default is not supported.

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.

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 mls qos ip** command:

```
Router# show mls 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
```

Defining Traffic Classification

The following sections contain information on how to classify CoPP traffic:

- [Traffic Classification Overview, page 36-32](#)
- [Traffic Classification Guidelines, page 36-33](#)
- [Sample Basic ACLs for CoPP Traffic Classification, page 36-33](#)

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 router. Traffic should be permitted but should never pose a risk to the router; 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 MSFC resources, starving other important processes; CoPP can be used to rate limit ARP packets to prevent this situation. Currently, 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 MSFC. The undesirable classification is particularly useful when known traffic destined for the router 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 MSFC 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 MSFC. 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 accommodate 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 on page 36-33](#).

Traffic Classification Guidelines

When defining traffic classification, follow these guidelines and 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 MSFC) 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 router'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 router:

```
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 router 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 router 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 router:

```
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 router 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 router-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 router that originated the pings:

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

This example shows how to allow pings to the router:

```
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
```



Note

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
```

Configuring Sticky ARP

Sticky ARP prevents MAC address spoofing by ensuring that ARP entries (IP address, MAC address, and source VLAN) do not get overridden. The router maintains ARP entries in order to forward traffic to end devices or other routers. ARP entries are usually updated periodically or modified when ARP broadcasts are received. During an attack, ARP broadcasts are sent using a spoofed MAC address (with a legitimate IP address) so that the router learns the legitimate IP address with the spoofed MAC address and begins to forward traffic to that MAC address. With sticky ARP enabled, the router learns the ARP entries and does not accept modifications received through ARP broadcasts. If you attempt to override

the sticky ARP configuration, you will receive an error message. For a complete description of the system error messages, refer to the *Cisco 7600 Series Router Cisco IOS System Message Guide* at this URL:

http://www.cisco.com/en/US/docs/ios/12_2sx/system/messages/122sxsms.html

**Note**

Release 12.2(18)SXF and later releases support sticky ARP configurability.

To configure sticky ARP on a Layer 3 interface, perform the following task:

	Command	Purpose
Step 1	Router(config)# interface <i>type</i> ¹ <i>slot/port</i>	Selects the interface on which sticky ARP is applied.
Step 2	Router(config-if)# ip sticky-arp	Enables sticky ARP.
	Router(config-if)# no ip sticky-arp ignore	Removes the previously configured sticky ARP command.
Step 3	Router(config-if)# ip sticky-arp ignore	Disables sticky ARP.

1. *type* = ethernet, fastethernet, gigabitethernet, or tengigabitethernet

This example shows how to enable sticky ARP on interface 5/1:

```
Router# configure terminal
Router(config)# interface gigabitethernet 5/1
Router(config-if)# ip sticky-arp
Router(config-if)# end
Router#
```

**Tip**

For additional information (including configuration examples and troubleshooting information), see the documents listed on this page:

http://www.cisco.com/en/US/products/hw/routers/ps368/tsd_products_support_series_home.html

